

TEKNILLINEN KORKEAKOULU
Sähkö- ja tietoliikennetekniikan osasto

Juha Merimaa

REAALIAIKAINEN AURALISAATIO SIGNAALIPROSESSORILLA

Diplomityö, joka on jätetty opinnäytteenä tarkastettavaksi
diplomi-insinöörin tutkintoa varten Espoossa 13.6.2000.

Työn valvoja

Professori Matti Karjalainen

Tekijä:	Juha Merimaa	
Työn nimi:	Reaaliaikainen auralisaatio signaaliprosessorilla	
Päivämäärä:	13.6.2000	Sivumäärä: 63
Osasto:	Sähkö- ja tietoliikennetekniikan osasto	
Professuuri:	S-89 Akustiikka ja äänenkäsittelytekniikka	
Työn valvoja:	Professori Matti Karjalainen	
<p>Tämän diplomityön tavoitteena on ollut suunnitella ja toteuttaa reaaliaikaisesti toimiva kahdeksankanavainen auralisaattori. Auralisaatiolla ei tässä pyritä simuloimaan huoneakustiikkaa, vaan parantamaan signaalien erottuvuutta monikanavakuuntelussa. Auralisaatiomenetelmänä on käytetty binauraalista kuuloke- ja kaiutinauralisaatiota sekä VBAP-algoritmillä toteutettua amplitudipanorointia.</p> <p>Työssä käydään ensin läpi suuntakuulon toimintaa ja eri auralisaatiomenetelmien teoriaa. Binauraalinen auralisaatio perustuu kuulijan korviin saapuvan äänen mallintamiseen. Äänentoistomenetelmänä voidaan käyttää stereokuulokkeita tai -kaiuttimia. Amplitudipanoroinnissa suuntavaikutelma luodaan toistamalla sama signaali kahdesta tai useammasta kaiuttimesta eri amplitudeilla.</p> <p>Seuraavaksi esitellään projektissa käytetty laitteisto ja käsitellään reaaliaikaisen auralisaation toteutusta. Rajallisen laskentakapasiteetin takia suodatinsuunnittelu ja suodatinrakenteet vaikuttavat huomattavasti binauraalisen auralisaation toimintaan. Toteutuksen pohjana on käytetty Analog Devicesin ADSP-21061 SHARC -signaaliprosessoria ja KEMAR-keinopään HRTF-vasteita. Binauraalisen auralisaation lähdekoodi on kirjoitettu assembler-kielillä ja optimoitu mahdollisimman tehokkaaksi. VBAP-algoritmi on ohjelmoitu C-kielillä.</p> <p>Lopuksi käsitellään lyhyesti auralisaation reaaliaikaisen ohjauksen asettamia vaatimuksia ja ohjauksen toteutusta. Ohjaus voidaan suorittaa sitä varten suunnitellulla laitteistolla tai PC:n avulla. Auralisaation toiminnan kokeellinen arviointi ei kuulunut työn tavoitteisiin, mutta tuloksista esitetään subjektiivisia arvioita.</p>		
Avainsanat:	auralisaatio, suuntakuulo, binauraalinen tekniikka, amplitudipanorointi, VBAP, digitaalinen signaalinkäsittely, signaaliprosessori	

HELSINKI UNIVERSITY OF TECHNOLOGY ABSTRACT OF THE
MASTER'S THESIS

Author:	Juha Merimaa	
Name of the Thesis:	Real-Time Auralization with a Signal Processor	
Date:	June 13, 2000	Number of pages: 63
Department:	Electrical and Communications Engineering	
Professorship:	S-89 Acoustics and Audio Signal Processing	
Supervisor:	Professor Matti Karjalainen	
	<p>The objectives of the thesis were to design and implement a stand-alone real-time 8-channel auralizer. The purpose of the auralization is, instead of simulating room acoustics, to enhance multi-channel listening. The auralization methods include binaural headphone and loudspeaker auralization as well as amplitude panning. Amplitude panning is implemented with VBAP algorithm.</p> <p>An overview of spatial hearing and different auralization methods is given first. Binaural auralization relies on modeling the sounds signals at the listener's ears. Binaural sound can be reproduced with stereo headphones or loudspeakers. In amplitude panning, localization cues are created by applying the same sound signal to two or more loudspeakers with different amplitudes.</p> <p>The hardware of the project is described and limitations of a real-time implementation are discussed. Due to limited computational power, different filter design and implementation methods can have great effect on binaural auralization. The implementation is based on an Analog Devices ADSP-21061 SHARC digital signal processor and KEMAR dummy head HRTFs. The binaural auralization software is written in the SHARC assembly language and optimized carefully. The VBAP algorithm has been implemented with the C programming language.</p> <p>Finally, the requirements and implementation of a real-time control system for the auralization are discussed. Possible controlling methods include the use of specific hardware or PC software. Formal evaluation of the performance of the auralizer is beyond the scope of this theses. Informal results are, however, presented.</p>	
Keywords:	auralization, directional hearing, binaural technology, amplitude panning, VBAP, digital signal processing, signal processor	

Alkulause

Tämä diplomityö on tehty Teknillisessä korkeakoulussa Akustiikan ja äänenkäsittelytekniikan laboratoriossa osana Maanpuolustuksen tieteellisen neuvottelukunnan rahoittamaa projektia. Tilaajan puolesta yhteyshenkilönä on toiminut Seppo Madekivi, jolle lausun kiitokset mielenkiintoisen projektin käynnistämisestä. Työ on ollut haastavaa ja mikäli mahdollista, lisännyt entisestään kiinnostustani äänenkäsittelyyn ja psykoakustiikkaan.

Työskentely Akustiikan laboratoriossa on ollut mukavaa. Erityisesti haluan kiittää työn valvojaa ja ohjaajaa, professori Matti Karjalaista, joka muiden kiireidensä lomassa ehti lukea ja kommentoida hyvin nopealla aikataululla kirjoitettua diplomityötäni. Myös muulta laboratorion henkilökunnalta olen aina tarvittaessa saanut tukea ja neuvoja työhön liittyvissä asioissa.

Monet henkilöt ovat omalla työllään vaikuttaneet tämän projektin edistymiseen. Kari Haution suunnittelema ja rakentama signaaliprosessorikortin monikanava-laajennus on ollut käytössä työni alusta alkaen. Ilman Ville Pulkin kehittämää VBAP-algoritmia suunnittelemani auralisaattori ei olisi yhtä monipuolinen. Kiitos myös lähdekoodien antamisesta käyttööni. Projektin aikana apua on ollut myös Klaus Riedererin akustiikan laboratoriossa suorittamista HRTF-mittauksista.

Lopuksi kiitokset Elinalle, vanhemmilleni, ystävilleni sekä Veijo Speksille. Siitä hyvästä, että olette olemassa.

Otaniemessä 12. kesäkuuta 2000

Juha Merimaa

Sisältö

Alkulause	i
Sisältö	ii
Symboli- ja lyhenneluettelo	iv
1 Johdanto	1
2 Binauraalinen tekniikka	3
2.1 Ihmisen tila- ja suuntakuulo	4
2.2 HRTF-vasteet	6
2.3 Binauraalinen synteesi	9
2.4 HRTF-vasteiden interpolointi	13
2.5 Kuulokekuuntelu ja -kompensointi	15
2.6 Binauraalinen kaiutintoisto	17
3 Amplitudipanorointi	23
3.1 Horisontaalinen amplitudipanorointi	24
3.2 VBAP	25
3.3 Kolmiulotteinen VBAP	27
3.4 Monikanavaiset äänentoistojärjestelmät	28
3.5 Amplitudipanoroinnin erityispiirteitä	33
4 Laitteisto	34
4.1 ADSP-21061 SHARC	34
4.2 Monikanavalaajennus	36
4.3 Käyttöliittymä	37
4.4 Ohjaus PC:llä	38
5 Toteutus	40
5.1 Binauraalinen synteesi	40
5.2 Kuulokekompensointi	45
5.3 Ristiinkuulumisen esto	47
5.4 VBAP-algoritmi	50
5.5 Reaaliaikainen ohjaus	52

6 Tulosten arviointi	55
7 Yhteenveto	57
Viitteet	58
A Esimerkkejä lähdekoodista	64
A.1 HRTF-suodatin	64
A.2 Ristiinkuulumisen esto	65
A.3 VBAP	66

Symboli- ja lyhenneluettelo

c_φ	Vaakakulman interpolaatiokerroin.
c_δ	Korkeuskulman interpolaatiokerroin.
g	Amplitudipanoroinnin vahvistuskerroin.
$G(z)$	Ristiinkuulumiseneston suodatinmatriisi.
$H_c(z)$	Kontralateraalin siirtofunktio.
$H_i(z)$	Ipsilateraalinen siirtofunktio.
l	Kaiuttimen suuntavektori.
L	Kaiutinmatriisi.
p	Virtuaalisen äänilähteen suuntavektori.
r	Äänilähteen etäisyys.
\hat{x}_l	Vasemman kaiuttimen signaali.
\hat{x}_r	Oikean kaiuttimen signaali.
y_l	Kuulijan vasempaan korvaan saapuva signaali.
y_r	Kuulijan oikeaan korvaan saapuva signaali.
δ	Äänilähteen korkeuskulma.
φ	Äänilähteen vaakakulma.

AD	Analog to Digital.
ALU	Arithmetic Logic Unit.
BRIR	Binaural Room Impulse Response.
DA	Digital to Analog.
DAG	Data Address Generator.
DIP	Dual Inline Package.
DMA	Direct Memory Access.
DSP	Digital Signal Processing.
EEPROM	Electrically Erasable Programmable Read Only Memory.
EPROM	Erasable Programmable Read Only Memory.
FAQ	Frequently Asked Questions.
FIR	Finite Impulse Response.
HRIR	Head-Related Impulse Response.
HRTF	Head-Related Transfer Function.
ICE	In Circuit Emulator.
IDE	Integrated Development Environment.
IID	Interaural Intensity Difference.
IIR	Infinite Impulse Response.
ILD	Interaural Level Difference.
I/O	Input/Output.
ITD	Interaural Time Difference.
LCD	Liquid Crystal Display.
LTI	Linear Time Invariant.
MAC	Multiply Accumulate.
MLD	Masking-Level Difference.
PROM	Programmable Read Only Memory.
SHARC	Super Harvard Architecture Computer.
SRAM	Static Random Access Memory.
VBAP	Vector Base Amplitude Panning.

1 Johdanto

Suomen merialueiden liikennettä tarkkaillaan jatkuvasti monilla valvonta-asemil-la. Pinta-alusten liikkeitä seurataan sekä tutkan avulla että optisesti. Vedenalai-nen valvonta puolestaan tapahtuu kuuntelemalla alusten lähettämiä äänisignaaleja vedenalaisten mikrofonien eli hydrofonien avulla. Akustiset olosuhteet veden alla vaihtelevat hyvin voimakkaasti vuodenajan ja säätilan mukaan ja merenkäyn-nin aiheuttama kohina voi välillä olla hyvin voimakasta. Tällaisissa olosuhteissa ihminen pystyy koneita paremmin erottamaan mahdollisten alusten signaalit.

Nykyisillä menetelmillä yksi kuuntelija pystyy samanaikaisesti kuuntelemaan kah-ta tai kolmea hydrofonia. Monikanavakuuntelun ongelmana on signaalien erot-tuvuus. Useamman hydrofonin signaalit voidaan helposti summata, mutta täl-löin ne kaikki peittävät toisiaan ja helposti käy niin, että alusten heikko ääni ei enää erotu kasvaneesta taustakohinasta. Lisäksi kuuntelijan on mahdotonta suo-raan kuulohavainnon perusteella tietää, mistä hydrofonista mikäkin signaali tu-lee. Kuuntelua voidaan kuitenkin helpottaa hyödyntämällä ihmisen suuntakuulon ominaisuuksia.

Kuuntelukokeissa [Saber et al., 1991] on havaittu, että kohinan hyötysignaalia peittävä vaikutus vähenee maksimissaan jopa 15 dB siirrettäessä signaali- ja ko-hinalähde eri suuntiin kuulijan ympärillä. Lisäksi jos kanavien äänilähteet sijoitetaan eri suuntiin, kuulohavainnon perusteella on mahdollista myös päätellä, mistä kanavasta mikäkin signaali tulee. Suuntavaikutelma olisi yksinkertaisin-ta synnyttää toistamalla kaikki kuuntelukanavat omista kaiuttimistaan. Useiden kaiuttimien sijoitus pieneen kuuntelutilaan on kuitenkin melko epäkäytännöllistä. Lisäksi kuuntelutilan taustamelu ja huono akustiikka saattavat häiritä kuuntelua.

Ns. binauraalisen tekniikan avulla on mahdollista simuloida missä tahansa suun-nassa kuulijan ympärillä olevia äänilähteitä kuulokkeiden tai kahden kaiuttimen avulla. Tällaista keinotekoista tilavaikutelman luomista kutsutaan auralisaatioksi. Käsitteenä auralisaatio on analoginen näköaistin yhteydessä käytettävälle termille visualisointi [Kleiner et al., 1993]. Auralisaatio sisältää useimmiten äänilähteiden suunnan lisäksi myös huoneakustiikan simuloinnin. Yleensä huoneakustiikka aut-taakin suuntavaikutelman luomisessa. Vesikuuntelun erikoistapauksessa kaiuton kuulokeauralisaatio on kuitenkin havaittu parhaiten toimivaksi monikanavakuun-telumenetelmäksi [Maijala, 1999].

Binauraalisen tekniikan lisäksi kaiutinauralisaatioon voidaan käyttää myös perinteistä amplitudipanoroitua. Kaiuttimien välissä sijaitsevia äänilähteitä simuloidaan tällöin toistamalla sama signaali useammasta kaiuttimesta eri amplitudilla. Amplitudipanoroinnin etuna binauraaliseen kaiutinauralisaatioon nähden on tyydyttävä toiminta paljon huonommissa akustisissa olosuhteissa ja laajemmalla kuuntelualueella. Käytettävissä olevat auralisaatiosuunnat riippuvat kaiuttimien lukumäärästä ja sijoituksesta. Jos kaiuttimia on riittävästi, amplitudipanorointi saattaa olla järkevä vaihtoehto binauraaliselle tekniikalle.

Tämän työn tavoitteen on ollut suunnitella ja toteuttaa mahdollisimman yleiskäyttöinen auralisaattori vedenalaisen kuuntelun tarpeisiin. Laitteen tulee pystyä reaaliaikaisesti auralisoimaan enintään kahdeksan monofonista sisääntulosignaalia millä tahansa edellä mainituista menetelmistä. Auralisaatiosuuntien pitää lisäksi olla säädettävissä.

Auralisaattori on toteutettu Analog Devicesin ADSP-21061 SHARC -liukulusignaali prosessorilla. Prosessorin alustana toimii kahdeksankanavaiseksi laajennettu EZ-KIT Lite -evaluaatiokortti. Auralisaatio-ohjelmisto on kirjoitettu suurimmaksi osaksi assembler-ohjelmointikielellä. Auralisaation ohjausta varten korttiin on liitetty nestekidenäyttö ja painokytkimiä. Ohjaus voidaan suorittaa myös PC:llä Windows-ympäristössä toimivan ohjelman avulla. PC-ohjelmalla on samalla mahdollista päivittää auralisaatiosuodattimia.

Dokumentaatio on jaettu osiin seuraavasti: Ihmisen suuntakuulon ja binauraalisen auralisaation teoriaa käsitellään luvussa 2. Luku 3 kuvaa amplitudipanoroitua ja siihen käytettyä VBAP-algoritmia. Luku 4 sisältää tarkemman esityksen käytetystä laitteistosta. Luvussa 5 käsitellään suodatinsuunnittelua sekä auralisaation toteutusta ohjelmistotasolla. Lopuksi luku 6 esittää lyhyesti käytännön kokemuksia laitteiston toiminnasta.

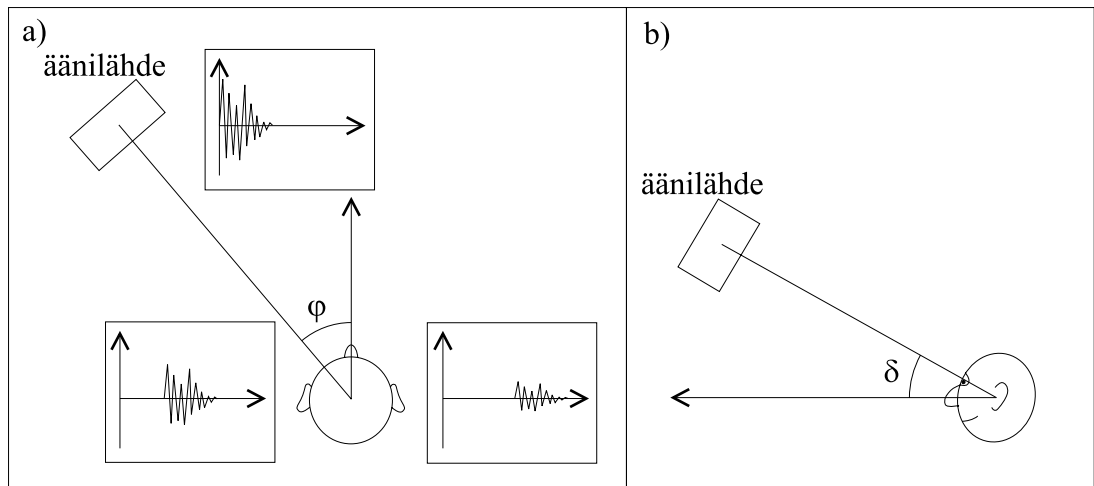
2 Binauraalinen tekniikka

Ihmisen tila- ja erityisesti suuntakuulo pohjautuvat suurelta osin ihmisen kaksikorvaisuuteen. Tilassa sijaitsevan äänilähteen signaali ei koskaan saavu täsmälleen samanlaisena kuulijan molempiin korviin. Signaalien erojen ja osittain yhteisten ominaisuuksien perusteella aivot pystyvät muodostamaan jonkinlaisen kuvan kuunteluympäristöstä, sekä yleensä melko tarkasti paikallistamaan äänilähteen. Äänilähteen paikallistumista kuulohavainnon perusteella kutsutaan lokalisaatioksi.

Binauraalisen tekniikan lähtökohtana on ajatus, että kuulijan tärykalvoille saapuvat signaalit yksin määräävät syntyvän kuulohavainnon. Jos molempiin korviin pystytään toisistaan riippumatta tuottamaan halutut signaalit, voidaan tarkasti jäljitellä mitä tahansa äänitapahtumaa ja akustista tilaa. Binauraalisten äänitysten idea on hyvin vanha. Keksijä Clement Ader teki ensimmäisiä binauraalisia kokeita jo vuonna 1881 [Sunier ja Sunier, 2000]. Nykyisissä äänityksissä ääni tallennetaan pienoismikrofoneilla ihmisen tai ns. keinopään korvista ja toistetaan myöhemmin esimerkiksi kuulokkeilla.

Digitaalisen signaalinkäsittelyn (DSP) ja signaaliprosessoreiden kehitys on mahdollistanut myös ns. binauraalisen synteessin, jossa tila- ja suuntavaikutelma liitetään jälkepäin monofoniseen äänisignaaliin. Binauraalisen synteessin avulla voidaan luoda kokonaisia virtuaalisia ääniympäristöjä. [Lehnert ja Blauert, 1991] kuvaavat erästä ensimmäisistä toteutuksista, jossa tilan mallinukseen käytetään äänilähteen ja ensimmäisten heijastusten binauraalista synteesiä sekä erillistä jälkikaiunta-algoritmia. Reaaliaikainen binauraalinen auralisaatio vaatii kuitenkin paljon prosessoritehoa ja mallinnuksessa joudutaan yleensä tekemään kompromisseja. Tässä työssä käsitellään jatkossa ainoastaan kaiuttoman tilan simuloimista.

Binauraalisen synteessin tuloksena on kaksi äänikanavaa, jotka siis sisältävät kaiken tilakuulon tarvitseman informaation. Nämä kanavat voidaan toistaa joko stereokuulokkeiden tai -kaiuttimien avulla. Kuulokkeilla on mahdollista saavuttaa hyvin tarkasti oikeassa tilassa esiintyvän äänitapahtuman kaltainen aistihavainto. Kaiutintoistossa eri korville tarkoitettujen signaalien ristiinkuuluminen pitää kompensoida ylimääräisten suodattimien avulla. Käytännössä ristiinkuuluminen ja kuuntelutilan akustiikka heikentävät kaiutintoiston lopputulosta jonkin verran.



Kuva 1: a) Etuviistossa olevan äänilähteen ääni saapuu kuulijan eri korviin hieman eri aikaan ja eri tasoisena. φ määrittää äänilähteenlähteen vaakakulman ja b) δ lähteen korkeuskulman.

2.1 Ihmisen tila- ja suuntakuulo

Ihmisen suuntakuulon toiminta perustuu pääasiassa kahden taajuusriippuvan lokalisaatiovihjeen havaitsemiseen. Äänen tullessa sivusuunnasta se saapuu toiseen korvaan hieman aikaisemmin kuin toiseen. Tästä aikaerosta käytetään lyhennettä ITD (Interaural Time Difference). Suurilla taajuuksilla pää lisäksi vaimentaa kauempaan korvaan saapuvaa signaalia jolloin korvien välille syntyy tasoero ILD (Interaural Level Difference). ITD on maksimissaan n. 0,8 ms, kun taas ILD saattaa olla kymmeniä desibelejä [Huopaniemi, 1999]. Kuva 1a) havainnollistaa ILD:n ja ITD:n syntymistä.

Äänilähteen suunta kuulijaan nähden määritellään vaakakulman φ (azimuutti, kuva 1a) ja korkeuskulman δ (elevaatio, kuva 1b) avulla niin, että suunta $\varphi = \delta = 0$ sijaitsee suoraan kuulijan edessä. Lisäksi äänilähteen paikan yksikäsitteiseen määrittämiseen tarvitaan lähteen etäisyys r . Koordinaatiston origo sijaitsee korvakäytävien välistä janan keskipisteessä. Tasoa $\delta = 0$ kutsutaan horisontaalitasoksi ja tasoa $\varphi = 0$ mediaanitasoksi. Lähempänä äänilähdettä olevaa korvaa nimitetään ipsilateraaliseksi ja kauempana olevaa kontralateraaliseksi.

Suuntakuulon tarkkuus riippuu äänilähteen suunnasta ja signaalista. Vaakakulman havaitseminen tapahtuu yleensä huomattavasti korkeuskulmaa tarkemmin.

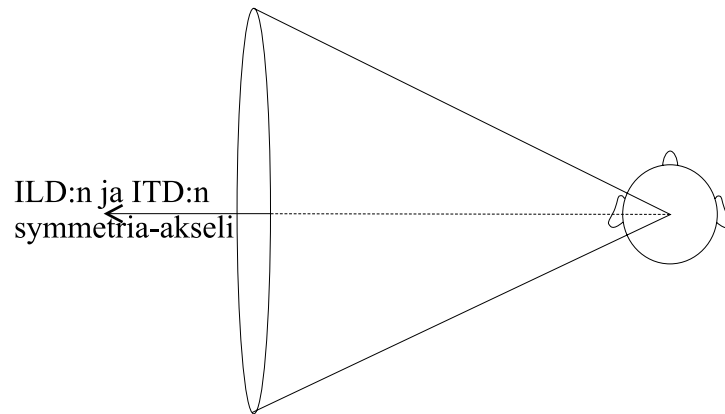
Suoraan edessä ihminen pystyy havaitsemaan laajakaistaista signaalia lähettävän äänilähteen vaakakulman parhaimmillaan n. $\pm 3,6^\circ$ tarkkuudella. Sivuilla tarkkuus on heikompi, tyypillisesti n. $\pm 10^\circ$. Kapeakaistaisilla signaaleilla virheet saattavat olla vielä suurempia. Suunnan muutokset sen sijaan havaitaan helpommin. Suoraan edessä muutosten erotuskynnys on parhaimmillaan noin 1° . [Blauert, 1997]

Mediaanitasossa korkeuskulman muutokset eivät juuri vaikuta ILD- ja ITD-vihjeisiin. Sen sijaan molempiin korviin saapuvat signaalit muuttuvat lähes samalla tavalla korkeuskulman funktiona. Muutokset johtuvat korvalehtien, hartioiden aiheuttamien heijastusten sekä pään pystysuuntaisen epäsymmetrisyyden vaikutuksesta. Koska yksilölliset erot erityisesti korvalehdissä voivat olla melko suuria, korkeuskuulon tarkkuus vaihtelee huomattavasti eri ihmisillä. Kuuntelukokeissa suoraan edessä olevan äänilähteen korkeuskulmahavainnoille on saatu tarkkuusarvoja väliltä $\pm 4-17^\circ$ [Blauert, 1997]. Kuulijan yläpuolella tarkkuus on vielä huonompi, mutta paranee sivuilla korkeuskulman muutoksen tällöin synnyttämien ILD- ja ITD-vihjeiden ansiosta.

ILD:n ja ITD:n merkittävä rooli aiheuttaa helposti sekaannuksia tiettyjen suuntien välillä. Erityisesti ITD-vihjeet ovat hyvin samanlaisia korvien läpi kulkevan akselin suhteen symmetrisissä kulmissa sijaitsevilla äänilähteillä. Tällaiset symmetriset suunnat muodostavat ns. sekaannuskartion (engl. cone of confusion, kuva 2). Esimerkiksi horisontaalitasossa 30° kulmasta tuleva ääni voi synnyttää havainnon suunnassa 150° . Ilmiöstä käytetäänkin usein nimitystä etu-takasekaannus.

Myös etu-takasekaannuksien tiheys vaihtelee huomattavasti eri ihmisten välillä. Yleensä horisontaalitasoon sekaannukset tapahtuvat aina samalla henkilöllä samaan suuntaan, mutta suunta voi olla kumpi tahansa. Etu-takasekaannukset liittyvät lähinnä tilanteisiin, joissa kuuliija ei pysty liikuttamaan päätään suhteessa todellisiin tai virtuaalisiin äänilähteisiin. Normaalitilanteessa pään kääntelyn yhteydessä tapahtuvat ILD- ja ITD-muutokset auttavat ratkomaan lähes kaikki sekaannukset. [Wightman ja Kistler, 1999]

Äänilähteen etäisyyden arviointi pelkän kuulohavainnon perusteella on huomattavasti suunnan havaitsemista vaikeampaa. Äänitaso antaa jonkinlaisia vihjeitä, mutta yleensä etäisyysarvio tehdään kuuntelutilan akustiikan avulla. Etäisyysvaikutelma syntyy monimutkaisella tavalla suoran äänen ja huoneheijastusten



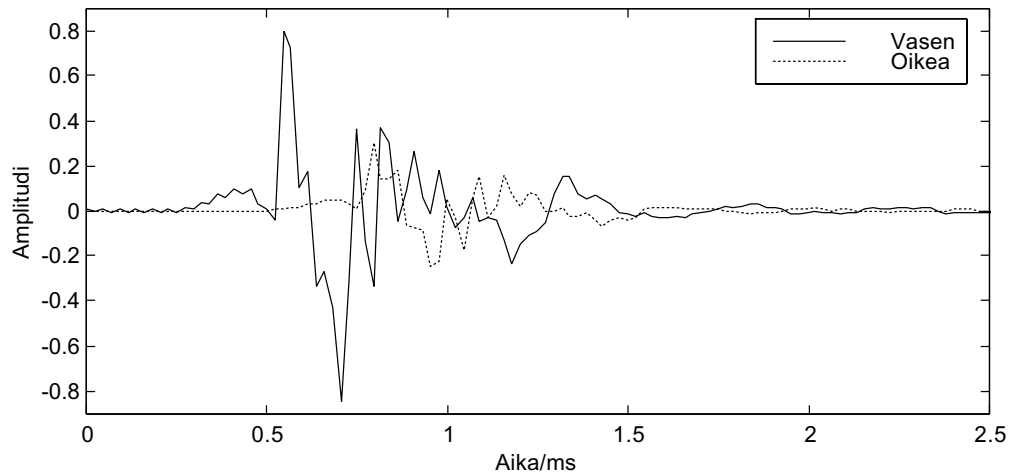
Kuva 2: Sekaannuskartio. Siirtofunktiot eri kohdissa kartion pinnalla olevista äänilähteistä kuulijan korviin eroavat hyvin vähän toisistaan. Pyöreällä päällä ILD- ja ITD-vihjeet olisivat samalla etäisyydellä aivan samanlaisia.

perusteella. Huonetilassakin tarkka havainto edellyttää, että kuuntelija tuntee tilan ja äänilähteen. Mikäli etäisyyttä halutaan simuloida, äänitapahtuma täytyy siis sitoa tiettyyn tilaan. [Blauert, 1997, Karjalainen, 1999]

2.2 HRTF-vasteet

Äänen saapumista korviin tulosuunnasta riippuen voidaan kuvata ns. HRTF-vasteiden (Head-Related Transfer Function) avulla. Tietyn suuntainen HRTF-vaste koostuu kaiuttomassa tilassa mitatuista siirtofunktioista kyseisessä suunnassa olevasta äänilähteestä kuulijan molempiin korviin. Vastaavista impulssivasteista käytetään lyhennystä HRIR (Head-Related Impulse Response). Vasteet riippuvat ihmisen pään, korvalehtien ja ylävartalon muodoista ja ovat siten yksilölliset.

HRTF-vasteet muuttuvat äänilähteen suunnan lisäksi myös etäisyyden funktiona. Etäisyyden vaikutus vasteiden sisältämiin ITD-vihjeisiin on hyvin pieni. ILD:t sen sijaan kasvavat kaikilla taajuuksilla äänilähteen siirtyessä lähemmäs. Muutoksia aiheuttavat lähikenttäilmiöt alkavat vaikuttaa äänilähteen ollessa noin metrin etäisyydellä kuulijan päästä. Erityisesti pienillä taajuuksilla muutos on huomattava, koska kaukana olevan äänilähteen synnyttämä ILD on hyvin pieni. Etäisyyden vaikutusta HRTF:iin on tutkittu sekä teoreettisesti pallonmuotoisella päämallilla [Duda ja Martens, 1998] että kokeellisesti [Huopaniemi, 1999].

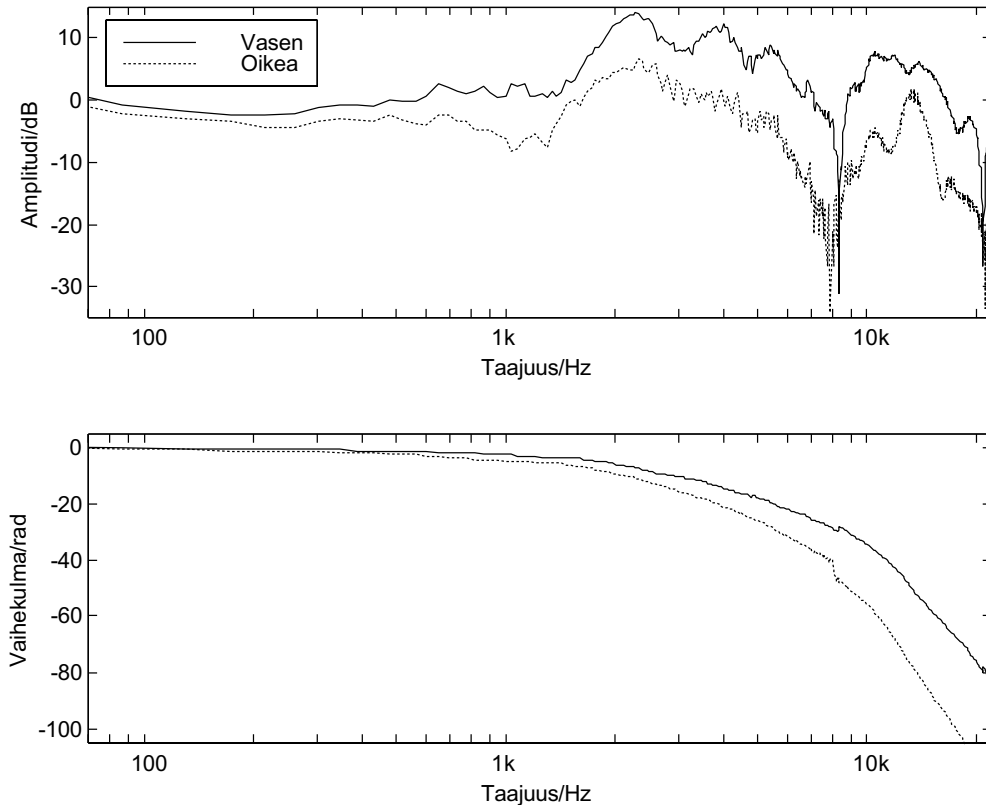


Kuva 3: KEMAR-keinopään impulssivasteet suunnasta $\varphi = 30^\circ$ saapuvalla äänelle.

HRTF:iä vastaavat siirtofunktio mittaukset voidaan tehdä myös normaalissa huone tilassa. Näin saatavista impulssivasteista käytetään nimitystä BRIR (Binaural Room Impulse Response). Huonevaste koostuu suoran äänen lisäksi useista eri suunnista eri aikoina ja tasoisina saapuvista heijastuksista, jotka riippuvat äänilähteen ja kuulijan sijainnista tilassa. Lisäksi äänilähteen suuntaavuusominaisuudet vaikuttavat vasteisiin.

Mitatut HRTF-vasteet muodostavat pohjan binauraaliselle auralisaatiolle. Monissa sovelluskohteissa yksilöllisten vasteiden mittaaminen ja käyttö on kuitenkin mahdotonta. Tällöin auralisaatioon voidaan käyttää sopivasti valittuja oikean ihmisen tai keinopään HRTF:iä. Keinopäät pyritään tekemään akustisilta ominaisuuksiltaan keskimääräisen ihmispään kaltaisiksi. Kuvassa 3 on esitetty KEMAR-keinopään impulssivasteet [Gardner ja Martin, 1994] ja kuvassa 4 vastaavat magnitudi- ja vaihevasteet horisontaalitasossa 30° suunnasta vasemmalta saapuvalla äänelle. Mittauksissa on käytetty korvalehteä DB-061.

Eri ihmisten HRTF:t eroavat toisistaan sekä ILD- että ITD-vihjeiden suhteen, mutta molemmat vihjeet muuttuvat melko samankaltaisesti suunnan funktiona [Kendall, 1995]. Ei-yksilöllisiin HRTF-vasteisiin perustuvassa auralisaatioissa ääni ei yleensä lokalisoitu aivan yhtä tarkasti kuin henkilökohtaisia vasteita käytettäessä. Horisontaalitasossa lokalisaatio on usein samaa luokkaa kuin yksilöllisillä HRTF:illä. Mediaanitasossa, jossa korvien väliset signaalierot ovat pienempiä, lo-



Kuva 4: KEMAR-keinopään magnitudi- ja vaihevasteet suunnasta $\varphi = 30^\circ$ saapuvalla äänellä.

kalisaatiovirheiden määrä kasvaa usein huomattavasti. Horisontaalitason virtuaaliset lähteet saattavat muissakin suunnissa joskus paikallistua jonkin verran tason ylä- tai alapuolelle. [Møller et al., 1996b]

Verrattaessa ei-yksilöllisten virtuaalisten lähteiden aiheuttamia aistimuksia vapaassa kentässä sijaitseviin todellisiin lähteisiin on todettu, että suurin osa ihmisistä havaitsee korkeuskulmat molemmissa tapauksissa oikein mutta hieman eri tarkkuudella. Osa ihmisistä kuitenkin havaitsee korkeuskulmat ainoastaan todellisilla lähteillä ja osa ei niilläkään. Lisäksi yksilöimättömän äänen pohjana käytetyt HRTF:t vaikuttavat asiaan niin, että jos ne ovat henkilöltä, joka ei koskaan kuule korkeuskulmia oikein, muutkaan eivät välttämättä pysty havainnoimaan äänilähteen korkeuskulmaa [Wenzel et al., 1993]. Vaakakulman havaitsemisessa yksilölliset erot ovat pienempiä.

Erot eri keinopäiden välillä ovat myös huomattavia. Hyvän keinopään HRTF:ien

avulla tuotettu binauraalinen ääni lokalisoituu yleensä suunnilleen yhtä hyvin kuin satunnaisesti valitun oikean ihmispään mittaustuloksia käytettäessä. Valitsemalla huolellisesti “tyypillinen” ihmispää, voidaan kuitenkin päästä parempaan lokalisaatiotarkkuuteen. Erityisesti mediaanitasossa parannus voi olla melko suuri. Kuuntelukokeissa on lisäksi havaittu, että lokalisaatiotarkkuus paranee usein kokeiden loppua kohti. Tämä saattaa viitata jonkinlaiseen adaptoitumiseen yksilöllisten HRTF:ien tapauksessa. [Møller et al., 1996a, Møller et al., 1997]

Paremmen lokalisaation saavuttamiseksi mitattuja HRTF:iä on yritetty käsitellä monilla tavoilla. [Zhang et al., 1998] raportoivat etu-takasekaannuksien kadonneen lähes kokonaan korostamalla HRTF:ien spektrin suuntariippuvia muutoksia. Lisäksi suurennettua päätä mallintavien HRTF:ien avulla on saatu aikaan normaalia parempia lokalisaatiotuloksia [Begault, 1994, Huopaniemi, 1999].

HRTF-vasteet voidaan mitata korvasta kolmesta eri paikasta: aivan tärykalvon edestä, avoimen korvakäytävän suulta sekä suljetun korvakäytävän suulta. Eyksilöllisessä kuuloketoistossa mittauspaijalla ei ole merkitystä, kunhan käytetään asianmukaista kuulokekompensointia (kts. kappale 2.5). Mikäli toistettava ääni ei ole yksilöllistä, mutta kuulokekompensointiin on käytettävissä kuulijan omat mittaustulokset, suljetun korvakäytävän suulta mitattujen HRTF:ien käyttö minimoi syntyvän virheen. Tällöin HRTF:issä ei ole mukana mittauspään korvakäytävän resonanssia, joka ei yksilöllisiä kompensointisuodattimia käytettäessä kumoutuisi täydellisesti. [Møller et al., 1996b].

2.3 Binauraalinen synteesi

Binauraalisessa synteessissä monofoniseen signaaliin liitetään suuntainformaatio suodattamalla signaalia kahdella digitaalisella suodattimella, jotka mallintavat halutun suuntaisia HRTF-vasteita. Tuloksena on binauraalinen signaali, joka sisältää erilliset signaalit molemmille korville. Tarkkojen HRTF:ien käyttö synteessissä on kuitenkin laskennallisesti melko raskasta, joten varsinkin reaaliaikaisissa sovelluksissa joudutaan yleensä tyytymään approksimaatioihin.

Binauraalinen synteesi voidaan kuvata lineaarisena aikainvarianttina (LTI) järjestelmänä. Merkitään monofonista sisääntulosignaalia symbolilla $x_m(n)$ ja halutun suuntaisia HRIR-vasteita $h_l(n)$:llä ja $h_r(n)$:llä. Binauraalinen signaali y saadaan

nyt konvoluutiolla

$$\mathbf{y} = \mathbf{h} * x_m(n) \quad (1)$$

missä

$$\mathbf{y} = \begin{bmatrix} y_l(n) \\ y_r(n) \end{bmatrix}, \mathbf{h} = \begin{bmatrix} h_l(n) \\ h_r(n) \end{bmatrix} \quad (2)$$

Sama voidaan esittää taajuusalueessa yhtälöillä

$$\mathbf{Y} = \mathbf{H}X_m(f) \quad (3)$$

$$\mathbf{Y} = \begin{bmatrix} Y_l(f) \\ Y_r(f) \end{bmatrix}, \mathbf{H} = \begin{bmatrix} H_l(f) \\ H_r(f) \end{bmatrix} \quad (4)$$

Käytetään tästä lähtien nimitystä HRTF kuvaamaan sekä taajuusalueen että aika-alueen esitystä. Olkoon $H(f)$ mikä tahansa yksittäinen HRTF-vaste. Tämä voidaan jakaa amplitudi- ja vaihekomponentteihin

$$H(f) = |H(f)|e^{j\varphi(f)} \quad (5)$$

missä $\varphi(f)$ kuvaa vaihetta. Mikä tahansa LTI-järjestelmä voidaan edelleen esittää minimivaiheisen suodattimen ja kokopäästösuodattimen kaskadina

$$H(f) = |H(f)|e^{j\varphi_{mp}(f)}e^{j\varphi_{ap}(f)} \quad (6)$$

HRTF:ien tapauksessa kokopäästösuodattimen vaihe $\varphi_{ap}(f)$ on lähes lineaarinen taajuuden funktio, joten sitä voidaan approksimoida taajuusriippumattomalla viiveellä. [Jot et al., 1995, Huopaniemi, 1999] Tämä approksimaatio aiheuttaa ainoastaan pienen ITD-virheen, joka on kuuntelukokeissa todettu huomaamattomaksi [Kistler ja Wightman, 1992, Kulkarni et al., 1999].

Minimivaiheinen suodatin $H_{mp}(f)$ voidaan laskea ikkunoimalla reaalista kepstriä

seuraavanlaisella ikkunalla

$$w_{mp}(n) = \begin{cases} 1, & n = 1 \text{ tai } n = N/2 + 1 \\ 2, & n = 2, \dots, N/2 \\ 0, & n = N/2 + 2, \dots, N \end{cases} \quad (7)$$

kun kepstrin pituus N on parillinen. Reaalinen kepstri määritellään magnitudivasteen logaritmin Fourier-käänteismuunnoksena

$$c(n) = F^{-1}\{\ln|H(f)|\} \quad (8)$$

Minimivaiheisen suodattimen taajuusvaste saadaan nyt ikkunoidusta kepstristä [Oppenheim ja Schaffer, 1975, s. 357]

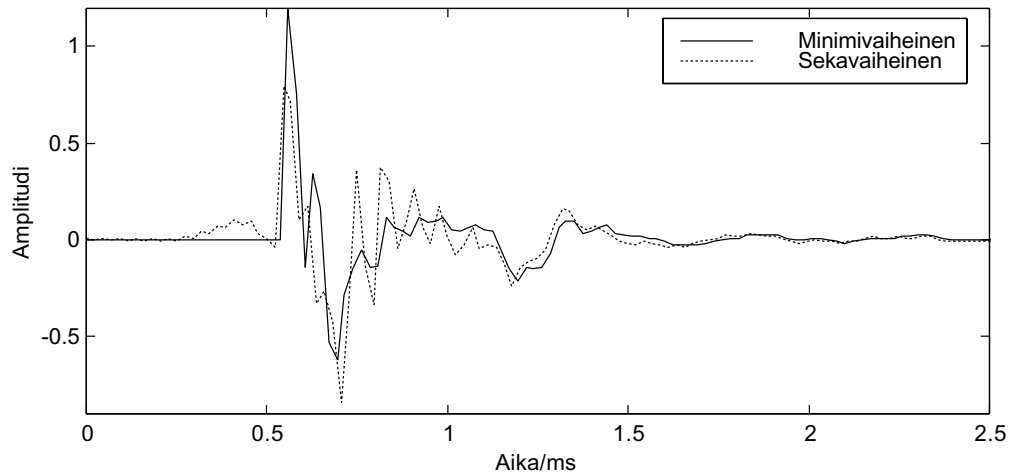
$$H_{mp}(f) = e^{F\{c(n)w_{mp}(n)\}} \quad (9)$$

Minimivaiheinen impulssivaste voidaan edelleen laskea Fourier-käänteismuunnoksella. Kokopäästösuodattimen vaihe saadaan yksinkertaisesti vähentämällä minimivaiheisen suodattimen vaihe kokonaisvaiheesta

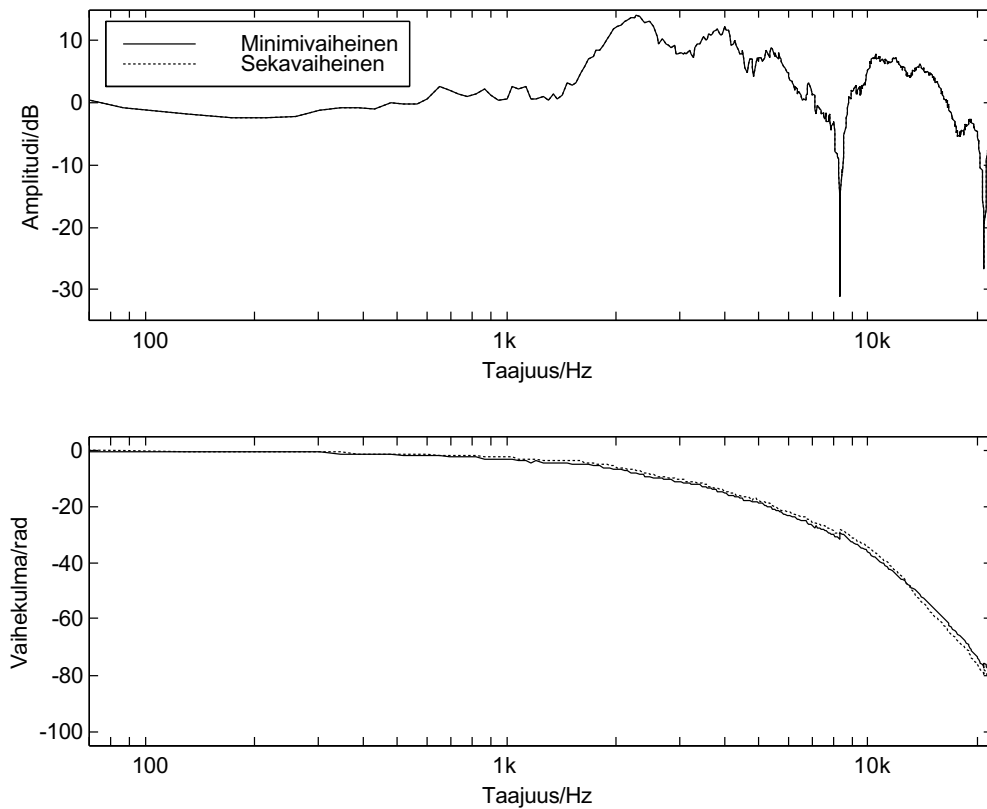
$$\varphi_{ap}(f) = \varphi(f) - \varphi_{mp}(f) \quad (10)$$

Tämän “ylimääräisen” vaiheen toteuttaman viiveen approksimaatio saadaan sovitamalla vaihevasteeseen φ_{ap} suora välille 1 kHz - 5 kHz [Jot et al., 1995]. Kuvassa 5 on esitetty horisontaalitasossa 30° suunnasta vasempaan korvaan saapuvan äänen impulssivaste sekä sen approksimaatio taajuusriippumattoman viiveen ja minimivaiheisen suodattimen avulla. Kuva 6 esittää vastaavat magnitudi- ja vaihevasteet. Suoran sovitus vaihefunktioon tapahtuu lineaarisella taajuusasteikolla.

Minimivaiheisen suodattimen z -muunnoksen kaikki nollat sijaitsevat kompleksitasossa yksikköympyrän sisäpuolella. Tällaisen suodattimen aiheuttama energiaviive on kaikista mahdollisista saman magnitudivasteen toteuttavista suodattimista pienin. [Oppenheim ja Schaffer, 1975] Näinollen rajallisella laskentakapasiteetilla voidaan energiamielessä tarkimmin toteuttaa juuri minimivaiheinen suodatin. Erillisen viiveen toteutus taas on laskennallisesti hyvin kevyt operaatio. Nollien sijainti yksikköympyrän sisällä aiheuttaa myös sen, että suodattimelle on olemassa stabiili käänteissuodatin.



Kuva 5: KEMAR-keinopään vasempaan korvaan suunnasta $\varphi = 30^\circ$ saapuvan äänen impulssivaste sekä tämän minimivaiheinen rekonstruktio.



Kuva 6: KEMAR-keinopään magnitudi- ja vaihevasteet $\varphi = 30^\circ$ suunnasta vasempaan korvaan saapuvalla äänellä sekä tämän minimivaiheiselle rekonstruktioille.

Jos binauraaliseen synteisiin käytetään binauraalisia huonevasteita (BRIR), edellä kuvattua minimivaiheista rekonstruktiota ei voida käyttää. Huomattavaa on myös, että äänilähteiden erilaisen suuntaavuuden takia synteessin tuloksena saatava kuulohavainto vastaa mittauksessa käytetyn kaiuttimen toistamaa signaalia tilassa, eikä välttämättä havaintoa, jonka haluttu äänilähde kyseisessä tilassa aiheuttaisi. [Jot et al., 1995]. Huonetiloja voidaan kuitenkin simuloida tehokkaammin mallintamalla suora ääni sekä joitakin ensimmäisiä heijastuksia kaiuttomien HRTF-suodattimien avulla ja diffuusi jälkikaiunta erillisellä algoritmilla. [Kleiner et al., 1993, Huopaniemi, 1999]

2.4 HRTF-vasteiden interpolointi

HRTF-vasteiden mittaaminen riittävän tiheään kaikista tarvittavista suunnista on käytännössä mahdotonta. Myös toteutus rajoittaa yleensä suurinta mahdollista käytettävää datamäärää, joten osa vasteista joudutaan laskemaan interpoloimalla lähimpiä käytettävissä olevia vasteita. Interpoloation tuloksena saatavien vasteiden todenmukaisuus ei ole aivan itsestään selvää, mutta sopivilla menetelmillä voidaan käytännössä saada hyviä tuloksia.

Kaksiulotteisessa tapauksessa interpolointi suoritetaan lähimpien halutun auralisaatiosuunnan molemmiin puoliin mitattujen HRTF:ien välillä. Yksinkertaisin toteutustapa on interpoloida mitattuja impulssivasteita lineaarisesti aika-alueessa. Sekavaiheisilla suodatintoteutuksilla tämä vastaa tapausta, jossa ääni toistetaan kahdesta mitatuissa suunnissa olevasta kaiuttimesta samanaikaisesti. HRTF:ien eripituisten viiveiden takia interpoloinnin seurauksena voi tällöin syntyä selvästi kuultavia kampsuodatinilmiöitä. HRTF:ien minimivaiheinen rekonstruktio tarjoaa mahdollisuuden suorittaa interpolointi erikseen ITD- ja ILD-vihjeille. ITD:n paloittain lineaarinen interpolointi tuottaa hyvin tarkasti oikeita tuloksia, kunhan mittauspisteet ovat riittävän lähellä toisiaan. ILD-interpolointi pitäisi suorittaa minimivaiheisten suodattimien magnitudille, jotta kampsuodatinilmiöitä voitaisiin välttyä.

Minimivaiheisen suodattimen vaihe on mahdollista laskea jälkeenpäin sen logaritmisesta magnitudista Hilbert-muunnoksella [Oppenheim ja Schaffer, 1975]. Hilbert-muunnoksen lineaarisuuden takia interpoloidusta magnitudista laskettu vaihe kuitenkin vastaa suoraan vaiheita interpoloimalla saatua tulosta. Suodatti-

men kompleksisen spektrin kompleksisen logaritmin interpolointi vastaa siis minimivaiheisessa tapauksessa magnitudi-interpolointia. Interpolointi aikatasossa taas vastaa suodattimen kompleksisen spektrin interpolointia. Jos minimivaiheisten suodattimien magnitudit eivät poikkea kovin paljon toisistaan, myös vaiheet ovat edellisen perusteella lähes samat. Kun suodatinkertoimien interpolointi tehdään siis riittävän tiheästi mitattujen minimivaiheisten HRTF-vasteiden välillä, pahoja kampsuodatinilmiöitä ei esiinny. [Jot et al., 1995]

Teoriassa magnitudi-interpolointikaan ei ole paras mahdollinen menetelmä, koska se eliminoi taajuussuunnassa tapahtuvat muutokset niin, että spektrissä mahdollisesti liikkuvat terävät piikit tai kuopat vaimenevat mittauspisteiden välillä. Paras tapa olisi kuvata interpolointia varten vaakakulman funktiona jaksollisesti muuttuvat suureet, kuten taajuusriippumaton viive, Fourier-sarjan avulla [Jot et al., 1995]. Viiveen ja minimivaiheisten suodatinkertoimien paloittain lineaarisella interpoloinnilla on kuitenkin saatu hyviä tuloksia. Kuuntelukokeissa [Wenzel ja Foster, 1993] lokalisaatiotarkkuus ei juuri kärsinyt, vaikka auralisaatioon käytetyt suodattimet interpoloitiin 30° päässä toisistaan olevista mittaustuloksista. Sekavaiheisilla FIR-suodattimilla sen sijaan kampsuodatinilmiöt olivat selvästi havaittavissa erityisesti lähteiden tai kuulijan liikkeessa.

Kaksiulotteisessa tapauksessa haluunsuuntaisen minimivaiheisen HRTF-suodattimen kertoimet $h_E(n)$ voidaan laskea viereisten suodattimien kertoimista $h_A(n)$ ja $h_B(n)$ kaavalla

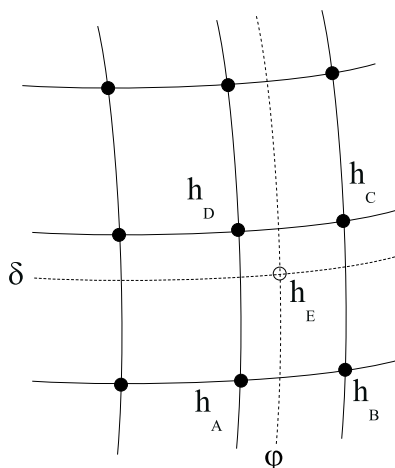
$$h_E(n) = (1 - c_\varphi)h_A(n) + c_\varphi h_B(n) \quad (11)$$

missä c_φ on vaakakulman interpolaatiokerroin. Jos suodattimia A ja B vastaavien auralisaatiosuuntien vaakakulmat ovat φ_A ja φ_B , interpolaatiokerroin kulmalle φ saadaan kaavalla

$$c_\varphi = \frac{\varphi - \varphi_A}{\varphi_B - \varphi_A} \quad (12)$$

Jos interpoloinnissa käytettävät suodattimet edustavat tasavälein φ_{grid} valittuja vaakakulmia, interpolaatiokerroin voidaan esittää muodossa

$$c_\varphi = \frac{\varphi \bmod \varphi_{grid}}{\varphi_{grid}} \quad (13)$$



Kuva 7: Minimivaiheiset HRTF-suodatinkertoimet $h_E(n)$ saadaan bilineaarisella interpoloinnilla ympäröivien pisteiden välillä.

Kolmiulotteisessa tapauksessa interpolointi voidaan suorittaa bilineaarisesti neljän lähimmän auralisaatiosuunnan suodatinkertoimien välillä kaavalla

$$h_E(n) = (1 - c_\varphi)(1 - c_\delta)h_A(n) + c_\varphi(1 - c_\delta)h_B(n) + c_\varphi c_\delta h_C(n) + (1 - c_\varphi)c_\delta h_D(n) \quad (14)$$

Tässä c_δ kuvaa korkeuskulman interpolaatiokerrointa, joka saadaan tasavälisessä tapauksessa vastaavasti kuin vaakakulmalla

$$c_\delta = \frac{\delta \bmod \delta_{grid}}{\delta_{grid}} \quad (15)$$

Kolmiulotteista interpolointia on havainnollistettu kuvassa 7. Taajuusriippumattomien viiveiden interpolointi suoritetaan kerroininterpolointia vastaavilla kaavoilla.

2.5 Kuulokekuuntelu ja -kompensointi

Kaupalliset kuulokkeet optimoidaan yleensä kuulostamaan samalta kuin kaiuttimilla toistettu ääni. Kuten kuvasta 4 näkyy, äänilähteen ja kuulijan korvien välisen siirtofunktion taajuusvaste ei ole kovin tasainen ja vaihtelee äänilähteen sijainnin mukaan. Kuulostaakseen samalta kuin kaiuttimet kuulokkeiden pitää siis toteuttaa jonkinlainen HRTF:iä vastaava siirtofunktio. Binauraalisen syntee-

sin tuloksena saatava signaali sisältää kuitenkin jo kertaalleen HRTF:ien vaikutuksen, joten kuulokkeiden siirtofunktio pitää kompensoida pois.

Kuulokekompensointi voidaan toteuttaa joko erillisillä suodattimilla tai yhdistämällä se auralisaatiossa käytettäviin HRTF-suodattimiin. Kompensointiin riittää yleensä pelkkä ekvalisointi, jolloin käänteissuodattimet voidaan toteuttaa minimivaiheisina. Yksinkertaisimmin kompensointisuodattimet voidaan suunnitella suoraan korvista mitattujen kuulokevasteiden pohjalta. Tällöin toisto saadaan optimoitua tietylle kuulokemallille. Mittauksen suorittamiseen on kuitenkin syytä kiinnittää huomiota.

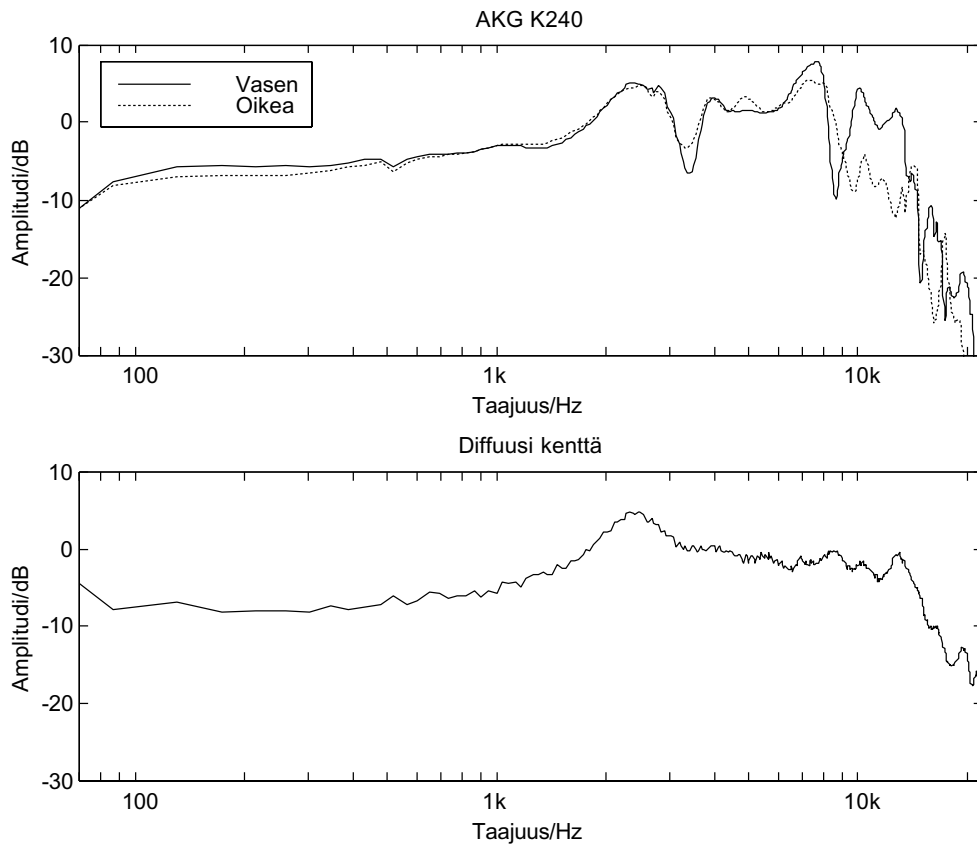
Kuulokkeiden asento vaikuttaa huomattavasti kuulokevasteisiin. Keinopäämittauksissa vasteissa on suurilla taajuuksilla havaittu yli 10 dB:n eroja kuulokkeiden eri sijoituskertojen välillä. Vaihtelu riippuu tietysti kuulokkeiden mallista ja käyttäjistä. Pahimmassa tapauksessa kuitenkin yhden mittauksen perusteella lasketut kompensointisuodattimet saattavat jopa heikentää kuuloketoistoa. Useampien mittausten keskiarvona lasketuilla suodattimilla on sen sijaan saatu melko hyviä tuloksia, vaikka täydellistä kompensointia ei näinkään saada aikaan. [Kulkarni ja Colburn, 2000]

Yleiskäyttöisessä auralisaattorissa saattaa olla järkevämpää suunnitella kompensointi kuulokkeiden yleisten suunnitteluperiaatteiden mukaisesti. Kuulokeissa käytetään yleisesti kahta erilaista ekvalisointia. Ns. vapaakenttäekvalisoinnissa referenssinä käytetään vapaassa kentässä suoraan kuulijan edessä olevan äänilähteen korviin tuottamaa signaalia. Diffuusin kentän ekvalisoinnissa puolestaan referenssinä on signaali, joka saadaan kuulijan korviin, kun ääni saapuu samanaikaisesti kaikkialta ympäriltä. Diffuusin kentän ekvalisointia pidetään yleisesti luonnollisemman kuuloisena. [Theile, 1986]

Diffuusin kentän vasteita voidaan arvioida laskemalla mitattujen HRTF:ien tehokeskiarvo erikseen molemmille korville. Jos mittaussuunnat peittävät tasaisesti kuulijaa ympäröivää pallopintaa, diffuusin kentän vasteet voidaan laskea taajuusalueessa kaavalla

$$|H_{df}(f)| = \sqrt{\frac{1}{M} \sum_{i=1}^M |H_i(f, \varphi_i, \delta_i)|^2} \quad (16)$$

missä M on kaikkien mitattujen HRTF:ien lukumäärä ja $H_i(f, \varphi_i, \delta_i)$ indeksiin i



Kuva 8: AKG K240 -kuulokkeiden vasteet sekä HRTF:istä lasketut diffuusin kentän vasteet.

liittyvän HRTF:n taajuusvaste. Tarvittaessa kutakin HRTF:ää voidaan painottaa kertoimella, joka on suhteessa siihen liittyvään avaruuskulmaan. [Jot et al., 1995]

Kuvassa 8 on esitetty KEMAR-keinopäällä mitatut [Gardner ja Martin, 1994] kuulokevasteet AKG K240 -kuulokkeille sekä HRTF:istä kaavalla (16) lasketut diffuusin kentän vasteet. Vasteet ovat muodoltaan melko samankaltaiset. Kuulokemittauksista nähdään selvästi, että eri korvien käyrissä on suurilla taajuuksilla selviä eroja.

2.6 Binauraalinen kaiutintoisto

Binauraalisen signaalin toisto stereokuulokkeilla on melko helppoa, koska kuulijan korville meneviä signaaleja voidaan kontrolloida toisistaan riippumatta. Stereokaiutintoistossa molempien kaiuttimien toistamat signaalit saapuvat molem-

piin korviin, mutta hieman erilaisena. Tämä ristiinkuuluminen voidaan kuitenkin estää erillisen signaalinkäsittelyn avulla. Ristiinkuulumisen kumoamisella toteutettua binauraalista kaiutinauralisaatiota kutsutaan joissain lähteissä myös transauraaliseksi auralisaatioksi.

Kaiutinsignaalien \hat{x}_l ja \hat{x}_r toistoa voidaan kuvata yhtälöillä

$$\mathbf{y} = \mathbf{H} \hat{\mathbf{x}} \quad (17)$$

missä

$$\mathbf{y} = \begin{bmatrix} y_l(n) \\ y_r(n) \end{bmatrix}, \quad \hat{\mathbf{x}} = \begin{bmatrix} \hat{x}_l(n) \\ \hat{x}_r(n) \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} H_{ll}(z) & H_{rl}(z) \\ H_{lr}(z) & H_{rr}(z) \end{bmatrix} \quad (18)$$

Matriisi \mathbf{H} kuvaa siirtofunktioita kaiuttimista kuulijan korviin kuvan 9 mukaisesti. $y_l(n)$ ja $y_r(n)$ ovat kuulijan korviin päätyvät signaalit. Jotta kuulijalle saataisiin toistettua binauraalinen signaali $\mathbf{x} = [x_l(n) \ x_r(n)]^T$, on etsittävä käänteismatriisi $\mathbf{G} = \mathbf{H}^{-1}$, jolla kaiutinsignaaleille $\hat{\mathbf{x}} = \mathbf{G} \mathbf{x}$ saadaan $\mathbf{y} = \mathbf{x}$. Käänteismatriisi voidaan analyttisesti kirjoittaa muodossa

$$\mathbf{G} = \frac{1}{H_{ll}(z)H_{rr}(z) - H_{lr}(z)H_{rl}(z)} \begin{bmatrix} H_{rr}(z) & -H_{rl}(z) \\ -H_{lr}(z) & H_{ll}(z) \end{bmatrix} \quad (19)$$

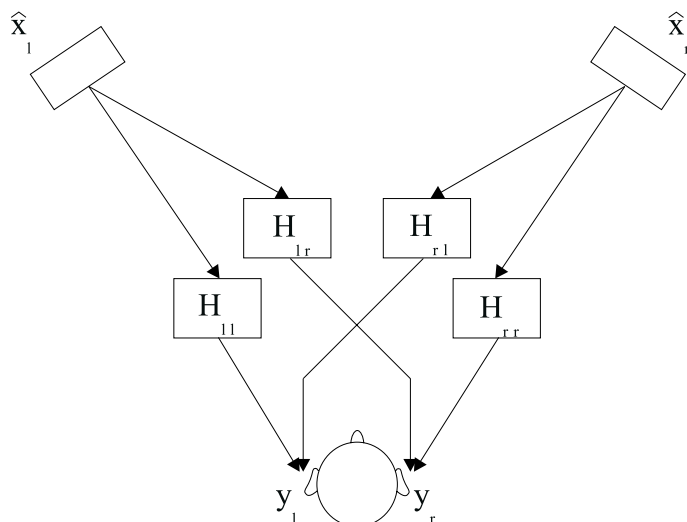
Kaavassa esiintyvän käänteissuodattimen laskenta ei kuitenkaan yleisesti ottaen ole mahdollista nimittäjässä esiintyvien ei-minimivaiheisten komponenttien takia. [Huopaniemi, 1999]

Täysin symmetrisessä tilanteessa (kaiuttimien sijoitus samoin kuin pää ovat symmetrisiä) $H_{lr}(z) = H_{rl}(z)$ ja $H_{ll}(z) = H_{rr}(z)$. Merkitään ipsilateraalista siirtofunktiota $H_i(z) = H_{ll}(z) = H_{rr}(z)$ ja kontralateraalista $H_c(z) = H_{lr}(z) = H_{rl}(z)$. Yhtälö (19) pelkistyy nyt muotoon

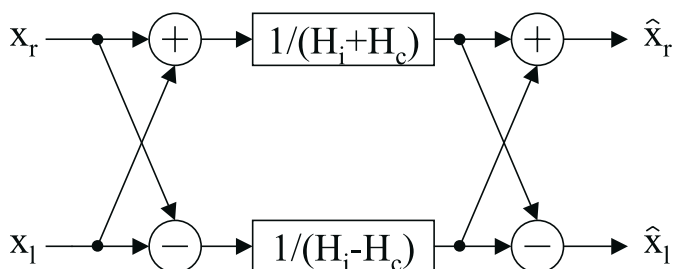
$$\mathbf{G} = \frac{1}{H_i^2(z) - H_c^2(z)} \begin{bmatrix} H_i(z) & -H_c(z) \\ -H_c(z) & H_i(z) \end{bmatrix} \quad (20)$$

Sekoitinmatriisiin

$$\mathbf{D} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (21)$$



Kuva 9: Äänen kulkeutuminen kuulijan korviin stereokaiutintoistossa.



Kuva 10: Ristiinkuulumisenestosuodattimien toteutus sekoitinrakenteella.

avulla voidaan edelleen kirjoittaa

$$\mathbf{G} = \mathbf{D} \begin{bmatrix} \frac{1}{H_i(z) + H_c(z)} & 0 \\ 0 & \frac{1}{H_i(z) - H_c(z)} \end{bmatrix} \mathbf{D} \quad (22)$$

Tällainen suodatin on toteutettavissa kuvan 10 mukaisella sekoitinrakenteella. Normalisointikertoimia $1/\sqrt{2}$ ei ole merkitty kuvaan. [Huopaniemi, 1999]

Kaava (22) sisältää edelleen kaksi käänteissuodatinta, jotka eivät sellaisenaan välttämättä ole toteutettavissa. Suodattimien $H_i(z) + H_c(z)$ ja $H_i(z) - H_c(z)$ on kuitenkin osoitettu olevan yhteisesti minimivaiheisia. Tämä tarkoittaa, että suodattimet voidaan jakaa minimivaiheisiin komponentteihin ja taajuusriippumattomaan viiveeseen, joka on yhtä suuri molemmille suodattimille. Näinollen sekä summa- että erotussuodatin voidaan yhdessä lyhentää minimivaiheiseksi, jolloin

stabiilit käänteissuodattimet voidaan muodostaa. [Cooper ja Bauck, 1989]

Kaiuttomassa tilassa siirtofunktiot $H_i(z)$ ja $H_c(z)$ vastaavat mitattuja HRTF:iä. Todellisessa huonetilassa HRTF:ien käyttö on kuitenkin vain karkea approksimaatio todellisille siirtofunktioille, minkä seurauksena HRTF:ien avulla toteutettu ristiinkuulumisen esto ei koskaan toimi täydellisesti. Eniten binauraalisen kaiutinauralisaation toimintaa haittaavat noin yhden millisekunnin aikana saapuvat huoneheijastukset [Kendall, 1995]. Valitettavasti geneerisiä huonevasteita ei kuitenkaan ole olemassa, joten yleiskäyttöisessä auralisaattorissa joudutaan aina tyytymään vain osittaiseen kumoutumiseen.

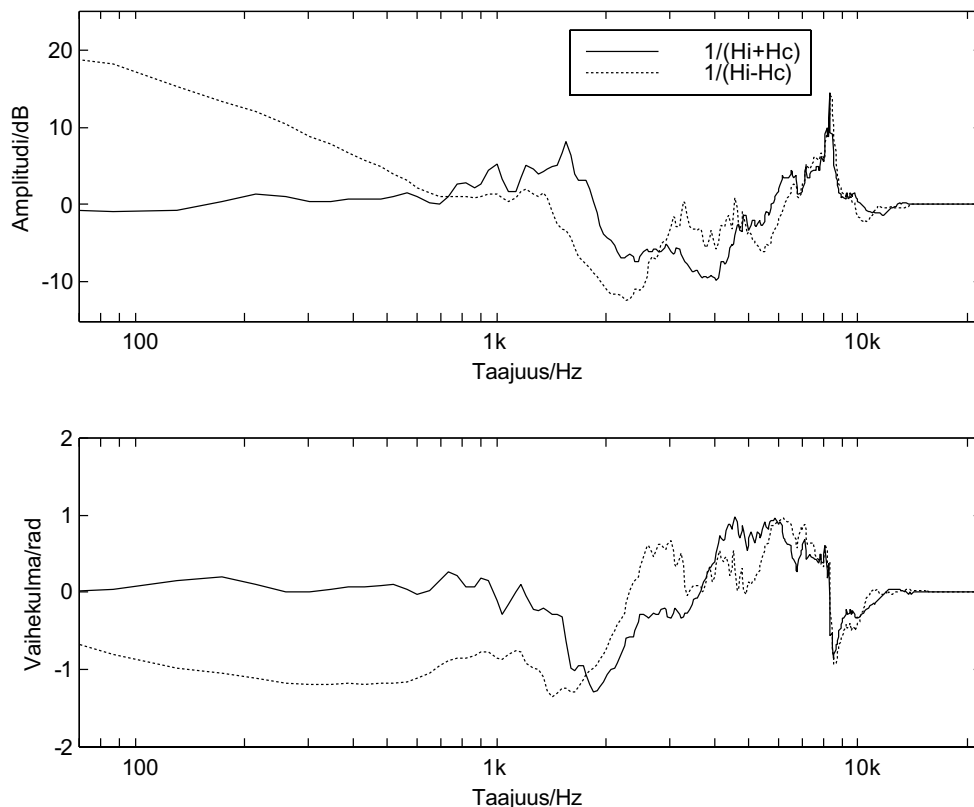
Toinen binauraalisen kaiutinauralisaation ongelma on kuuntelualueen rajoittuminen hyvin pieneen alueeseen (engl. "sweet spot"). Työn aikana tehtyjen vapaa-muotoisten kuuntelukokeiden perusteella tätä aluetta voidaan laajentaa rajoittamalla ristiinkuulumisen kumoamista suurilla taajuuksilla. Samalla myös toiminta huonoissa akustisissa olosuhteissa paranee. Ilmiö voidaan ymmärtää ottamalla huomioon äänen pieni aallonpituus suurilla taajuuksilla.

Ristiinkuulumisen esto perustuu siihen, että kontralateraalisen kaiuttimen signaali summautuu kuulijan korvan kohdalla ipsilateraalista kaiuttimesta sopivasti lähetetyn vastakkaisvaiheisen signaalin kanssa. Kun suurilla taajuuksilla äänen aallonpituus on pieni, hyvin pienetkin muutokset pään sijainnissa tai asennossa riittävät muuttamaan kumoutumisen summautumiseksi. Tämän seurauksena suuret taajuudet korostuvat, ääni muuttuu epäluonnollisen kuuloiseksi ja auralisaation toiminta heikkenee.

Ristiinkuulumisen kumoamista voidaan rajoittaa ikkunoimalla käänteissuodattimia $1/[H_i(z) + H_c(z)]$ ja $1/[H_i(z) - H_c(z)]$ taajuusalueessa niin, että niiden magnitudivaste lähestyy korkeilla taajuuksilla ykköstä. Sijoittamalla kaavaan (22) suodattimien tilalle 1 nähdään, että käänteismatriisi \mathbf{G} pelkistyy tällöin yksikkömatriisiksi

$$\mathbf{G} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (23)$$

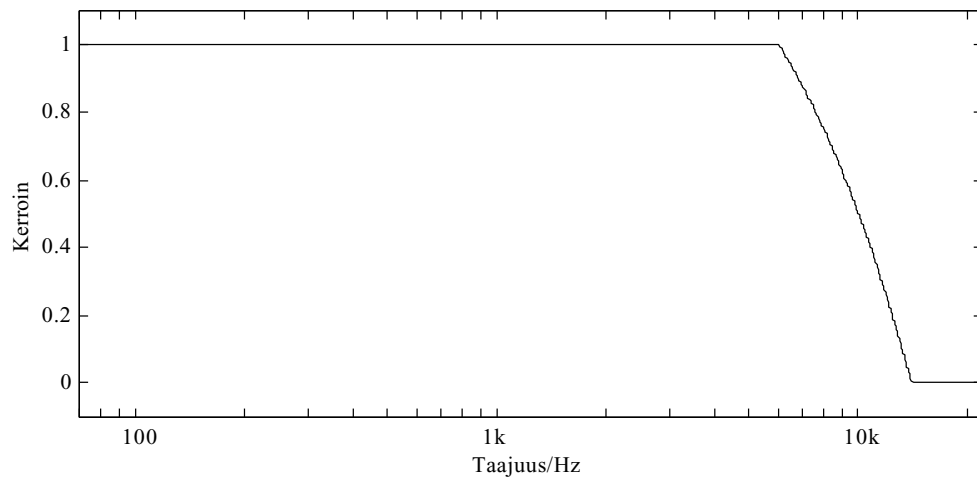
Tämä vastaa binauraalisen signaalin toistoa kaiuttimista sellaisenaan. Yli 10 kHz:n taajuuksilla ja $\pm 30^\circ$ suuntiin horisontaalitasossa sijoitetuilla kaiuttimilla ipsilateraalisen ja kontralateraalisen signaalin ero on minimissään n. 10 dB.



Kuva 11: Käänteissuodattimien $1/[H_i(z) + H_c(z)]$ ja $1/[H_i(z) - H_c(z)]$ magnitudi ja vaihevasteet.

Tätä suurempaa ILD:tä ei suurilla tajuuksilla siis pystyttyä tuottamaan. 10 dB on kuitenkin enemmän kuin huonosti toimivalla ristiinkuulumisenestolla saadaan aikaan.

Kuvassa 11 on esitetty vaakatason $\pm 30^\circ$ HRTF:ien perusteella suunniteltujen ikkunoitujen ristiinkuulumisenestosuodattimien $1/[H_i(z) + H_c(z)]$ ja $1/[H_i(z) - H_c(z)]$ magnitudi- ja vaihevasteet. Ikkunointi on suoritettu logaritmiselle magnitudivasteelle kuvan 12 mukaisella ikkunafunktiolla. Painotuskerroin muuttuu lineaarisesti yhdestä nollaan välillä 6–14 kHz. 6 kHz:n taajuudella äänen aallonpituus on normaalissa huoneenlämmössä n. 5,7 cm ja 14 kHz:llä n. 2,4 cm. Rajataajuudet on valittu subjektiivisten kuunteluvaikutelmien perusteella. Jos kuulijan paikka ei ole kovin tarkka, ikkunointi on syytä aloittaa jo pienemmiltä taajuuksilta. Optimaalisten rajataajuuksien ja ikkunafunktion valinta vaatisi ilmiön tarkempaa tutkimista ja kuuntelukokeiden järjestämistä.



Kuva 12: Ikkuna, jolla kuvan 11 logaritmista magnitudivastetta on painotettu. Painotuskerroin muuttuu välillä 6–14 kHz lineaarisesti taajuuden funktiona yhdestä nollaan.

Ristiinkuulumisen esto tuottaa melko suuria amplitudeja pienillä taajuuksilla. Näitä olisi mahdollista rajoittaa samaan tapaan kuin suurien taajuuksien amplitudeja. Rajoittamisesta ei myöskään olisi haittaa lokalisaation kannalta, koska n. 100 Hz:ä pienemmillä taajuuksilla ääni ei juuri synnytä suuntahavaintoa. Vapaamuotoisten kuuntelukokeiden perusteella pienet taajuudet eivät kuitenkaan korostu liikaa, joten niiden rajoittaminen ei auralisaation kannalta ole välttämätöntä. Sen sijaan äänentoistolaitteiston ja suodatinsuunnittelun kannalta rajoittamisesta saattaisi olla hyötyä.

Binauraalista kaiutintintoistoa on tutkittu myös eri kulmissa kuulijan edessä sijaitsevilla kaiuttimilla. Tulosten mukaan auralisoidun äänen kuuntelualuetta voidaan laajentaa käyttämällä pienempää kulmaa. Lähellä toisiaan olevilla kaiuttimilla pienitaajuuksien äänien ristiinkuulumisen esto vaatii kuitenkin entistä enemmän energiaa, joten kaiuttimia ei voida sijoittaa aivan vierekkäin. 10° kulmalla kaiuttimien välissä on saatu hyviä tuloksia [Kirkeby et al., 1998].

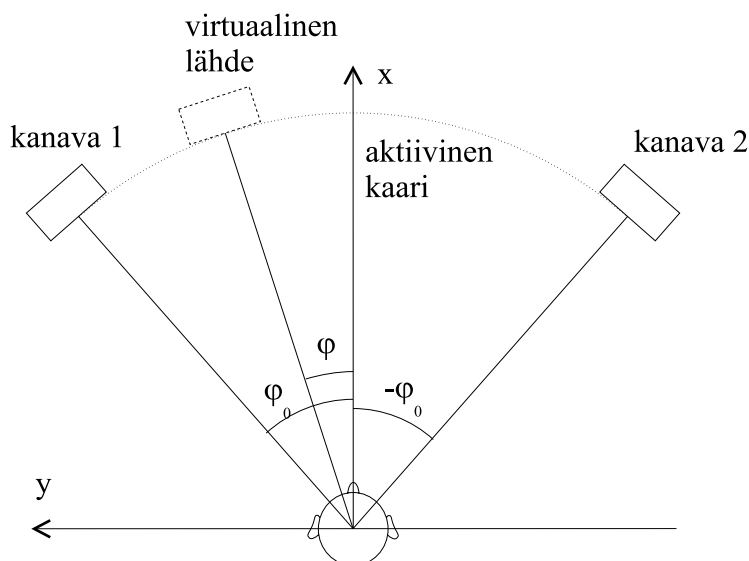
3 Amplitudipanorointi

Amplitudipanorointi on hyvin yleisesti käytössä oleva auralisaatiomenetelmä. Esi-merkiksi lähes kaikki stereofoniset äänitteet hyödyntävät amplitudipanorointia jossain muodossa. Menetelmä toimii niin, että äänisignaali toistetaan samanvaiheisena, mutta eri amplitudilla kahdesta tai useammasta kaiuttimesta. Tämä synnyttää kuulijalle havainnon yhdestä virtuaalisesta äänilähteestä kaiuttimien välisellä alueella. Riippuen kaiuttimien sijoituksesta ja lukumäärästä tätä aluetta voidaan nimittää aktiiviseksi kaareksi (kuva 13) tai kolmioksi (kuva 15).

Virtuaalisen lähteen suuntahavainnon muodostuminen amplitudipanoroinnin perusteella on hyvin monimutkainen prosessi. Suurilla taajuuksilla kuulijan pää vaimentaa melko tehokkaasti kontralateraalisen kaiuttimen signaalia. Normaalissa stereokaiutinkuuntelussa kaiuttimien välinen amplitudiero muuttuu tällöin melko suoraan korvien väliseksi ILD:ksi. Pienillä taajuuksilla kaiuttimien signaalit taas saapuvat molempiin korviin lähes samantasoisina. Signaalien summautumisen vaikutuksesta kaiuttimien tasoero muuttuu kuitenkin korvien väliseksi vaiheeroksi jonka aivot käsittelevät ITD:nä. [Pulkki et al., 1999]

Sekä pienet että suuret taajuudet lokalisoituvat amplitudipanoroinnissa lähes samaan suuntaan. Ongelmia aiheuttaa taajuusalue välillä 1–2 kHz. Tällä alueella virtuaaliset lähteet saattavat paikallistua jopa 10° sivuun samalla amplitudisuhteella panoroiduista suurempi- ja pienempitaajuisista lähteistä, kun kaiuttimet on sijoitettu $\pm 30^\circ$ suuntiin. Pienillä ja suurilla taajuuksilla lokalisaatiosuunnan vaihtelut ovat korkeintaan joitakin asteita. 1–2 kHz:n taajuusalueella lokalisaatio tapahtuu sekä ILD- että ITD-vihjeiden perusteella. Syynä virtuaalisten lähteiden siirtymiseen ovatkin ilmeisesti auralisaatiosuuntaan nähden liian suuret ILD-vihjeet. [Pulkki et al., 1999]

Eri taajuisten signaalikomponenttien hieman eri suuntaisesta lokalisaatiosta johtuen amplitudipanoroinnilla muodostetut virtuaaliset lähteet eivät ole aivan tarkkoja. Tästä huolimatta amplitudipanorointi toimii monissa tapauksissa binauraalista kaiutinauralisaatiota paremmin. Amplitudipanorointi ei ole kovin herkkä kuunteluympäristön akustiikalle ja toimii tyydyttävästi melko suurella kuuntelualueella. Sopivalla kaiuttimien sijoituksella voidaan lisäksi helposti muodostaa virtuaalisia lähteitä kuulijan taakse ja sivuille, eli suuntiin joissa binauraalinen kaiutinauralisaatio toimii heikosti.



Kuva 13: Tyypillinen kaksikanavainen kuuntelujärjestelmä. Kaiuttimien 1 ja 2 synnyttämän virtuaalisen lähteen kulma φ määräytyy sinilain (24) mukaan.

3.1 Horisontaalinen amplitudipanorointi

Kahden kaiuttimen välisessä amplitudipanoroinnissa virtuaalisen lähteen lokalisatiokulmaa φ kaiuttimien vahvistuskertoimien g_1 ja g_2 funktiona voidaan approksimoida sinilailla [Bauer, 1961]

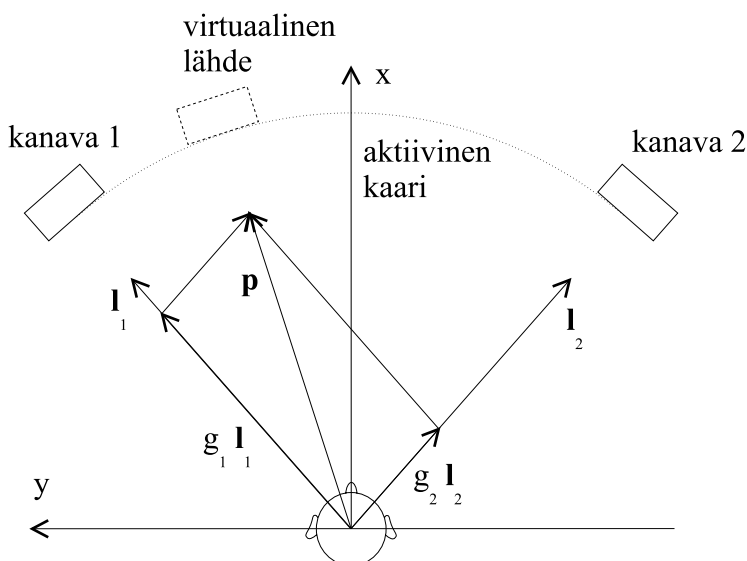
$$\frac{\sin \varphi}{\sin \varphi_0} = \frac{g_1 - g_2}{g_1 + g_2} \quad (24)$$

missä $0^\circ < \varphi_0 < 90^\circ$, $-\varphi_0 \leq \varphi \leq \varphi_0$ ja $g_1, g_2 \geq 0$. Kulmat on määritelty kuvan 13 mukaisesti siten, että suunta $\varphi = 0^\circ$ on suoraan kuulijan edessä ja kaiuttimet on sijoitettu symmetrisesti $\pm\varphi_0$ asteen kulmiin. Sinilaki toimii tarkimmin pienitaajuisilla signaaleilla kuulijan katsoessa suoraan eteenpäin [Pulkki et al., 1999]. Jos kuulija sitävastoin seuraa katseellaan havaitsemaansa virtuaalisen lähteen suuntaa, tangenttilaki [Bernfeld, 1973]

$$\frac{\tan \varphi}{\tan \varphi_0} = \frac{g_1 - g_2}{g_1 + g_2} \quad (25)$$

antaa oikeamman tuloksen. Kuuntelukokeet ovat kuitenkin osoittaneet, että sini- ja tangenttilain ero voidaan usein jättää huomioimatta. [Pulkki, 1997]

Kaavat (24) ja (25) antavat ainoastaan vahvistuskertoimien suhteen halutulle au-



Kuva 14: Kaksikanavainen amplitudipanorointi vektoriesityksenä.

ralisaatiosuunnalle. Eri suuntaisten virtuaalisten lähteiden voimakkuus voidaan pitää vakiona normalisoimalla vahvistuskertoimet kaavalla

$$g_1^2 + g_2^2 = C \quad (26)$$

missä $C > 0$ on kokonaisäänitehon määräävä vakio. C vaikuttaa myös jonkin verran virtuaalisen lähteen etäisyshavaintoon niin, että voimakkaampi ääni tuntuu tulevan lähempää. Koska tarkka etäisyssimulointi on kuitenkin mahdotonta simuloimatta samalla jotain tilaa, virtuaalisen lähteen ajatellaan yleensä syntävän kaiuttimia yhdistävälle aktiiviselle kaarelle.

3.2 VBAP

Vector Base Amplitude Panning (VBAP) [Pulkki, 1997] on vektorialgebran muotoon formuloitu amplitudipanorointialgoritmi. Kaksiulotteisessa tapauksessa sisääntulosignaalin panorointi suoritetaan kahden kaiuttimen välillä seuraavasti. Olkoon $\mathbf{l}_1 = [l_{11} \ l_{12}]^T$ ja $\mathbf{l}_2 = [l_{21} \ l_{22}]^T$ kaiuttimia kohti osoittavat yksikkövektorit (kuva 14), joissa yläviite T tarkoittaa matriisitransponointia. l_{11} ja l_{12} ovat kaiuttimen 1 suuntaisen yksikkövektorin x- ja y-komponentit ja l_{21} ja l_{22} vastaavasti kaiuttimen 2 suuntaisen yksikkövektorin x- ja y-komponentit. Vektorit \mathbf{l}_1 ja \mathbf{l}_2 muodostavat nyt kannan, jonka avulla kaiuttimien väliin sijoittuvaa

virtuaalista lähdettä kohti osoittava yksikkövektori $\mathbf{p} = [p_1 \ p_2]^T$ voidaan lausua

$$\mathbf{p} = g_1 \mathbf{l}_1 + g_2 \mathbf{l}_2, \quad g_1, g_2 \in [0, 1] \quad (27)$$

Sama voidaan esittää matriisimuodossa

$$\mathbf{p} = \mathbf{L} \mathbf{g} \quad (28)$$

missä $\mathbf{L} = [\mathbf{l}_1 \ \mathbf{l}_2]$ ja $\mathbf{g} = [g_1 \ g_2]^T$. Tästä voidaan ratkaista \mathbf{g} halutun suuntaiselle virtuaaliselle lähteelle

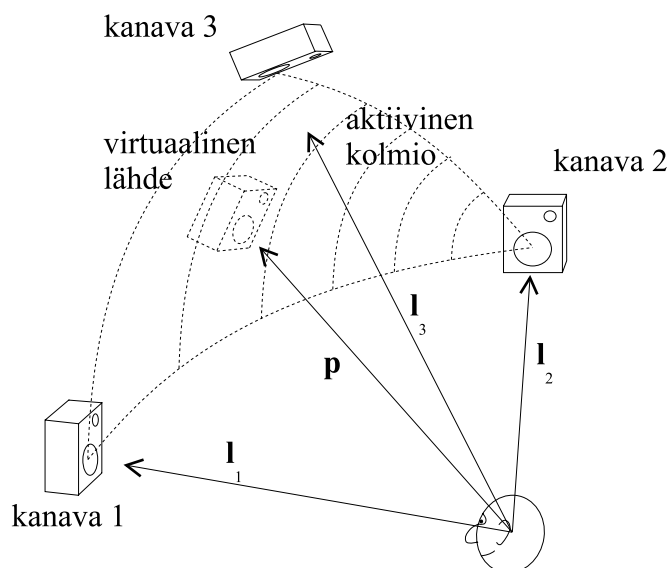
$$\mathbf{g} = \mathbf{L}^{-1} \mathbf{p} = \begin{bmatrix} l_{11} & l_{21} \\ l_{12} & l_{22} \end{bmatrix}^{-1} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \quad (29)$$

missä käänteismatriisi \mathbf{L}^{-1} on olemassa sillä edellytyksellä, että $\varphi_0 \neq 0^\circ$ ja $\varphi_0 \neq 90^\circ$. Kaavalla (29) saatujen vahvistuskertoimien g_1 ja g_2 voidaan osoittaa toteutuvan tangentialilaki [Pulkki, 1997]. Jotta eri suuntiin sijoitetut virtuaaliset lähteet saadaan vielä keskenään yhtä voimakkaiksi, vahvistuskertoimet pitää normalisoida kaavalla

$$\mathbf{g}_{\text{scaled}} = \frac{\sqrt{C} \mathbf{g}}{\sqrt{g_1^2 + g_2^2}} \quad (30)$$

Käänteismatriisin \mathbf{L}^{-1} olemassaolon ehdot vastaavat vaatimusta, että vektorit \mathbf{l}_1 ja \mathbf{l}_2 ovat lineaarisesti riippumattomia. Tapaus $\varphi_0 = 0^\circ$ vastaa tilannetta, jossa molemmat kaiuttimet ovat samassa suunnassa ja virtuaalisen lähteen synnyttäminen kaiuttimien väliin on siis mahdotonta. Kun $\varphi_0 = 90^\circ$, kaiuttimet ovat puolestaan suoraan kuulijan sivuilla ja tilanne vastaa enemmänkin äänen kuuntelemista kuulokkeilla. Amplitudipanoroidun äänen kuulokekuuntelussa puhutaan yleensä lokalisaation sijaan lateralisaatiosta, koska äänilähde tuntuu paikallistuvan pään sisälle [Grantham, 1995]. Kumpikaan tapaus ei ole auralisaation kannalta kiinnostava vaihtoehto.

Vaikka VBAP rajoittaa mahdolliset auralisaatiokulmat kaiuttimien välille, auralisaatioaluetta voidaan laajentaa yksinkertaisesti sijoittamalla useampia kaiuttimia kuuntelupaikan ympärille. Panorointi suoritetaan edelleen kahden kaiuttimen välillä, mutta kaiutinpari kullekin auralisoitavalle äänelle valitaan erikseen. Monikanavapanorointia käsitellään tarkemmin kappaleessa 3.4



Kuva 15: Kolmiulotteinen amplitudipanorointi. Virtuaalinen lähde voidaan kolmen kaiuttimen avulla sijoittaa mihin tahansa aktiivisen kolmion sisällä.

3.3 Kolmiulotteinen VBAP

Tyypillistä kahdesta vaakatasossa olevasta kaiuttimesta koostuvaa stereofonista kuuntelujärjestelmää voidaan laajentaa lisäämällä siihen kolmas kaiutin eri korkeudelle. Tällainen järjestelmä, jossa kuuntelija ja kaikki kaiuttimet eivät sijaitse samassa tasossa, mahdollistaa kolmiulotteisen amplitudipanoroinnin. Kuvan 15 mukaisesti virtuaalinen lähde voidaan nyt sijoittaa kaiuttimien rajaaman aktiivisen kolmion sisään.

VBAP-algoritmi voidaan yleistää kolmiulotteiseksi seuraavasti [Pulkki, 1997]. Olkoon $\mathbf{l}_1 = [l_{11} \ l_{12} \ l_{13}]^T$ kaiuttimen 1 suuntaan ja \mathbf{l}_2 ja \mathbf{l}_3 vastaavasti kaiuttimien 2 ja 3 suuntiin osoittavat yksikkövektorit. Kuten kaksiulotteisessa tapauksessa vektorit \mathbf{l}_1 , \mathbf{l}_2 ja \mathbf{l}_3 muodostavat kannan, jonka avulla virtuaalisen lähteen suunta $\mathbf{p} = [p_1 \ p_2 \ p_3]^T$ voidaan lausua

$$\mathbf{p} = \mathbf{L}\mathbf{g} \quad (31)$$

missä $\mathbf{g} = [g_1 \ g_2 \ g_3]^T$. g_1 , g_2 ja g_3 ovat vastaavannumeroisten kanavien vahvis-

tuskertoimet ja $\mathbf{L} = [\mathbf{l}_1 \ \mathbf{l}_2 \ \mathbf{l}_3]$. Tästä voidaan ratkaista \mathbf{g}

$$\mathbf{g} = \mathbf{L}^{-1}\mathbf{p} = \begin{bmatrix} l_{11} & l_{21} & l_{31} \\ l_{12} & l_{22} & l_{32} \\ l_{13} & l_{23} & l_{33} \end{bmatrix}^{-1} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \quad (32)$$

sillä edellytyksellä, että kantavektorit ovat lineaarisesti riippumattomia, jolloin käänteismatriisi \mathbf{L}^{-1} on olemassa. Kaavalla (32) lasketut vahvistuskertoimet voidaan edelleen normalisoida tuottamaan yhtä voimakas lähde missä tahansa suunnassa kaavalla

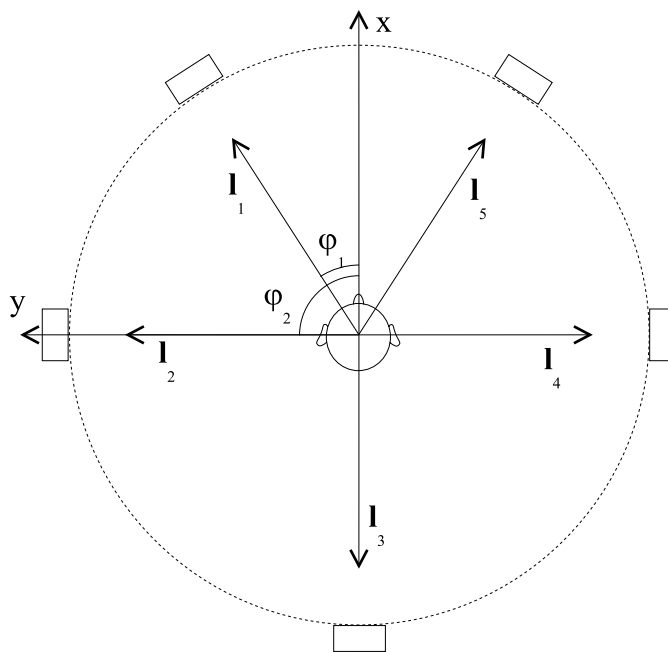
$$\mathbf{g}_{\text{scaled}} = \frac{\sqrt{C}\mathbf{g}}{\sqrt{g_1^2 + g_2^2 + g_3^2}} \quad (33)$$

Kuten kaksiulotteisessa tapauksessa, systeemiä voidaan laajentaa lisäämällä siihen useampia kaiuttimia. Panorointi suoritetaan aina kolmen kaiuttimen välillä, mutta optimaalisten kaiuttimien valinta kolmioihin on huomattavasti monimutkaisempaa kuin kaksiulotteisessa tapauksessa.

3.4 Monikanavaiset äänentoistojärjestelmät

Jos äänentoistojärjestelmä sisältää enemmän kuin amplitudipanoroinnin asettaman minimimäärän kaiuttimia, joudutaan valitsemaan mitä kaiuttimia kunkin sisääntulosignaalin panorointiin käytetään. Minimimäärät ovat kaksiulotteisessa panoroinnissa kaksi ja kolmiulotteisessa kolme kaiutinta. Terävimmät, mahdollisimman tarkasti yhteen pisteeseen lokalisoituvat virtuaaliset lähteet saadaan aikaan käyttämällä kaikilla auralisaatiosuunnilla mahdollisimman lähellä toisiaan sijaitsevia kaiuttimia. Tällöin panoroinnin eri taajuuksilla synnyttämät toisistaan poikkeavat ILD- ja ITD-vihjeet ovat vähiten ristiriidassa keskenään. Lähteiden terävyys riippuu myös panorointisuunnasta kaiuttimien välisellä alueella ja on terävimmillään kaiuttimien kohdalla [Pulkki, 1999b].

Kaksiulotteisessa tapauksessa amplitudipanorointi suoritetaan aina lähimpänä toisiaan olevien samassa tasossa olevien kaiuttimien välillä. Kaiutinparin valinta on helppoa: Järjestetään kaiuttimet vaakakulman φ mukaiseen järjestykseen ja valitaan auralisaatiosuuntaa lähimpänä olevat kaiuttimet sen molemmin puolin. Mahdolliset parit muodostuvat vierekkäisistä kaiuttimista, jolloin kukin kaiutin

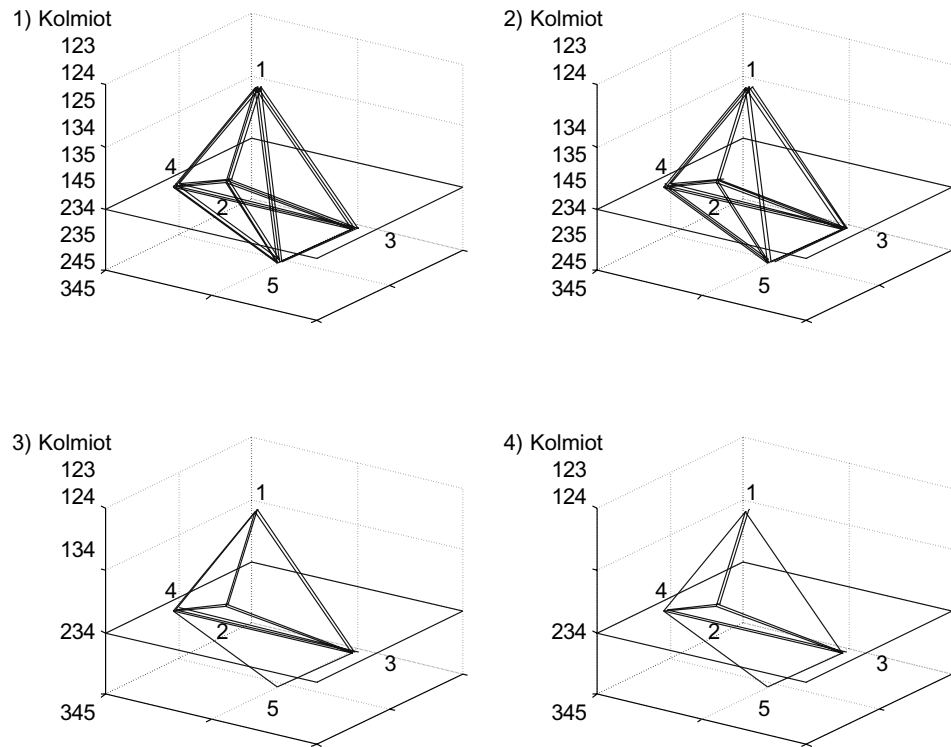


Kuva 16: Esimerkki monikanavaisesta 2-ulotteisesta amplitudipanoroinnista. Parit (1, 2), (2, 3), (3, 4), (4, 5) ja (5, 1) kattavat kaikki mahdolliset auralisaatiokulmat.

voi kuulua enintään kahteen pariin. Kaiuttimien välinen kulma ei kuitenkaan saa ylittää 180° , jotta niitä voitaisiin pitää parina. Kuva 16 esittää erään mahdollisen kaksiulotteisen kokoonpanon.

Vierekkäisten kaiuttimien valinta tuottaa pareja, joiden aktiiviset kaaret eivät koskaan ole päällekkäisiä. Näinollen mihin tahansa suuntaan auralisoitava virtuaalinen lähde voidaan tuottaa korkeintaan yhden parin avulla. Jos käänteismatriisit \mathbf{L}_n kaikille mahdollisille kaiutinpareille lasketaan valmiiksi, parin valinta halutunsuuntaiselle auralisaatiolle voidaan suorittaa laskemalla vahvistuskertoimet g_1 ja g_2 kaikille pareille ja valitsemalla se pari, jolla molemmat kertoimet ovat positiivisia. Mikäli tällaista paria ei löydy, kaiuttimien sijoitus ei mahdollista auralisaatiota haluttuun suuntaan. [Pulkki, 1997]

Optimaalisten kaiutinkolmioiden valinta kolmiulotteisessa tapauksessa on huomattavasti vaikeampi tehtävä. Kaiuttimien sijoituksesta riippuen monet auralisaatiosuunnat voidaan yleensä tuottaa useilla eri kolmioilla, jotka voivat olla eri muotoisia ja kattaa eri kokoisia osia koko auralisaatioalueesta. Alueen jako aina terävimmän mahdollisen lähteen synnyttäviin ei-päällekkäisiin kolmioihin voidaan

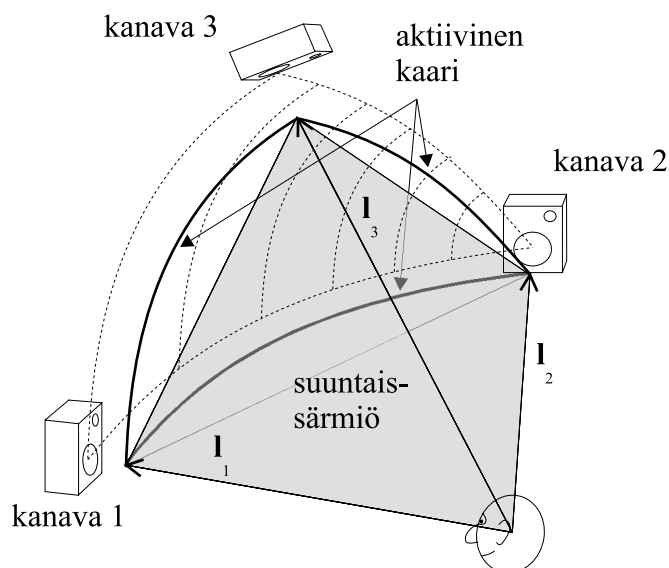


Kuva 17: Optimaalisten kaiutinkolmioiden valinta-algoritmin eri vaiheet viiden kaiuttimen järjestelmässä. Kolmioiden sivut on kuvattu suorilla viivoilla pallonpinnan kaarien sijaan. Kaiuttimet 3 ja 4 ovat kuulijan pään tasolla. Kaiuttimet 1, 2 ja 5 ovat suoraan edessä niin, että kaiutin 2 on hieman kuulijan päätä ylempanä. 1) Kaikki kolmiot. 2) Kaiuttimet 1, 2 ja 5 ovat samalla pystyviivalla. 3) Aktiivinen kaari 3–4 on lyhyempi kuin 1–5 ja 2–5. 4) Kaiutin 2 on kolmion 134 sisällä.

suorittaa seuraavan nelivaiheisen algoritmin avulla [Pulkki ja Lokki, 1998]:

1. Muodostetaan kaikki mahdolliset kolmen kaiuttimen kombinaatiot.
2. Poistetaan kaiutinkolmiot, joiden aktiivinen kolmio on liian pieni suhteessa sivujen yhteispituuteen.
3. Etsitään kolmioparit, joissa kaiuttimien väliset aktiiviset kaaret leikkaavat toisensa ja poistetaan kolmio, jonka aktiivinen kaari on pidempi.
4. Poistetaan kolmiot, joiden sisällä on kaiutin.

Algoritmin toimintaa on havainnollistettu esimerkillä kuvassa 17. Liian kapeiden



Kuva 18: Kaiutinvektorien ja kuulijan muodostama suuntaissärmiö, jonka tilavuuden suhdetta aktiivisten kaarien pituuteen käytetään kriteerinä liian kapeiden kolmioiden hylkäämiseen.

kolmioiden eliminointi vaiheessa 2 voidaan suorittaa seuraavasti: Olkoon \mathbf{l}_i , \mathbf{l}_j ja \mathbf{l}_k kuuntelupisteestä kohti kaiuttimia i , j ja k osoittavat yksikkövektorit. Lasketaan näiden muodostaman suuntaissärmiön tilavuus (kuva 18)

$$\frac{|\mathbf{l}_i \times \mathbf{l}_j \cdot \mathbf{l}_k|}{6} \quad (34)$$

ja jaetaan se kaiutinparien aktiivisten kaarien yhteispituudella

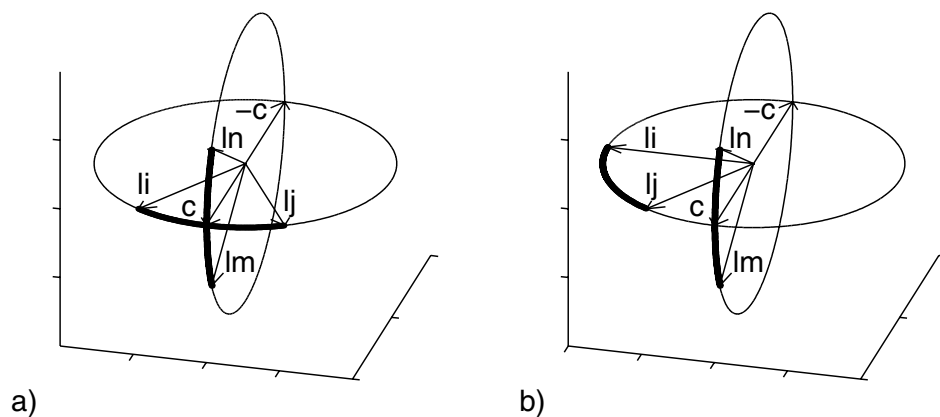
$$|\angle(\mathbf{l}_i, \mathbf{l}_j)| + |\angle(\mathbf{l}_j, \mathbf{l}_k)| + |\angle(\mathbf{l}_i, \mathbf{l}_k)| \quad (35)$$

missä \angle tarkoittaa pienintä vektorien välistä kulmaa. Kaiutinkolmioiden hylkäys ehdoksi saadaan nyt

$$\frac{|\mathbf{l}_i \times \mathbf{l}_j \cdot \mathbf{l}_k|}{|\angle(\mathbf{l}_i, \mathbf{l}_j)| + |\angle(\mathbf{l}_j, \mathbf{l}_k)| + |\angle(\mathbf{l}_i, \mathbf{l}_k)|} \leq k \quad (36)$$

missä $k \approx 0.01$ on vakio.

Leikkaavien aktiivisten kaarien etsintä tapahtuu myös vektorialgebran avulla. Kaiutinparien (i, j) ja (m, n) kohdalla leikkaavuus voidaan selvittää seuraavasti: Määritellään kaiuttimia kohti osoittavat yksikkövektorit \mathbf{l}_i , \mathbf{l}_j , \mathbf{l}_m ja \mathbf{l}_n . Kumpikin



Kuva 19: a) Vektoriparien $\{\mathbf{l}_i, \mathbf{l}_j\}$ ja $\{\mathbf{l}_n, \mathbf{l}_m\}$ aktiiviset kaaret leikkaavat toisensa, jolloin \mathbf{c} osoittaa näihin molempiin. b) Aktiiviset kaaret eivät leikkaa toisiaan, jolloin kumpikaan \mathbf{c} tai $-\mathbf{c}$ ei osoita molempiin kaariin.

vektoripari $\{\mathbf{l}_i, \mathbf{l}_j\}$ ja $\{\mathbf{l}_n, \mathbf{l}_m\}$ virittää nyt tason. Näiden tasojen leikkauspisteet $\pm\mathbf{c}_c$ etäisyydellä 1 origosta (kuva 19) saadaan kaavasta

$$\mathbf{c}_c = ((\mathbf{l}_i \times \mathbf{l}_j) \times (\mathbf{l}_m \times \mathbf{l}_n)) \quad (37)$$

Nämä leikkauspisteet ovat ainoat pisteet, joissa kaiutinparia $\{\mathbf{l}_i, \mathbf{l}_j\}$ yhdistävä kaari voi leikata kaiuttimien $\{\mathbf{l}_n, \mathbf{l}_m\}$ välisen kaaren. Näinollen, jos, ja vain jos jompikumpi leikkauspisteistä $\pm\mathbf{c}_c$ sijaitsee molempien kaiutinparien kaarilla, kaaret risteävät. Tämä ehto on yhtäpitävä sen kanssa, että

$$\angle(\mathbf{c}, \mathbf{l}_i) + \angle(\mathbf{c}, \mathbf{l}_j) = \angle(\mathbf{l}_i, \mathbf{l}_j) \text{ ja } \angle(\mathbf{c}, \mathbf{l}_m) + \angle(\mathbf{c}, \mathbf{l}_n) = \angle(\mathbf{l}_m, \mathbf{l}_n) \quad (38)$$

pätee jommallekummalle $\mathbf{c} = \mathbf{c}_c$ tai $\mathbf{c} = -\mathbf{c}_c$.

Eliminoimalla leikkaavat aktiiviset kaaret saadaan poistettua suurin osa päällekkäisyyksistä eri kaiutinkolmioiden auralisaatioalueissa. Algoritmin viimeinen vaihe eliminoi loput päällekkäisyydet poistamalla kolmiot, joiden sisällä on kaiutin. Tämä tapahtuu käymällä läpi kaikki jäljelläolevat kolmiot ja laskemalla kaikille kolmioon kuulumattomille kaiuttimille niiden suuntaisen auralisaation vahvistuskertoimet g_1 , g_2 ja g_3 . Jos jollain kaiuttimella kaikki vahvistuskertoimet ovat positiivisia, se sijaitsee kolmion sisällä, ja kyseinen kolmio pitää eliminoida.

3.5 Amplitudipanoroinnin erityispiirteitä

Auralisaatio kuulijan sivuille muodostaa erään erikoistapauksen. Jos amplitudipanorointi tapahtuu kuulijan etu- ja takapuolella olevien kaiuttimien välillä, virtuaalisen lähteen suunta on hyvin herkkä kuulijan pään liikkeille ja pienille virheille vahvistuskertoimissa. [Theile ja Plenge, 1977] Tämän aiheuttamat ongelmat voidaan kuitenkin kiertää sijoittamalla kaiutin suoraan kuulijan molemmille sivuille. Kaikkien kaiuttimien pitää myös olla melko tarkasti yhtä kaukana kuulijasta, jotta amplitudipanorointi toimii optimaalisesti. VBAP-algoritmi ei ota huomioon sivujen erikoisasemaa, eikä kaiuttimien etäisyyksien vaikutusta lopputulokseen.

Amplitudipanoroinnissa virtuaalisten lähteiden terävyys vaihtelee auralisaatio-suunnan ja kaiuttimien sijoituksen mukaan. Lähteiden terävyyttä voidaan parantaa kompensoimalla eri taajuuksien paikallistumista eri suuntiin taajuusriippuvien vahvistuskertoimien avulla [Pulkki et al., 1999]. Liikkuvien lähteiden tapauksessa terävyyden vaihtelu saattaa edelleen olla häiritsevää. Terävyyttä voidaan tällöin tasoittaa huonontamalla tarkkuutta terävimmissä kohdissa kaiuttimien lähellä. Tämä tapahtuu suorittamalla panorointi niin, että kaiuttimien lähellä käytetään useampaa kaiutinta kuin olisi välttämätöntä [Pulkki, 1999b].

4 Laitteisto

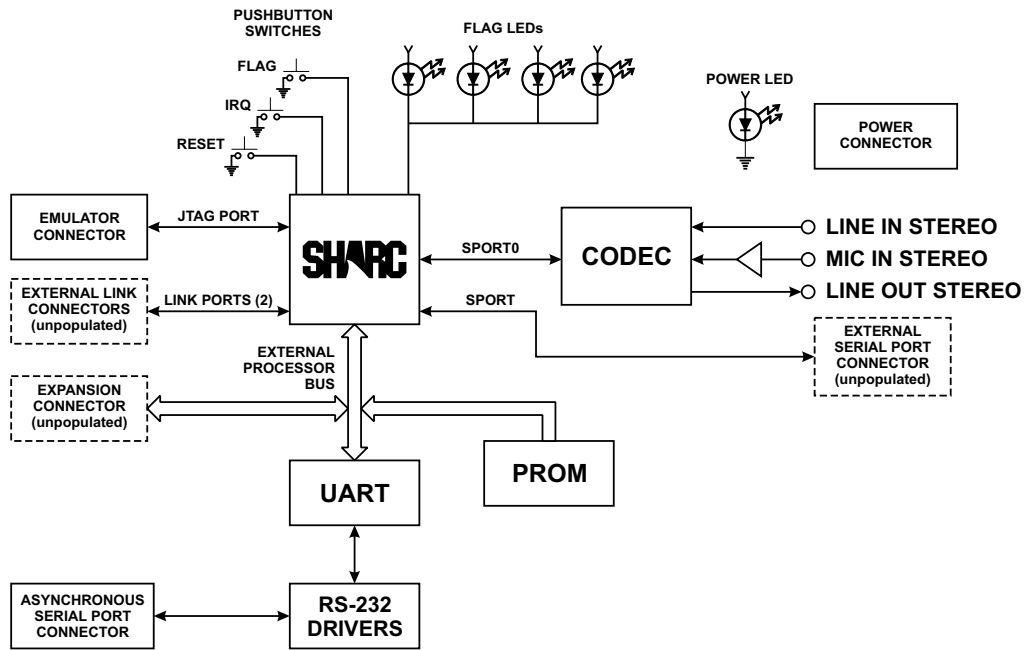
Auralisaattorin toteutukseen on valittu Analog Devicesin [Analog Devices, 2000a] ADSP-21061 SHARC -signaaliprosessori ja EZ-KIT Lite -evaluaatiokortti. Alunperin kaksikanavainen kortti on laajennettu kahdeksankanavaiseksi. Korttiin on lisäksi liitetty nestekidenäyttö ja painokytkimiä auralisaation ohjausta varten. Koko laitteisto on koteloitu kahden yksikön räkkikoteloon.

EZ-KIT Lite -signaaliprosessorikortin [Analog Devices, 1997a] arkkitehtuuri on esitetty kuvassa 20. Kortti sisältää prosessorin lisäksi kaksi 16-bittistä AD- ja DA-muunninta AD1847-piirille integroituna. Ulkoista datasiirtoa varten kortilla on RS-232-rajapinta sekä prosessorin ulkoisiin väyliin yhdistetyt liittimet. Lisäksi kortti sisältää 32-nastaisen DIP-mikropiirikannan, johon voidaan liittää PROM-tai Flash-muistia prosessorin käynnistystä varten. Prosessorin maksimikellotaajuus on 50 MHz, mutta EZ-KIT Lite käyttää sitä 40 MHz:llä.

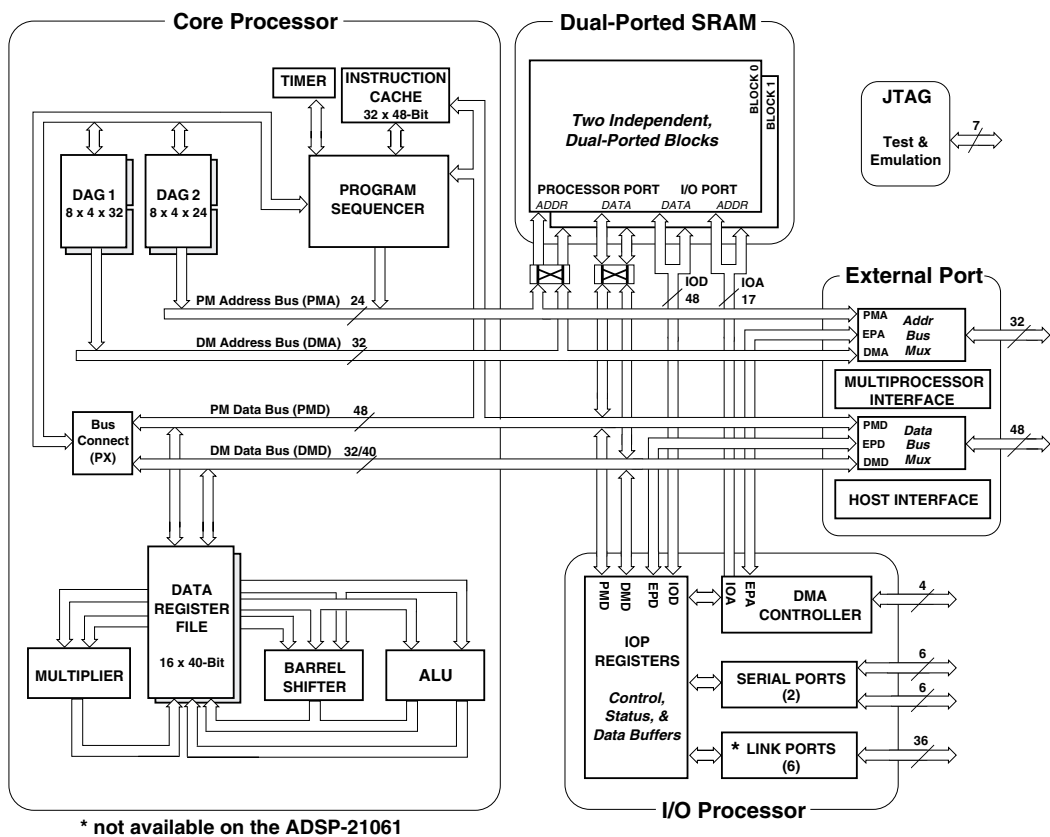
4.1 ADSP-21061 SHARC

ADSP-21061 SHARC (Super Harvard Architecture Computer) on 32-bittinen liukulukusignaaliprosessori. Tästä eteenpäin käytetään nimitystä SHARC viittaamaan nimenomaan ADSP-21061:een. SHARC on osa useita liukulukusignaaliprosessoreja sisältävää ADSP-21000-perhettä. Prosessorit perustuvat nimensä mukaisesti modifioituun Harvard-arkkitehtuuriin [Lapsley et al., 1995, s. 51]. SHARC sisältää kaksi toisistaan riippumatonta kaksiporttista 512 kilobitin SRAM-muistilohkoa sekä neljä dataväylää. Toinen muistilohko (DM) on ainoastaan datalle ja toista (PM) voidaan käyttää sekä datalle että ohjelmakoodille. SHARCin arkkitehtuuri on esitetty kuvassa 21. [Analog Devices, 1997b]

SHARCin laskenta tapahtuu erillisten kertojan, ALU:n (Arithmetic Logic Unit) ja siirtimen (engl. shifter) avulla. Laskentayksiköt osaavat käsitellä liukulukuja 40 ja kokonaislukuja 32 bitin tarkkuudella. Kukin yksikkö pystyy kellojakson aikana suorittamaan yhden operaation ja kirjoittamaan tuloksen haluttuun rekisteriin. Hyödyntämällä rinnakaisuutta on siis mahdollista saada valmiiksi yksi signaalinkäsittelyssä hyvin tärkeä MAC-operaatio (Multiply Accumulate) joka kellojaksolla.



Kuva 20: EZ-KIT Lite -signaaliprosessorikortin lohkokaavio.



Kuva 21: ADSP-2106x SHARC -signaaliprosessorin lohkokaavio.

SHARC sisältää 16 yleiskäyttöistä 40-bittistä rekisteriä. Kaikki laskentayksiköille tarkoitettu data pitää kierrättää näiden kautta. Prosessorin osoitteenmuodostus tapahtuu kahden DAG:n (Data Address Generator) avulla. Toinen näistä generoi osoitteita DM- ja toinen PM-muistilohkoon. SHARC tukee laitteistotasolla rengaspuskurien toteutusta sekä osoitteen käännettyä bittijärjestystä. PM-data- ja -osoiteväylät on jaettu DAG2:n ja ohjelmasekvensserin kesken, mikä saattaa joissakin tapauksissa hidastaa PM-muistilohkon käyttöä. Ohjelmakoodin kätkömuistin ansiosta on kuitenkin usein mahdollista käyttää sekä PM- että DM-väyliä datan siirtoon hidastamatta prosessorin toimintaa.

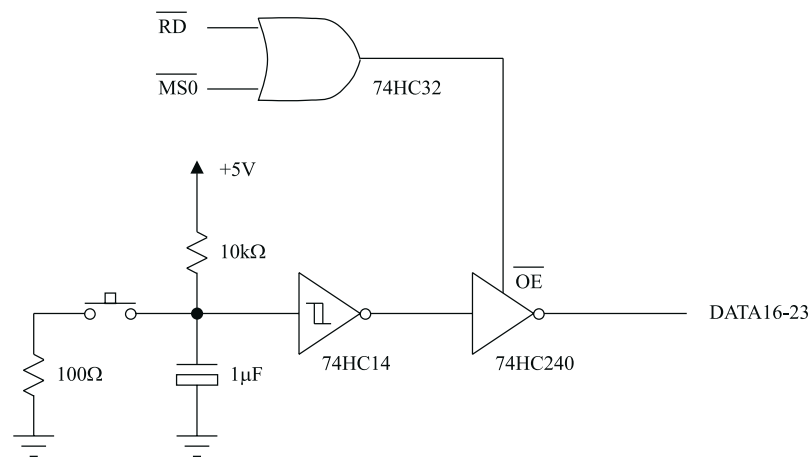
SHARCin sisäinen muisti on yhdistetty omilla data- ja osoiteväylillään prosessorin I/O-yksikköön. Tämän ansiosta I/O-yksikkö voi muusta prosessorista riippumatta siirtää dataa muistin ja ulkoisten väylien välillä. EZ-KIT Lite -kortin AD- ja DA-muuntimet on yhdistetty prosessoriin sen 40 Mbit/s nopeudella toimivan sarjaportin kautta. DMA (Direct Memory Access) huolehtii datan siirrosta prosessorin muistiin ja ilmoittaa uusista näytteistä keskeytyskutsun avulla.

Muita hyödyllisiä ominaisuuksia ovat ohjelmoitava ajastin ja ns. kontekstin vaihtomahdollisuus. Ajastin voidaan ohjelmoida suorittamaan keskeytyskutsu säännöllisin väliajoin. Kontekstin vaihto toimii niin, että kaikki sharcin rekisterit voidaan kerralla tai muutamassa osassa vaihtaa toisiin. Prosessori sisältää siis kahdet rekisterit, joista vain toiset ovat käytössä. Käyttämättä olevat rekisterit säilyttävät kuitenkin arvonsa.

Prosessorin käskyt koostuvat 48 bitin käskysanoista. Pitkät käskysanat ja yleiskäyttöiset rekisterit mahdollistavat melko ortogonaalisen käskykannan ja monet käskyistä voidaan toteuttaa ehdollisesti. SHARCin ALU osaa normaalin toiminnan lisäksi laskea karkeat arviot liukuluvun neliöjuurelle ja käänteisluvulle. Assembler-kieli käyttää C-tyylistä syntaksia. Esimerkiksi kahden liukuluvun summa lasketaan käskyllä $F0 = F1 + F2$; ja tulo käskyllä $F0 = F1 * F2$;

4.2 Monikanavalaajennus

EZ-KIT Lite -kortin tarjoamat kaksi sisään- ja ulostulokanavaa analogiselle äänelle eivät riitä suunniteltavalle auralisaattorille. Tämän takia työssä on käytetty Kari Haution suunnittelemaa ja toteuttamaa monikanavalaajennusta. Signaali-



Kuva 22: Käyttöliittymälajennuksen painonappien toteutus.

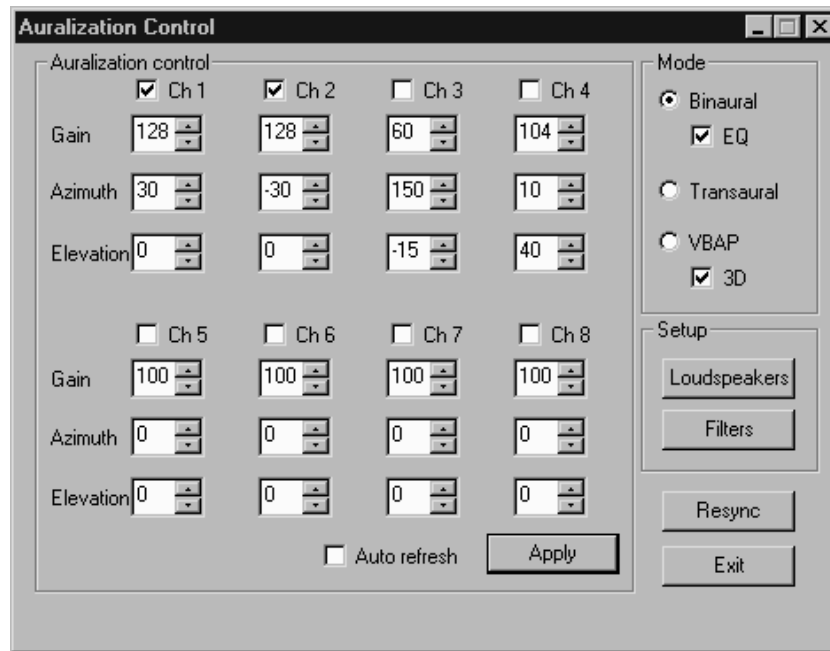
prosessorikorttiin yhdistettävä laite on toimintaperiaatteeltaan melko yksinkertainen. Laite sisältää kolme AD1847-koodekkia lisää. Näiden kaikkien kytkentä vastaa pääpiirteissään EZ-KIT Liten toteutusta. Koodekit on ketjutettu samaan sarjaporttiin niin, että ne lähettävät tai vastaanottavat dataa vuorotellen. Lopputuloksena saadaan käyttöön kahdeksan sisään- ja ulostulokanavaa.

4.3 Käyttöliittymä

Auralisaation ohjausta varten prosessorikorttiin on liitetty 16 merkin nestekidenäyttömoduli ja kahdeksan painokytkintä. Laajennus on yhdistetty SHARCin ulkoiseen muistiväylään niin, että kytkimien tila voidaan selvittää lukemalla dataa mistä tahansa ulkoisen muistin osoitteesta bankin 0 alueelta. Näytön ohjaus puolestaan tapahtuu kirjoittamalla samaan osoitealueeseen. Tarvittaessa laajennus olisi voitu sijoittaa tarkasti yksittäiseen ulkoisen muistin osoitteeseen, mutta tähän ei ollut tarvetta, koska 0-bankissa ei sijaitse mitään muuta.

Painokytkimet on toteutettu kuvan 22 mukaisella kytkennällä. Tieto kunkin kytkimen tilasta menee Schmitt-triggerillä varustetun invertterin 74HC14 sekä kolmitilaisella ulostulolla varustetun invertterin 74HC240:n läpi. Dataväylään liitetty ulostulo kytkeytyy päälle ainoastaan prosessorin lukiessa dataa ulkoisen muistin bankista 0. Prosessorin tehtäväksi jää tarkastaa kytkimien tila ajoittain.

Nestekidenäytön ohjaaminen on hieman hankalampaa. Sähköinen kytkentä on to-



Kuva 23: PC:lle kirjoitetun ohjausohjelman käyttöliittymä.

teutettu D-kiikuilla, joiden avulla signaaliprosessoria hitaammin toimivalle näytölle menevä data voidaan säilyttää riittävän pitkään. Näytön ohjaus tapahtuu antamalla käskyjä näyttömodulilla olevalle piirille, joka varsinaisesti ohjaa piste-matriisinäyttöä [Burian, 1996]. Käskyjen lähetys ja ajoitus on jätetty SHARCille. Käyttöliittymälaajennus ja sen yhteydessä tarvittava ohjelmisto on toteutettu Teknillisessä korkeakoulussa tietotekniikan erikoistyönä. Tarkempi kuvaus löytyy lähteestä [Merimaa, 2000].

4.4 Ohjaus PC:llä

SHARCin suorittama auralisaatio-ohjelmisto on talletettu signaaliprosessorikortille asennetulle flash-muistipiirille. Flash-muistin käyttö mahdollistaa auralisaattorin suodattimien ohjelmallisen päivityksen. Samoin muistille on mahdollista tallettaa muutama sarja valmiita auralisaatioasetuksia. Näitä toimintoja varten PC:lle on kirjoitettu Windows-ympäristössä toimiva ohjausohjelma. Ohjelmalla on mahdollista samalla ohjata myös itse auralisaatiota. Kuvassa 23 on esitetty ohjelman käyttöliittymä.

Ohjausohjelma on kirjoitettu Microsoftin Visual C++:lla [Microsoft, 1997]. Oh-

jelma kommunikoi auralisaattorin kanssa PC:n sarjaportin välityksellä. Signaali-prosessorin päässä sarjaportin liikenteestä huolehtii EZ-KITin mukana toimitettava kernel-ohjelma, jota on hieman modifioitu auralisaattorin tarpeisiin. Kernel tarkkailee jatkuvasti kortin RS-232-portin liikennettä ja ottaa vastaan määrämutoisia viestipaketteja [Analog Devices, 1997a, Analog Devices, 2000b]. Viestien sisältämien käskyjen perusteella kernel osaa siirtää dataa prosessorin muistin ja sarjaportin välillä.

PC-ohjelma avaa käynnistettäessä sarjaportin, ottaa yhteyden auralisaattoriin ja asettaa laitteiden välisen sarjaliikenteen parametrit. Tämän jälkeen se odottaa käyttäjän syötettä ja antaa syötteen mukaan tarvittavat komennot auralisaattorille. SHARCissa auralisaatioparametrit ja suodatinkertoimet sijaitsevat ennaltamääräytyissä muistiosoitteissa, joten niitä voidaan muuttaa kirjoittamalla dataa suoraan prosessorin muistiin. Kernel ilmoittaa vastaanotetusta datasta auralisaatio-ohjelmistolle keskeytyspyynnön avulla.

5 Toteutus

Auralisaatio-ohjelmiston kehitysympäristönä on käytetty Analog Devicesin VisualDSP:tä [Analog Devices, 1998]. VisualDSP on Windows-ympäristössä toimiva ns. IDE-ohjelmisto (Integrated Development Environment), joka sisältää editorin ja useiden lähdekooditiedostojen hallintaan tarvittavat apuneuvot. Ohjelmistoon kuuluvat assembler- ja C-kääntäjät sekä linkkeri. Virheiden etsimistä varten VisualDSP sisältää debuggerin, joka tukee työssä käytettyä EZ-ICE-emulaattoria (In Circuit Emulator) [Analog Devices, 1999]. EZ-ICE:n avulla ohjelmakoodin suoritusta on mahdollista tarkkailla suoraan prosessorilla.

Auralisointiohjelmisto on pyritty optimoimaan mahdollisimman tehokkaaksi niiltä osin, missä prosessorin suorituskyky rajoittaa auralisaation laatua. Binauraaliseen auralisaatioon liittyvä lähdekoodi on kirjoitettu kokonaisuudessaan assembler-kielellä. VBAP-toteutuksen pohjana on käytetty Pulkin Silicon Graphics-ympäristöön kirjoittamaa C-kielistä ohjelmistoa [Pulkki, 1999a], jota on muokattu sopivammaksi signaaliprosessoriympäristöön ja kyseisen sovelluksen tarpeisiin. Esimerkkejä lähdekoodista on esitetty liitteessä A.

Signaaliprosessorin rajallinen suorituskyky on otettu huomioon erityisesti suodatinsuunnittelussa, jossa erilaisilla suodatinrakenteilla voidaan samalla laskentateholla saada aikaan hyvin eritasoisia toteutuksia. Kaikki suodatinsuunnitteluun liittyvä laskenta on tehty MATLAB-ohjelmalla [Mathworks, 1996]. Seuraavassa käsitellään auralisaation toteutusta käytännössä. Suodatinsuunnittelua ja binauraalista auralisaatiota käsitellään kappaleissa 5.1–5.3. Kappaleessa 5.4 kuvataan VBAP-algoritmin toteutus ja kappaleessa 5.5 käsitellään koko auralisaation reaaliaikaista ohjausta.

5.1 Binauraalinen synteesi

Binauraalinen auralisaatio perustuu HRTF:iä approksimoiviin suodattimiin. Nämä voidaan toteuttaa useilla eri tavoilla. Yksinkertaisin toteutustapa on käyttää mitattuja impulssivasteita suoraan FIR-suodattimina (Finite Impulse Response). Mittaustulokset ovat kuitenkin tyypillisesti satojen näytteiden pituisia ja sisään-tulosignaali pitää suodattaa erikseen molemmille korville. Näin pitkien suodat-

timien käyttö reaaliaikaisessa monikanava-auralisaatiossa on mahdotonta, minkä seurauksena joudutaan tyytymään hieman huonompiin approksimaatioihin.

HRTF:ien jakamista minimivaiheiseen suodattimeen ja taajuusriippumattomaan viiveeseen käsiteltiin jo kappaleessa 2.3. Minimivaiheisen rekonstruktion avulla voidaan tehostaa HRTF:ien alussa olevan viiveen toteutusta. 44,1 kHz:n näytteenottotaajuudella 0,8 millisekunnin ITD vastaa noin 35 näytettä tyhjää kauemman korvan impulssivasteen alussa. Mahdollisuus toteuttaa tämä muutamalla signaaliprosessorin käsikyllä on huomattava parannus verrattuna koko impulssivasteen toteuttamiseen suodattimen avulla. Lisäksi HRTF:ien magnitudispektrin toteuttamiseen jää saman verran laskentakapasiteettia suunnasta riippumatta.

Toteutuksen kannalta sopivan pituinen minimivaiheinen FIR-suodatin voidaan suunnitella näytteistämällä suodattimen taajuusvastetta halutulla tarkkuudella ja laskemalla impulssivaste Fourier-käänteismuunnoksella. Vastaava toimenpide voidaan suorittaa aika-alueessa ikkunoimalla minimivaiheista impulssivastetta suorakaideikkunalla eli yksinkertaisesti katkaisemalla vaste haluttuun pituuteen. Suorakaideikkunan käyttö minimoi approksimaatiossa syntyvän kokonaisenergiavirheen, mutta aiheuttaa ns. Gibbsin ilmiön, joka näkyy aaltoiluna magnitudispektrin epäjatkuvuuskohdissa [Oppenheim ja Schaffer, 1975]. Koska HRTF:t eivät yleensä sisällä kovin jyrkkiä spektraalisia epäjatkuvuuksia, Gibbsin ilmiö ei kuitenkaan ole vakava ongelma. Suorakaideikkunan on havaittu toimivan HRTF-suodattimien suunnittelussa paremmin kuin muiden useimmiten käytettyjen ikkunafunktioiden [Huopaniemi, 1999].

Toinen vaihtoehto minimivaiheisten HRTF-suodattimien toteuttamiseen on käyttää takaisinkytkentää hyödyntäviä IIR-suodattimia (Infinite Impulse Response). IIR-suodattimen laskenta vaatii yhden MAC-operaation jokaista siirtofunktion napaa ja nollaa kohden, kun FIR-suodattimessa operaatioita tarvitaan yksi jokaista nollaa (kerrointa) kohden. Jatkossa käytetään termiä suodattimen astelukku kuvaamaan FIR-suodattimen kertoimien lukumäärää tai yhtä monta napaa ja nollaa sisältävän IIR-suodattimen napojen tai nollien lukumäärää. Tällöin IIR-toteutus vaatii kaksi kertaa niin paljon laskentatehoa kuin samanasteinen FIR-toteutus.

IIR-suodattimilla on mahdollista päästä FIR-suodattimia tehokkaampiin toteutuksiin tilanteissa, joissa tarvitaan jyrkkiä magnitudivasteita kapealla taajuusa-

lueella. IIR-suodattimien suunnittelu on kuitenkin vaikeampaa ja lopputulos riippuu käytetystä suunnittelumenetelmästä. Lähteessä [Huopaniemi et al., 1999] on esitetty joitakin kirjallisuudesta löytyviä astelukuja HRTF-suodatintoteutuksille. Käytetyt asteluvut vaihtelevat FIR-suodattimilla välillä 40–80 ja IIR-suodattimilla välillä 10–48. Näiden toteutusten vertailu on kuitenkin hyvin vaikeaa, koska osassa on käytetty yksilöllisiä ja osassa keinopää-HRTF:iä erilaisilla ekvalisoinneilla. Myös arviointimenetelmät vaihtelevat teoreettisesta arvioinnista kuuntelukokeisiin.

Cortex MK2 -keinopään HRTF:illä suoritetuissa kuuntelukokeissa on verrattu toisiinsa minimivaiheiseen rekonstruktioon perustuvia FIR- ja IIR-toteutuksia [Huopaniemi ja Karjalainen, 1997]. Tulosten mukaan 48 kHz:n näytteenottotaajuudella 40-asteinen FIR-suodatin on riittävä, jotta 75 % kuulijoista ei huomaa eroa referenssinä käytettyyn 300-asteiseen FIR-toteutukseen verrattuna. Vastaava luku Pronyn menetelmällä suunnitelluille IIR-suodattimille on 25. Ns. varpatuilla IIR-suodattimilla [Härmä et al., 2000] päästiin astelukuun 20, mutta tällaisten suodattimien toteutus vaatii 4 MAC-operaatiota astelukua kohden. FIR-toteutus toimii siis näistä vaihtoehdoista tehokkaimmin.

Edellisen perusteella auralisaattorissa on käytetty HRTF:ien minimivaiheista rekonstruktiota ja FIR-suodattimia. FIR-suodattimien käyttö helpottaa samalla myös HRTF:ien reaaliaikaista interpolointia. 44,1 kHz:n näytteenottotaajuudella mitatut Kemar-keinopään HRTF:t [Gardner ja Martin, 1994] on minimivaiheistettu MATLAB Signal Processing Toolboxiin [Mathworks, 1994] kuuluvalla `rceps`-funktiolla¹.

Minimivaiheiseen rekonstruktioon liittyvät taajuusriippumattomat viiveet on laskettu sovittamalla MATLABin `polyfit`-käskyllä suora alkuperäisen HRTF:n ja minimivaiheisen suodattimen vaihe-eroa kuvaavaan käyrään taajuusvälille 1–5 kHz [Jot et al., 1995]. Suoran kuvaaman taajuusriippumattoman viiveen aiheuttama negatiivinen vaihe näytteenottotaajuudella saadaan suoran kulmakertoimen k ja näytteenottotaajuuden f_s tulona. Viive näytteinä saadaan edelleen jakamalla ko-

¹Tuloksena saatavien suodattimien analyysin perusteella on havaittu, että funktio palauttaa joskus myös sekavaiheisia suodattimia, joissa muutama hyvin lähellä yksikköympyrää sijaitseva nolla on ympyrän ulkopuolella. Tämä on syytä huomioida interpoloinnissa ja erityisesti käänteissuodattimia suunniteltaessa.

konaisvaihe yhden näytteen aiheuttamalla vaihesiirrolla

$$\tau = -\frac{k f_s}{2\pi} \quad (39)$$

Näin saadut viiveet, samoin kuin kaikki suodatinkertoimet, on tallennettu signaaliprosessorin HRTF-tietokantaan 32-bittisinä liukulukuina.

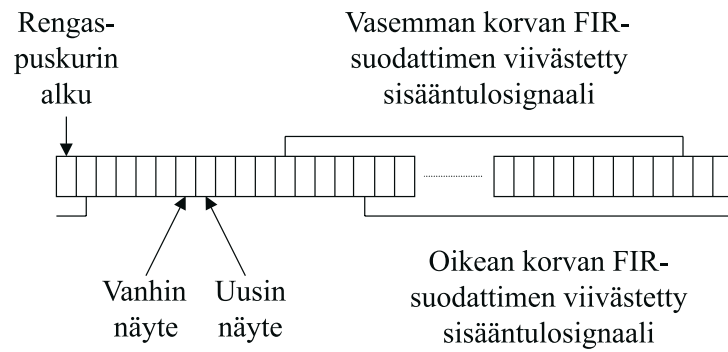
Reaaliaikaisen toteutuksen helpottamiseksi taajuusriippumattomat viiveet pyöristetään interpoloinnin jälkeen kokonaislukuiksi. Tämä aiheuttaa auralisaatioon pienen ITD-virheen, joka on maksimissaan $t_e = 1/f_s$. Vastaavaa virhettä havaitussa vaakakulmassa voidaan arvioida kaavalla [Huopaniemi, 1999, s. 70]

$$\varphi_e = \arcsin\left(\frac{ct_e}{2a}\right) \quad (40)$$

missä c on äänennopeus ilmassa (340 m/s) ja a pään säde (0,0875 m). Kaava on johdettu pallonmuotoiselle päälle äänilähteen ja eri korvien etäisyyseroista. 44,1 kHz:n näytteenottotaajuudella maksimivirheeksi saadaan siis n. $2,5^\circ$. Tämä on vähemmän kuin keskimääräisen kuuntelijan lokalisaatiotarkkuus suoraan edessä [Blauert, 1997]. Tarvittaessa parempaa tarkkuutta viiveet voitaisiin toteuttaa myös ns. murtoviivesuodattimilla [Laakso et al., 1996].

40 MHz:n kellotaajuudella toimiva SHARC-prosessori pystyy teoriassa laskemaan yhden FIR-kertoimen joka kellojaksolla. Käytännössä keskeytyskäsitteily ja prosessorin liukuhihnan aiheuttamat viivästykset vievät hieman prosessoriaikaa. Karkeasti voidaan arvioida, että jokaista sisääntulevaa kahdeksan näytteen sarjaa kohden on käytettävissä noin 900 kellojaksoa. Tästä kuulokekompensointiin tai ristiinkuulumisen estoon (kappaleet 5.2 ja 5.3) tarvitaan hieman yli 100 kellojaksoa ja kaikkien kanavien auralisaatiolle yhteisiin operaatioihin n. 20 kellojaksoa. Kun vielä jätetään jonkin verran prosessoriaikaa satunnaisten keskeytyksillä hoidettavien operaatioiden suorittamiseen, HRTF-suodatukseen jää n. 720 kellojaksoa eli 90 sisääntulokanavaa kohti.

Jokaisen sisääntulokanavan auralisoinnissa joudutaan laskemaan kaksi konvoluutiota FIR-suodattimien impulssivasteiden ja viivästetyjen sisääntulosignaalien välillä. Kunkin sisääntulevan signaalin näytteet tallennetaan yhteen rengasrakenteella toteutettuun viivelinjaan, josta konvoluutiosta tarvittavat näytteet luetaan HRTF-suodattimeen liittyvän taajuusriippumattoman viiveen mukaisella näytemäärällä viivästettynä. Suodatuksen toteutusta on havainnollistettu kuvassa 24.

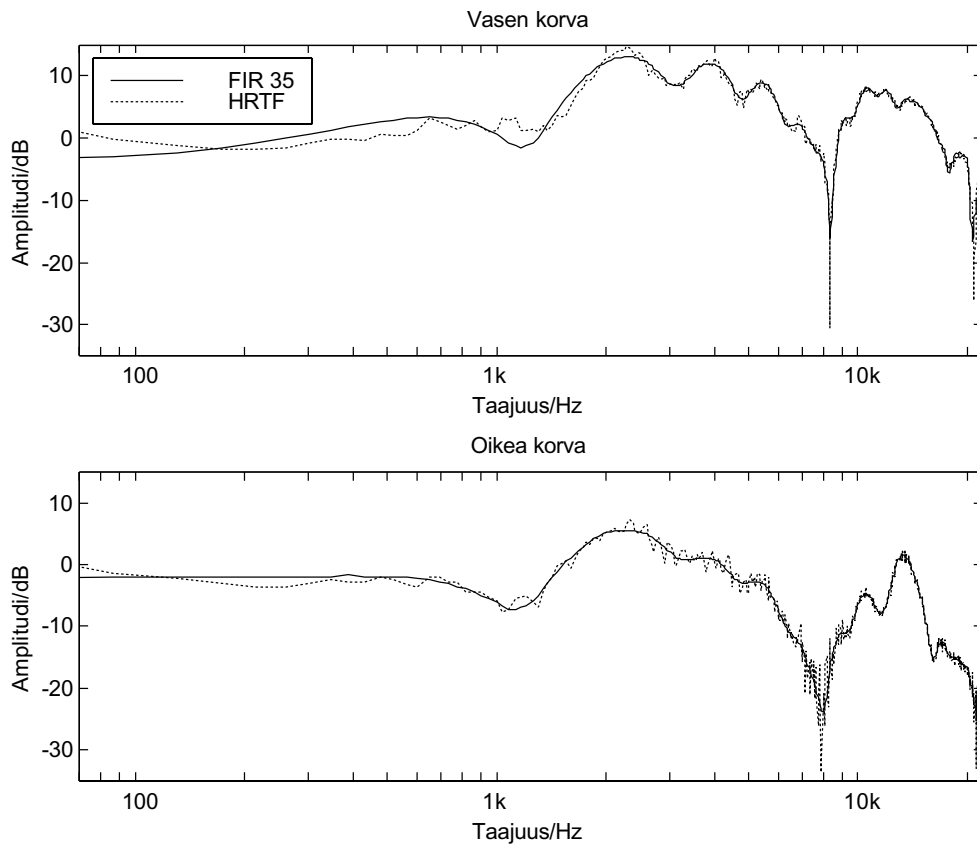


Kuva 24: Rengaspuskurilla toteutettu sisääntulosignaalin viivelinja. Uusin näyte tallennetaan aina vanhimman päälle. HRTF:ien taajuusriippumattomat viiveet toteutetaan käyttämällä suodatuksessa tarvittavalla näytemäärällä viivästettyjä signaaleja.

Näytteiden tallentamiseen, viiveiden toteutukseen ja suodatuksen aloittamiseen kuluu yhdellä kanavalla 20 kellojaksoa. Itse suodatukseen jää siis 70 kellojaksoa, mikä mahdollistaa 35-asteisten FIR-suodattimien käytön.

Kuvassa 25 on esitetty 35-asteisten minimivaiheisten FIR-suodatintoteutusten magnitudispektrit vasemmalta 30° suunnasta saapuvalle äänelle. KEMARin mitaustuloksille on ensin suoritettu systeemikompensointi jakamalla niiden kompleksinen spektri mittauksessa käytetyn kaiuttimen taajuusvasteella. Systeemikompensoidut vasteet on normalisoitu niin, että niiden taso välillä 100 Hz – 11 kHz on keskimäärin 0 dB. Näin saatujen täysimittaisten HRTF:ien magnitudispektrit on myös esitetty kuvassa. Magnitudispektreistä on edelleen muodostettu minimivaiheiset impulssivasteet, jotka on katkaistu 35-asteisiksi FIR-suodattimiksi. Kuvasta nähdään, että näinkin lyhyet suodattimet riittävät melko hyvään HRTF-aproksimaatioon.

Auralisaattorin HRTF-tietokanta koostuu elevaatiokulmilla $-30, 0, 30$ ja 60 astetta kymmenen asteen välein mitatuista sekä suoraan ylhäältä tulevan äänen HRTF:istä. Kemar-keinopää on oletettu symmetriseksi, jolloin ainoastaan toisen puolen vasteiden tallentaminen riittää. Oikealla puolella sijaitsevat virtuaaliset lähteet voidaan tuottaa vasemman puolen suodattimista vaihtamalla ulostulokanavat keskenään. Yhteensä auralisointiin tarvitaan siis $4 \times 19 + 1 = 77$ suodattinparia. 35-asteisilla suodattimilla ja erillisillä viiveillä tähän kuluu muistia 5544 sanaa, mikä vastaa noin 175 kilobittiä. Tällainen määrä voidaan helposti sijoittaa

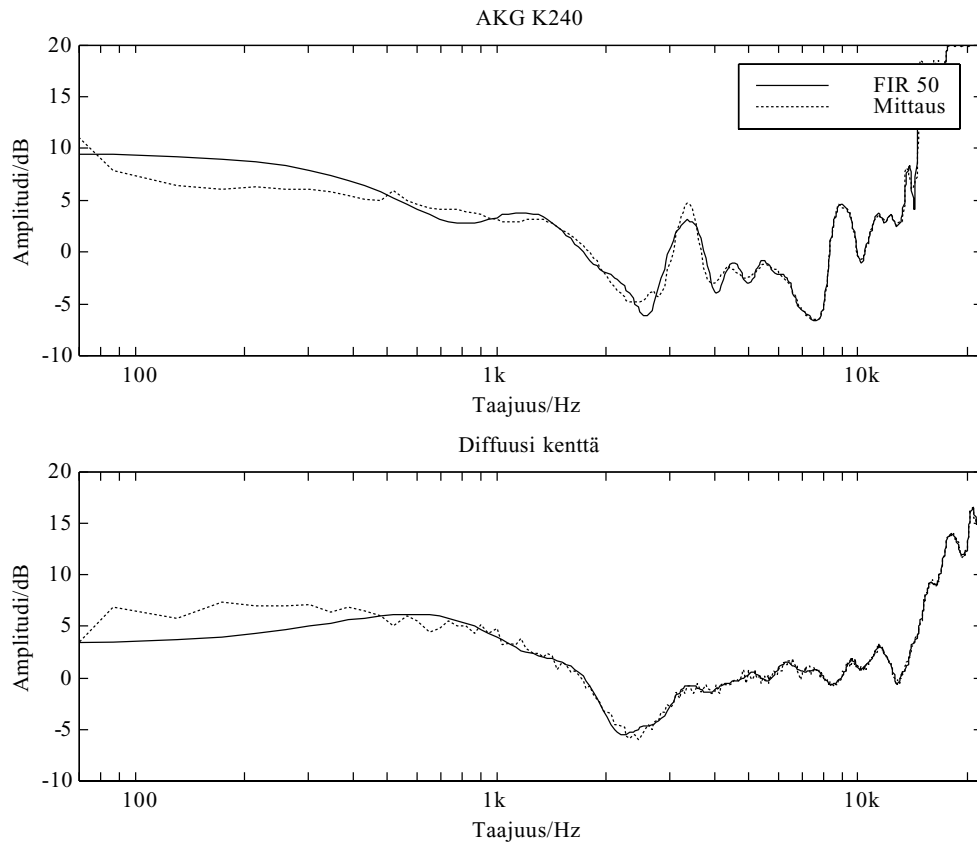


Kuva 25: Vaakatasossa 30° suunnasta vasemmalta saapuvan äänen HRTF:t ja näiden 35-asteiset minimivaiheiset FIR-suodatinapproksimaatiot.

signaaliprosessorin sisäiseen muistiin. Tallennettujen HRTF:ien välissä olevat vasteet interpoloidaan neljästä ympäröivästä mittaustuloksesta bilineaarisesti asteen tarkkuudella.

5.2 Kuulokekompensointi

Kuulokekompensointia varten auralisaattoriin on suunniteltu joitakin vaihtoehtoisia suodattimia. Käyttäjä voi valita, mitä suodatinta käytetään tai kytkeä halutessaan kompensoinnin kokonaan pois päältä. Käyttäjä voi ladata auralisaattoriin myös henkilökohtaisia kompensointisuodattimia PC:n käyttöliittymältä. Tilaa on varattu yhteensä viidelle 50-kertoimiselle FIR-suodattimelle. Kuulokkeet ja kuulijan korvat on oletettu riittävän symmetrisiksi, jotta molemmille korville voidaan käyttää samaa suodatinta.



Kuva 26: Kuulokekompensoinnissa käytetyt käänteissuodattimet ja niiden toteutus 50-asteisilla FIR-suodattimilla AKG K240 -kuulokkeille sekä diffuusin kentän ekvalisoinnille.

Auralisaattoriin on laskettu valmiiksi käänteissuodattimet suoraan edessä sijaitsevan äänilähteen sekä diffuusin kentän HRTF:ille. Näitä voidaan käyttää yleis-suodattimina vastaavasti ekvalisoiduille kuulokkeille. Lisäksi kolmelle KEMAR-keinopään HRTF-datojen mukana toimitettavista kuulokevasteista on laskettu mallikohtaiset käänteissuodattimet. Kuvassa 26 on esitetty AKG K240-kuulokkeiden ja diffuusin kentän ekvalisoinnin kompensoinnissa tarvittavien käänteissuodattimien magnitudivasteet sekä näiden toteutukset 50-asteisilla FIR-suodattimilla.

Mitattujen kuulokevasteiden käänteissuodattimet on laskettu keskiarvoistamalla ensin molempien korvien mittaustulosten magnitudispektrit ja normalisoimalla nämä keskimäärin tasolle 0 dB. Diffuusin kentän vasteet on laskettu HRTF:istä kaavalla (16). Suodattimet on edelleen minimivaiheistettu ja käänteissuodattimet on muodostettu taajuusalueessa rajoittaen niiden amplitudi enintään 20 dB:iin.

Joidenkin kuulokkeiden tapauksessa IIR-suodattimien käyttö olisi tarpeen, jos kuulokkeiden heikko bassotoisto haluttaisiin kokonaan korjata. Kymmenien desibelien korostuksella ei kuitenkaan saada hyvää tulosta aikaan, jos kuulokkeita ei ole jo valmiiksi suunniteltu toistamaan matalia ääniä riittävän voimakkaasti. Tämän vuoksi FIR-suodattimet on katsottu sopiviksi kompensointikäyttöön. FIR-suodatintoteutus helpottaa myös käyttäjän työtä, jos auralisaattoriin halutaan myöhemmin suunnitella henkilökohtaisia kompensointisuodattimia.

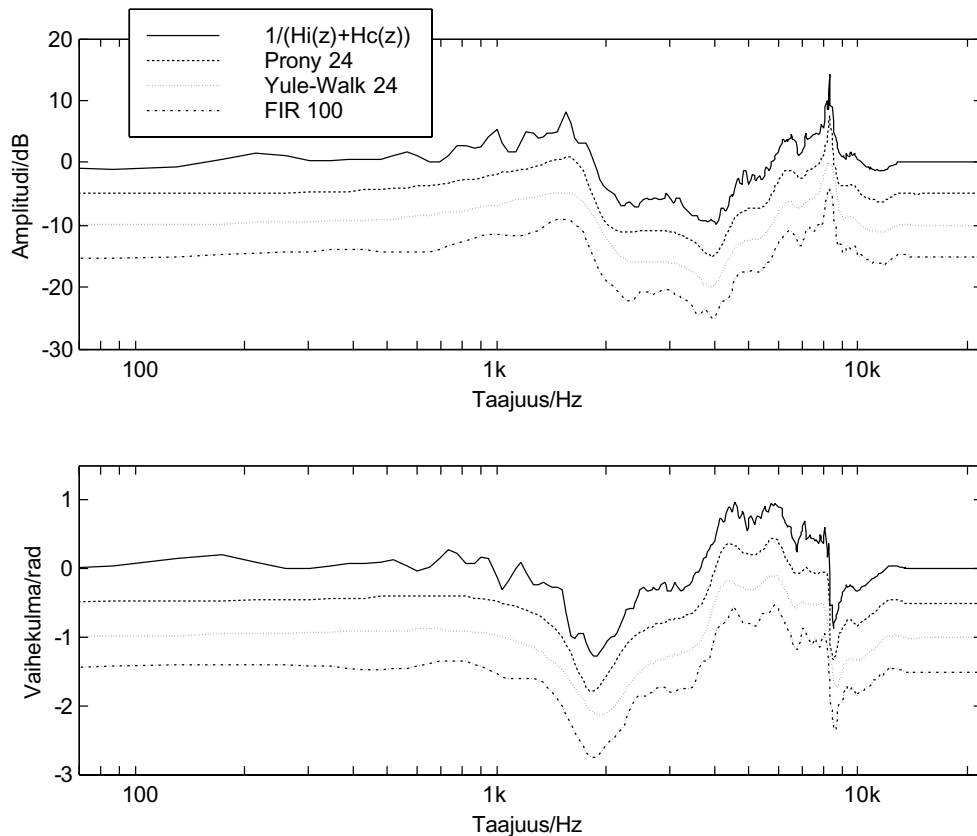
50-asteisen FIR-suodattimen laskenta molempien korvien signaaleille vaatii prosessoriaikaa hieman yli 100 kellojaksoa. Kuulokekompensointi olisi ollut mahdollista yhdistää myös HRTF-suodattimiin. Tällöin binauraaliseen kaiutinauralisaatioon olisi kuitenkin tarvittu erilliset kompensoimattomat HRTF-suodattimet, jolloin kaikki suodatinkertoimet eivät olisi enää mahtuneet signaaliprosessorin sisäiseen muistiin.

5.3 Ristiinkuulumisen esto

Ristiinkuulumisen kumoaminen johtaa yleensä impulssivasteeltaan melko pitkiin suodattimiin [Jot et al., 1995]. Tällöin IIR-suodatintoteutukset saattavat toimia FIR-suodattimia tehokkaammin. Kuvissa 27 ja 28 on vertailtu joitakin toteutuksia sekoitinrakenteisessa ristiinkuulumisenestossa (kts. kappale 2.6) tarvittaville käänteissuodattimille $1/[H_i(z) + H_c(z)]$ ja $1/[H_i(z) - H_c(z)]$.

Suodattimet on laskettu $\pm 30^\circ$ suuntaisista KEMAR-keinopään HRTF:ista muodostamalla ensin eri korvien vasteiden summa ja erotus. Näiden logaritmista magnitudia on ikkunoitu kuvan 12 mukaisella funktiolla. IIR-suodattimet on suunniteltu MATLAB Signal Processing Toolboxin [Mathworks, 1994] käskyillä `prony` ja `yulewalk`. FIR-suodattimet on laskettu aika-alueessa suorakaideikkunalla. Kuvista nähdään, että 100-asteiset FIR-suodattimet ovat selvästi liian lyhyitä. Erityisesti erotussuodattimessa suorakaideikkunoinnista johtuva Gibbsin ilmiö on nähtävissä voimakkaana aaltoiluna suodattimen magnitudivasteessa. Kuvien perusteella toteutukseen valittiin Pronyn menetelmällä suunnitellut 24-asteiset IIR-suodattimet.

IIR-suodatintoteutukset ovat FIR-suodattimia herkempiä laskennassa tapahtuville epätarkkuuksille. Auralisaattorin 24-asteiset suodattimet on toteutettu 12

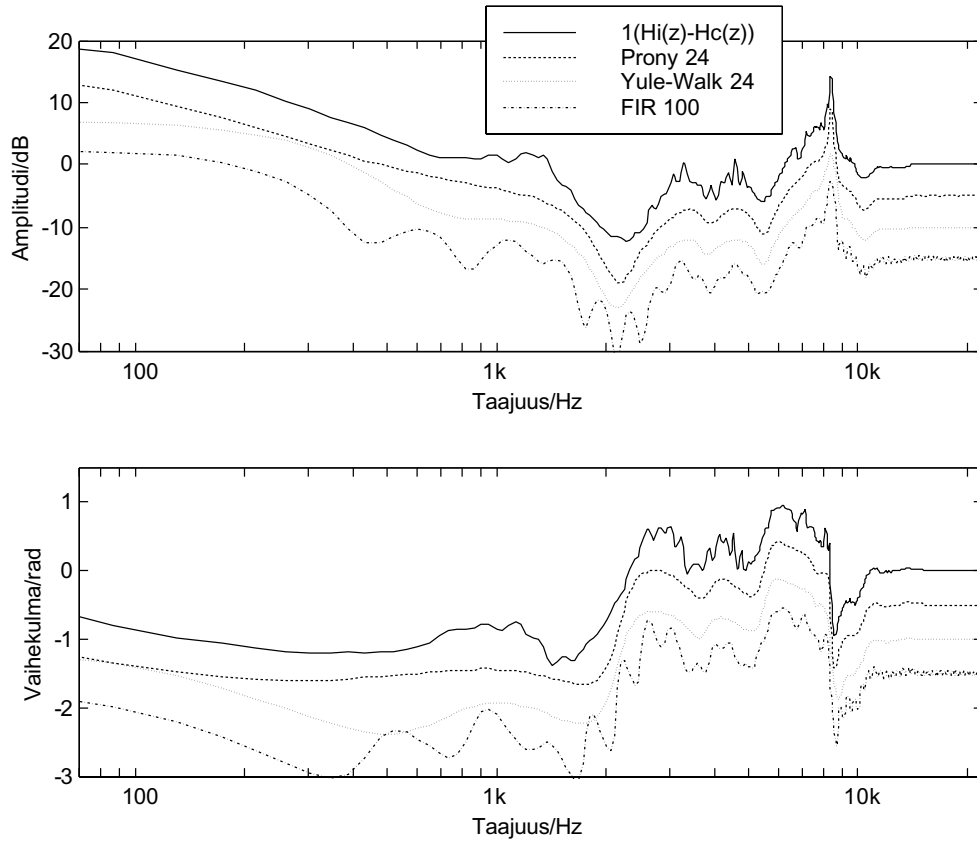


Kuva 27: Ristiinkuulumisen estossa käytettävän suodattimen $1/[H_i(z) + H_c(z)]$ magnitudi ja vaihespektrit sekä näiden kolme approksimaatiota. Vertailun helpottamiseksi magnitudispektrit on piirretty 5 dB:n ja vaihespektrit 0,5 radiaanin etäisyydelle toisistaan.

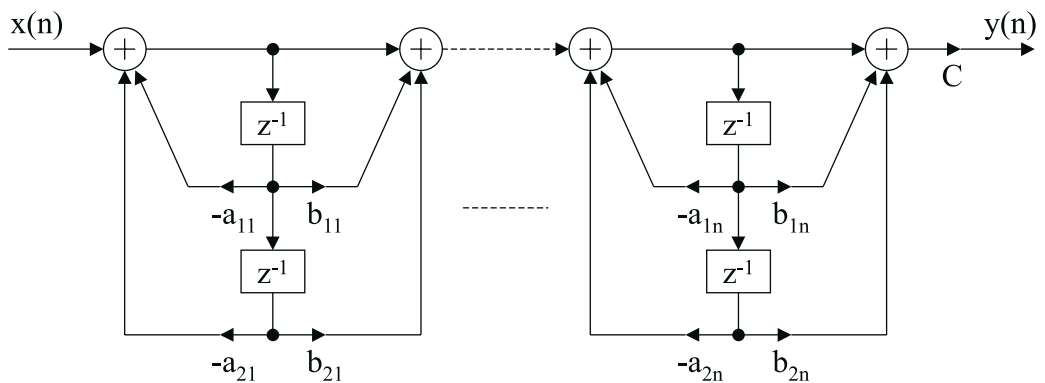
toisen asteen kanonisen lohkon kaskadina (kuva 29). Yhden toisen asteen lohkon siirtofunktio on

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \quad (41)$$

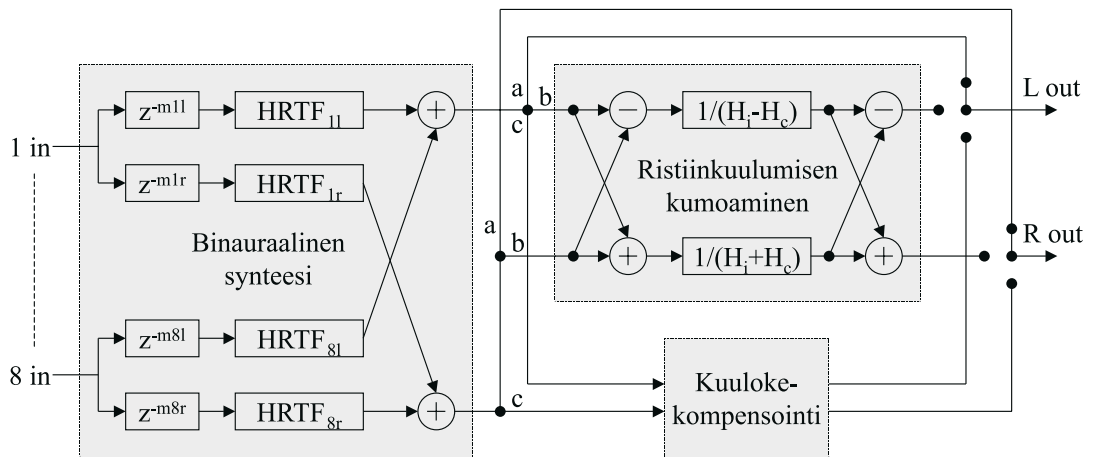
Tällä on z -tasossa kaksi napaa ja nollaa, jotka molemmat ovat kompleksikonjugattipareja. Useamman lohkon kaskadissa napa- ja nollaparit voidaan yhdistää monella eri tavalla. Kun parit valitaan niin, että ne sijaitsevat kompleksitasossa mahdollisimman lähellä toisiaan, saadaan magnitudivasteeltaan mahdollisimman tasaisia lohkoja. Tällöin niissä tapahtuvien pyöristysvirheiden aiheuttama kohina on myös pienin mahdollinen. [Ifeachor ja Jervis, 1993] Toteuksen nopeuttamiseksi auralisaattorin toisen asteen lohkot on edelleen skaalattu niin, että $b_0 = 1$ ja



Kuva 28: Ristiinkuulumisen estossa käytettävän suodattimen $1/[H_i(z) - H_c(z)]$ magnitudi ja vaihespektrit sekä näiden kolme approksimaatiota.



Kuva 29: n :stä peräkkäisestä toisen asteen kanonisesta lohokosta muodostuva IIR-suodatin. C kuvaa koko kaskadin yhteistä vahvistusta.



Kuva 30: Binauraalisen auralisaation lohkokkaavio. Binauraalisen synteesin tuloksena saatava signaali reititetään joko a) suoraan ulostuloon, b) ristiinkuulumisen kumoavien suodattimien läpi tai c) kuulokekompensointisuodattimien läpi.

kaskadin yhteinen vahvistus on toteutettu erillisellä kertoimella.

Kahden 24-asteisen IIR-suodattimen laskenta vaatii hieman yli 100 kellojaksoa prosessoriaikaa, eli saman verran kuin kuulokekompensointi. Auralisaattorissa HRTF-suodattimien ulostulo reititetään vaihtoehtoisesti joko kuulokekompensointi- tai ristiinkuulumisenestosuodattimien läpi. Kuvassa 30 on esitetty auralisaattorin koko binauraalisen osan lohkokkaavio.

5.4 VBAP-algoritmi

Auralisaattorin VBAP-toteutus eroaa melko paljon binauraalisen auralisaation toteutuksesta. VBAP-algoritmi on kirjoitettu kokonaisuudessaan C-kielellä Pulkin aikaisemman toteutuksen pohjalta [Pulkin, 1999a]. Pulkin lähdekoodia on muokattu jonkin verran signaaliprosessoriympäristöön sopivammaksi. Erityisesti muistinkäyttöä on optimoitu. Kaiutinkolmioiden ja -parien laskennassa käytetyt dynaamisesti allokoituneet linkitettyt listat on korvattu käyttämällä vahvistuskertoimille joka tapauksessa varattavaa kiinteää taulukkoa.

VBAP-algoritmi on toteutettu niin, että kaiutinkonfiguraation vaihtuessa valitaan valmiiksi ei-päällekkäiset kahden tai kolmen kaiuttimen ryhmät, jotka kattavat kaikki mahdolliset auralisaatiosuunnat. Valintaan käytetään kappaleessa 3.4

kuvattua algoritmia. Kaiutinryhmille lasketaan myös valmiiksi panoroinnin vahvistuskertoimien laskennassa tarvittavat käänteismatriisit. Merkitään kolmiulotteisen amplitudipanoroinnin käänteismatriisia tässä symbolilla

$$\mathbf{K} = \mathbf{L}^{-1} = \begin{bmatrix} k_{11} & k_{21} & k_{31} \\ k_{12} & k_{22} & k_{32} \\ k_{13} & k_{23} & k_{33} \end{bmatrix} \quad (42)$$

jolloin vahvistuskertoimet $\mathbf{g} = [g_1 \ g_2 \ g_3]^T$ saadaan kaavalla

$$\mathbf{g} = \mathbf{K}\mathbf{p} \quad (43)$$

missä vektori $\mathbf{p} = [p_1 \ p_2 \ p_3]^T$ on haluttuun auralisaatio-suuntaan kolmion sisällä osoittava yksikkövektori. Kaksiulotteisessa tapauksessa tarvittava matriisi \mathbf{K} on kokoa 2×2 . Tämä voidaan tallettaa kolmiulotteisen amplitudipanoroinnin matriisiin muodossa

$$\mathbf{K} = \begin{bmatrix} k_{11} & k_{21} & 0 \\ k_{12} & k_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (44)$$

Tällöin sekä kaksi- että kolmiulotteisessa toteutuksessa voidaan itse panorointiin käyttää täsmälleen samaa algoritmia. Kaksiulotteisessa tapauksessa saadaan aina $g_3 = 0$, jolloin panorointi tapahtuu kertoimilla g_1 ja g_2 niitä vastaavien kahden kaiuttimen välillä. Käyttäjän pitää kuitenkin määrittää, käytetäänkö kaksi- vai kolmiulotteista amplitudipanorointia, jotta kaiutinryhmät ja käänteismatriisit saadaan muodostettua oikein.

Kolmiulotteisten kaiutinkolmioiden valinta ja käänteismatriisien laskenta ovat laskennallisesti VBAP-algoritmin raskaimpia osia. Operaatioita ei onneksi tarvitse suorittaa kuin kaiutinkonfiguraation muuttuessa. VBAP:in muut osat toimivat helposti reaaliajassa. Virtuaalisen lähteen suunnan muuttuessa lähteelle pitää valita uusi kaiutinryhmä ja laskea vahvistuskerroinvektori \mathbf{g} . Ryhmän valinta tapahtuu laskemalla vahvistuskertoimet kaikille mahdollisille kaiutinryhmille. Koska ryhmien auralisaatioalueet eivät ole päällekkäisiä, oikea ryhmä löytyy ehdolla $g_1, g_2, g_3 \geq 0$. Jos mikään ryhmä ei täytä tätä ehtoa, halutunsuuntainen auralisaatio ei ole mahdollinen.

Kanavakohtainen vahvistus ja reititys kaiuttimille tehdään hyvin tehokkaasti toimivalla assembler-kielisellä rutiinilla. VBAP-algoritmia käytettäessä prosessori onkin suurimman osan ajasta tyhjäkäynnillä. Auralisaatiosuuntien muuttuessa tarvittava vahvistuskertoimien uudelleenlaskenta on myös melko kevyt operaatio.

5.5 Reaaliaikainen ohjaus

Auralisaation reaaliaikainen ohjaus käsittää auralisaatiomenetelmän valinnan sekä auralisaatiosuuntien ja eri kanavien vahvistuskertoimien muuttamisen laitteen toiminnan aikana. Lisäksi kuulokekompensointiin käytettäviä suodattimia on mahdollista vaihtaa. Näiden toimintojen suorittaminen voi hetkittäin vaatia melko paljon prosessoriaikaa. Binauraalisen auralisaation aikana lähes kaikki prosessorin teho käytetään kuitenkin suodatinlaskentaan. Hyödyntämällä SHARCin keskeytyskäsittelyä ohjaustoiminnot on mahdollista jakaa useamman näytejakson ajalle.

Auralisaattorin käynnistyksen yhteydessä tapahtuvien alustusten jälkeen kaikki ohjelmakoodin suoritus tapahtuu keskeytyskutsuilla. SHARC mahdollistaa useampien keskeytysten sisäkkäistämisen niin, että korkeamman prioriteetin omaava keskeytys voidaan käsitellä kesken matalemmän prioriteetin keskeytyksen. Keskeytysten prioriteetteja ei voi säätää, mutta joissakin tapauksissa on mahdollista valita, mitä keskeytystä käytetään.

Uusien sisääntulevien näytteiden käsittely tapahtuu prosessorin sarjaporttia lukevan DMA-ohjaimen suorittamalla keskeytyskutsulla. Tämän prioriteetti on ohjelmistossa tarvittavista keskeytyksistä toiseksi korkein. Näinollen prosessorin normaalin toiminnan aikana kaikki muu toiminta keskeytyy uusien näytteiden tullessa sisään. Näytteiden käsittelyä on edelleen nopeutettu kontekstin vaihdolla (kts. kappale 4.1). Näytekeskeytyksen alussa käyttöön vaihdetaan rekisterit, jotka on valmiiksi alustettu HRTF-suodattimien laskentaa varten. Keskeytyksen lopussa vaihdetaan jälleen takaisin aikaisempiin rekistereihin, ja prosessorin toiminta jatkuu siitä, mihin keskeytyksen alussa jäätiin.

Korkein prioriteetti on signaaliprosessorikortin RS-232-sarjaportin suorittamalla ulkoisella keskeytyskutsulla. Tämä kutsuu normaalisti kernel-ohjelmaa. Auralisaattorissa toteutus on kuitenkin tehty niin, että ulkoisen keskeytyskutsun käsit-

televä rutiini kutsuu toista matalampiprioriteettista keskeytystä, joka vasta ajaa tarvittavat kernel-rutiinit. Tällainen toteus hidastaa periaatteessa sarjaportin toimintaa, mutta ainakaan 9600 bps siirtonopeudella ongelmia ei ole esiintynyt.

Painokytkimien tilaa seurataan SHARCin ohjelmoitavan kellon suorittamilla keskeytyksillä. Kello on ohjelmoitu 30 Hz:n taajuudelle. Kytkimiien tilan perusteella kellokeskeytys muuttaa auralisaatioparametreja ja kirjoittaa uudet parametrit niihin liittyviin muistipaikkoihin. Kernel-ohjelma tekee tarvittaessa muutoksia näihin samoihin muistipaikkoihin. Parametrien muutoksista viestitetään auralisaatio-ohjelmistolle kutsumalla tähän tarkoitukseen varattua ohjelmallista keskeytystä.

Auralisaatioparametrien muutosten käsittely on melko monimutkainen operaatio. Virtuaalisten lähteiden suuntien muuttuessa ajetaan joko algoritmi, joka laskee interpoloimalla uudet HRTF-suodattimet, tai kutsutaan funktiota, joka laskee amplitudipanoroinnille uudet vahvistuskertoimet. HRTF-suodattimen nopea vaihto vastaa tilannetta, jossa joidenkin sisääntulevien näytteiden kohdalla impulssivaste muuttuu kesken toiseksi. Tämä saattaa aiheuttaa ääneen napsahduksina kuultavia transientteja. Matalaprioriteettisella keskeytyksellä tapahtuvassa suodattimien päivityksessä impulssivasteet saattavat jopa muuttua useammassa osassa. Kuuntelukokeiden perusteella transientit eivät kuitenkaan ole kuultavissa, jos suuntakulmien muutokset ovat riittävän pieniä. Tämä johtuu osittain suodattimien minimivaiheisista toteutuksista. Sekavaiheisilla suodattimilla napsahdukset ovat paljon selvempiä.

Napsahduksista voitaisiin päästä kokonaan eroon suodattamalla aina muutoksen jälkeen sisääntulevaa signaalia vähän aikaa kahdella suodattimella. Tällöin voitaisiin vaihtaa hitaasti suodattimen ulostulosta toiseen (engl. cross fade), jolloin napsahduksia ei kuuluisi. Tällainen toteutus on kuitenkin laskennallisesti hyvin raskas vaihtoehto, eikä sille ole tarvetta pääasiassa stationäärisille auralisaatio-suunnille suunnitellussa laitteessa.

Käyttäjän säädettävissä oleva kanavakohtaisen vahvistus huomioidaan auralisaation yhteydessä kertomalla sisääntuleva signaali erillisellä vahvistuskertoimella ennen muuta prosessointia. Vahvistuskerrointa säädetään logaritmisesti käyttäjän antaman lineaarisen parametrin mukaan.

Auralisaatioparametrien muutosten yhteydessä kutsuttava päivitysrutiini suorittaa myös muita operaatioita. Kun auralisaatiomenetelmä vaihtuu binauraalisen auralisaation ja amplitudipanoroinnin välillä, joudutaan laskemaan kaikki suodattimet tai vahvistuskertoimet uudestaan. Lisäksi rutiini hoitaa tarvittaessa kuulokekompensointisuodattimien vaihdon ja VBAP-algoritmin kaiutinkonfiguraatiomuutosten jälkeiset toimenpiteet, sekä huolehtii nestekidenäytön tekstin päivittämisestä.

6 Tulosten arviointi

Työssä toteutettu laitteisto toimii asetettujen teknisten tavoitteiden mukaisesti. Tavoitteena ei ollut arvioida kokeellisesti eri auralisaatiomenetelmien subjektiivista toimintaa, joten seuraavassa rajoitutaan epäformaaleihin havaintoihin. Kuunteluvaikutelmien perusteella binauraalinen kuulokeauralisaatio toimii melko hyvin horisontaalitasossa. Ääni paikallistuu yleensä haluttuun suuntaan, mutta etu-takasekaannuksia esiintyy joskus. Äänilähteen korkeuskulman simulointi sen sijaan toimii heikommin. Mediaanitasossa korkeuskulmaa on lähes mahdotonta arvioida. Sivuilla arviointi on hieman helpompaa.

Mediaanitason virtuaalisilla lähteillä ääni tuntuu melko usein paikallistuvan pään sisälle tai hyvin lähelle kuulijaa. Suoraan sivuilla suuntavaikutelma on luonnollisimman tuntuinen. Kuulokekuuntelussa diffuusin kentän kompensointisuodatin toimii varsin hyvin Sennheiser HD580 -kuulokkeita käytettäessä. AKG K240 -kuulokkeilla puolestaan diffuusin kentän kompensointi kuulostaa jonkin verran värityneeltä, mutta mitattujen kuulokevasteiden perusteella suunniteltu suodatin toimii hyvin.

Binauraalinen kaiutintoisto toimii tyydyttävästi kuulijan edessä noin 90° vaakakulmiin saakka. Suoraan takana olevien äänilähteiden simulointi ei sen sijaan onnistu lainkaan. Huoneakustiikka vaikuttaa myös hyvin paljon lopputulokseen. Hieman takaviistossa olevien äänilähteiden synnyttämä suuntahavainto on melko mielenkiintoinen, koska osa äänestä tuntuu tulevan edestä kaiuttimesta ja osa suunnilleen tarkoitetusta suunnasta.

Kaiutintoistossa käytettävä amplitudipanorointi ei ole yhtä herkkä huoneakustii-kan vaikutuksille kuin binauraalinen kaiutinauralisaatio. Amplitudipanoroinnilla auralisaatio voidaan sopivalla kaiuttimien sijoituksella saada toimimaan mihin suuntaan tahansa. Kuulijan koko ympäristön kattaminen vaatii tosin vähintään kaikki auralisaattoriin liitettävissä olevat kahdeksan kaiutinta. Auralisaation tarkkuus riippuu myös kaiuttimien sijoituksesta ja vaihtelee niin, että lähelle kaiuttimia auralisoidut virtuaaliset lähteet ovat terävimpiä.

Auralisaation monikanavakuuntelua selkeyttävä vaikutus on kiistaton. Vedena-laisessa kuuntelussa hyvissä sääolosuhteissa ja hiljaisen liikenteen aikana jopa kaikkien kahdeksan kanavan samanaikainen kuuntelu on mahdollista. Ilman au-

ralisaatiota taas jo kolmannen kanavan summaaminen kuulokkeisiin häiritsee eri kanavien erottumista toisistaan. Binauraalinen kuulokeauralisaatio vaikuttaa hyvältä menetelmältä vesikuuntelun tarpeisiin, koska se ei aseta juuri mitään vaatimuksia kuuntelutilalle. Kuuntelu kohtuullisen meluisassa ympäristössäkin on mahdollista käyttämällä ääntä eristäviä kuulokkeita.

Binauraalisen auralisaation toimintaa olisi mahdollista parantaa jonkin verran käyttämällä parempia HRTF:iä. Laitteistoa on kokeiltu myös TKK:n akustiikan laboratoriossa mitatuilla Cortex-keinopään HRTF-vasteilla [Riederer, 1998]. Näillä auralisaatio tuntuu toimivan tarkemmin. Erityisesti äänilähteen korkeuskulma on paremmin havaittavissa. Vielä keinopäävasteita parempiin tuloksiin voitaisiin päästä käyttämällä henkilökohtaisia HRTF:iä.

Parempi auralisaatio olisi mahdollista saavuttaa myös käyttämällä HRTF:ien kuvaukseen tarkempia suodattimia. Tämä vaatisi kuitenkin tehokkaamman prosessorin käyttöä tai laskentatehon lisäämistä muilla menetelmillä. Vaihtoehtoisesti auralisoitavien kanavien määrää voitaisiin vähentää tai käytettävää näytteenotto-taajuutta pienentää. Näytteenottotaajuuden pienentäminen kuitenkin kaventaisi auralisaattorin taajuuskaistaa ja suurentaisi taajuusriippumattomien viiveiden toteutuksessa syntyvää ITD-virhettä, ellei viiveitä toteutettaisi murtoviivesuodattimilla.

Vielä eräs virtuaalisten lähteiden lokalisaatiota parantava ratkaisu olisi pään liikkeitä seuraavan laitteen yhdistäminen auralisaattoriin. Tällaisen “head trackerin” avulla voitaisiin kompensoida kuulijan pään liikkeitä niin, että virtuaaliset lähteet eivät liikkuisi pään mukana. Pään liikuttelun avulla olisi mahdollista päästä eroon suuresta osasta etu-takasekaannuksia. Head trackerin toteutus vaatisi kuitenkin lisää laitteistosuunnittelua ja laskentakapasiteettia.

7 Yhteenveto

Tässä työssä on suunniteltu ja toteutettu SHARC-signaaliprosessoria käyttävä reaaliaikainen kahdeksankanavainen auralisaattori. Vaihtoehtoisina auralisaatiomenetelminä laitteistossa voidaan käyttää binauraalista kuuloke- tai kaiutinauralisaatiota sekä monikanavaista kaksi- tai kolmiulotteista amplitudipanorointia. Auralisaattori on suunniteltu ensisijaisesti parantamaan eri kanavien signaalien erottuvuutta monikanavakuuntelussa.

Binauraalinen auralisaatio tapahtuu suodattamalla sisääntulosignaalit mitattuja HRTF:iä approksimoivilla suodattimilla. Binauraaliseen auralisaatioon tarvittava ohjelmisto on kirjoitettu kokonaisuudessaan SHARCin assembler-kielellä. Suodatinsuunnittelun pohjana on käytetty KEMAR-keinopään HRTF-vasteita. Suodattimet on toteutettu taajuusriippumattoman viiveen ja minimivaiheisen FIR-suodattimen avulla. Eri kanavien binauraalisen synteessin lopputuloksena saatavat signaalit summataan yhteen ja reititetään vaihtoehtoisesti kuulokekompensointi- tai ristiinkuulumisenestosuodattimien läpi, tai tarvittaessa suoraan laitteen ulostuloon.

Kuulokekompensoinnin avulla voidaan ottaa huomioon kuulokkeiden epätasainen taajuusvaste. Auralisaattoriin on sisällytetty käänteissuodattimet kuulokkeissa yleisesti käytettäviä vapaan- ja diffuusinkentänekvalisointeja varten. Lisäksi kolmelle kuulokemallille on mittaustulosten perusteella laskettu omat käänteissuodattimet. Kuulokekompensointi on toteutettu molemmille korville yhteisillä FIR-suodattimilla.

Binauraalisessa kaiutinauralisaatiossa tarvittava ristiinkuulumisen kumoaminen tapahtuu sekoitinrakenteella toteutettujen IIR-suodattimien avulla. Suodattimet on suunniteltu HRTF-mittausten pohjalta $\pm 30^\circ$ suunnassa kuulijan edessä oleville stereokaiuttimille. Binauraalinen kaiutinauralisaatio toimii ainoastaan pienellä kuuntelualueella ja on hyvin herkkä kuunteluympäristön akustiikan vaikutukselle. Menetelmällä voidaan kuitenkin teoriassa kattaa kaikki auralisaatiosuunnat kahden kaiuttimen avulla.

Amplitudipanorointi rajoittaa käytettävissä olevat auralisaatiosuunnat kaiuttimien väliseen alueeseen. Amplitudipanorointi toimii kuitenkin suuremmalla kuuntelualueella ja huonommissa akustisissa olosuhteissa kuin binauraalinen kaiuti-

auralisaatio. Toisin kuin binauraalisessa tekniikassa, amplitudipanoroinnissa ei pyritä tuottamaan kuulijan korviin tarkalleen tietyssä suunnassa sijaitsevan äänilähteen aiheuttamia signaaleja, vaan virtuaalisia lähteitä simuloidaan toistamalla sama signaali monesta kaiuttimesta eri voimakkuudella. Amplitudipanorointi on toteutettu C-kielisellä VBAP-algoritmillä.

Auralisaation ohjaukseen voidaan käyttää joko sitä varten suunniteltua painokytkimistä ja nestekidenäytöstä koostuvaa käyttöliittymää tai PC:tä, johon auralisaattori voidaan yhdistää sarjaportin välityksellä. PC-käyttöliittymän avulla käyttäjän on myös mahdollista ladata auralisaattoriin omia kuulokekompensointi- tai HRTF-suodattimia.

Kirjoittaja on tehnyt kaiken työssä tarvittavan suodatinsuunnittelun KEMAR-keinopään HRTF-vasteiden pohjalta, sekä ohjelmoinut tarvittavat digitaaliset suodattimet assembler-kielellä. Ristiinkuulumisen estoa epäideaalisissa olosuhteissa on kuuntelukokeiden perusteella parannettu muokkaamalla käytettäviä suodattimia. Kirjoittaja on lisäksi suunnitellut ja toteuttanut kaiken auralisaation reaaliaikaisessa ohjauksessa tarvittavan laitteiston ja ohjelmiston, mukaanlukien PC:lle kirjoitetun ohjausohjelman. VBAP-algoritmin toteutuksessa kirjoittajan osuus rajoittuu ohjelmiston siirtämiseen signaaliprosessoriympäristöön.

Viitteet

- [Analog Devices, 1994] Analog Devices, Inc. (1994). *ADSP-21000 Family Applications Handbook Volume 1*.
- [Analog Devices, 1997a] Analog Devices, Inc. (1997a). *ADSP-2106x SHARC EZ-KIT Lite TM Reference Manual*.
- [Analog Devices, 1997b] Analog Devices, Inc. (1997b). *ADSP-2106x SHARC TM User's Manual*, second edition.
- [Analog Devices, 1998] Analog Devices, Inc. (1998). *VisualDSP User's Guide & Reference*.
- [Analog Devices, 1999] Analog Devices, Inc. (1999). *SHARC ADSP-21000 Family JTAG EZ-ICE User's Guide & Reference*.
- [Analog Devices, 2000a] Analog Devices, Inc. (2000a). <http://www.analog.com/>.
- [Analog Devices, 2000b] Analog Devices, Inc. (2000b). Update to SHARC EZ-KIT Lite message packet protocols. http://www.analog.com/techsupt/application_notes/EE_notes/pdf_files/ee_87.pdf. Engineer to Engineer Note EE-87.
- [Bauer, 1961] Bauer, B. B. (1961). Phasor analysis of some stereophonic phenomena. *Stereophonic Techniques*, pages 146–149. Audio Engineering Society, New York. 1986.
- [Begault, 1994] Begault, D. R. (1994). *3-D Sound for Virtual Reality and Multimedia*. Academic Press, Cambridge, Massachusetts.
- [Bernfeld, 1973] Bernfeld, B. (1973). Attempts for better understanding of the directional stereophonic listening mechanism. *AES 44th Convention*, Rotterdam, The Netherlands.
- [Blauert, 1997] Blauert, J. (1997). *Spatial Hearing*. The MIT Press, Cambridge, Massachusetts, revised edition.
- [Burian, 1996] Burian, C. J. (1996). LCD module technical reference (FAQ). <ftp://ftp.armory.com/pub/user/rstevew/LCD/lcdfaq.zip>.
- [Cooper ja Bauck, 1989] Cooper, D. H. ja Bauck, J. L. (1989). Prospects for transaural recording. *Journal of the Audio Engineering Society*, 37(1/2):3–19.

- [Duda ja Martens, 1998] Duda, R. O. ja Martens, W. L. (1998). Range dependence of the response of a spherical head model. *Journal of the Acoustical Society of America*, 104(5):3048–3058.
- [Gardner ja Martin, 1994] Gardner, B. ja Martin, K. (1994). HRTF measurements of a KEMAR dummy-head microphone. Technical Report 280, MIT Media Lab Perceptual Computing.
- [Grantham, 1995] Grantham, D. W. (1995). Spatial hearing and related phenomena. Moore, B. C. J., editor, *Hearing*, sivut 297–345. Academic Press.
- [Härmä et al., 2000] Härmä, A., Karjalainen, M., Savioja, L., Välimäki, V., Laine, U. K., ja Huopaniemi, J. (2000). Frequency-warped signal processing for audio applications. *AES 108th Convention*, Paris, France. Preprint 5171.
- [Huopaniemi, 1999] Huopaniemi, J. (1999). *Virtual Acoustics and 3-D Sound in Multimedia Signal Processing*. PhD thesis, Helsinki University of Technology.
- [Huopaniemi ja Karjalainen, 1997] Huopaniemi, J. ja Karjalainen, M. (1997). Review of digital filter design and implementation methods for 3-D sound. *AES 102nd Convention*, Munich, Germany. Preprint 4461.
- [Huopaniemi et al., 1999] Huopaniemi, J., Zacharov, N., ja Karjalainen, M. (1999). Objective and subjective evaluation of head-related transfer function filter design. *Journal of the Audio Engineering Society*, 47(4):218–239.
- [Ifeachor ja Jervis, 1993] Ifeachor, E. C. ja Jervis, B. W. (1993). *Digital Signal Processing: A Practical Approach*. Addison-Wesley, U.S.
- [Jot et al., 1995] Jot, J.-M., Larcher, V., ja Warusfel, O. (1995). Digital signal processing issues in the context of binaural and transaural stereophony. *AES 98th Convention*, Paris, France. Preprint 3980.
- [Karjalainen, 1999] Karjalainen, M. (1999). *Kommunikaatioakustiikka*. Teknillinen korkeakoulu. Akustiikan ja äänenkäsittelytekniikan laboratorio. Raportti 51. Esipainos.
- [Kendall, 1995] Kendall, G. S. (1995). A 3-D sound primer: Directional hearing and stereo reproduction. *Computer Music Journal*, 19(4):23–46.
- [Kirkeby et al., 1998] Kirkeby, O., Nelson, P. A., ja Hamada, H. (1998). Local sound field reproduction using two closely spaced loudspeakers. *Journal of the Acoustical Society of America*, 104(4):1973–1981.

- [Kistler ja Wightman, 1992] Kistler, D. J. ja Wightman, F. L. (1992). A model of head-related transfer functions based on principal components analysis and minimum-phase reconstruction. *Journal of the Acoustical Society of America*, 91(3):1637–1647.
- [Kleiner et al., 1993] Kleiner, M., Dalenbäck, B.-I., ja Svensson, P. (1993). Auralization—an overview. *Journal of the Audio Engineering Society*, 41(11):861–875.
- [Kulkarni ja Colburn, 2000] Kulkarni, A. ja Colburn, H. S. (2000). Variability in the characterization of the headphone transfer-function. *Journal of the Acoustical Society of America*, 107(2).
- [Kulkarni et al., 1999] Kulkarni, A., Isabelle, S. K., ja Colburn, H. S. (1999). Sensitivity of human subjects to head-related transfer-function phase spectra. *Journal of the Acoustical Society of America*, 105(5).
- [Laakso et al., 1996] Laakso, T. I., Välimäki, V., Karjalainen, M., ja Laine, U. K. (1996). Splitting the unit delay – tools for fractional delay filter design. *IEEE Signal Processing Magazine*, 13(1):30–60.
- [Lapsley et al., 1995] Lapsley, P., Bier, J., Shoham, A., ja Lee, E. A. (1995). *DSP Processor Fundamentals*. IEEE Press, New York.
- [Lehnert ja Blauert, 1991] Lehnert, H. ja Blauert, J. (1991). Virtual auditory environment. *91 ICAR. Fifth International Conference on Advanced Robotics. Robots in Unstructured Environments*, New York, USA.
- [Maijala, 1999] Maijala, P. (1999). Binauraalisen teknologian käyttö kuuntelujärjestelmissä. Maanpuolustuksen tieteellinen neuvottelukunta. Raporttisarja A.
- [Mathworks, 1994] Mathworks, Inc. (1994). *Signal Processing Toolbox User's Guide*.
- [Mathworks, 1996] Mathworks, Inc. (1996). *Using MATLAB*.
- [Merimaa, 2000] Merimaa, J. (2000). S-88.128 Tietotekniikan erikoistyö: SHARC EZ-KIT Lite -signaaliprosessorikortin käyttöliittymälajennus. Teknillinen korkeakoulu. Signaalinkäsittelytekniikan laboratorio.
- [Microsoft, 1997] Microsoft (1997). *Visual C++ User's Guide*.

- [Møller et al., 1996a] Møller, H., Jensen, C. B., Hammershøi, D., ja Sørensen, M. F. (1996a). Using a typical human subject for binaural recording. *AES 100th Convention*, Copenhagen, Denmark. Preprint 4157.
- [Møller et al., 1997] Møller, H., Jensen, C. B., Hammershøi, D., ja Sørensen, M. F. (1997). Evaluation of artificial heads in listening tests. *AES 102nd Convention*, Munich, Germany. Preprint 4404.
- [Møller et al., 1996b] Møller, H., Sørensen, M. F., Jensen, C. B., ja Hammershøi, D. (1996b). Binaural technique: Do we need individual recordings. *Journal of the Audio Engineering Society*, 44(6):451–469.
- [Oppenheim ja Schafer, 1975] Oppenheim, A. V. ja Schafer, R. W. (1975). *Digital Signal Processing*. Prentice Hall, Englewood Cliffs, New Jersey.
- [Pulkki, 1997] Pulkki, V. (1997). Virtual sound source positioning using vector base amplitude panning. *Journal of the Audio Engineering Society*, 45(6):456–466.
- [Pulkki, 1999a] Pulkki, V. (1999a). 3D and 2D VBAP implementation for SGI workstations with one 8-channel ADAT card. Version 0.1. http://www.acoustics.hut.fi/~ville/software/VBAP_demo/.
- [Pulkki, 1999b] Pulkki, V. (1999b). Uniform spreading of amplitude panned virtual sources. *Proc. 1999 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, USA.
- [Pulkki et al., 1999] Pulkki, V., Karjalainen, M., ja Välimäki, V. (1999). Localization, coloration, and enhancement of amplitude-panned virtual sources. *Proceedings of the AES 16th International Conference*, pages 257–278, Rovaniemi, Finland.
- [Pulkki ja Lokki, 1998] Pulkki, V. ja Lokki, T. (1998). Creating auditory displays with multiple loudspeakers using VBAP: A case study with DIVA project. *International Conference on Auditory Display*, Glasgow, England.
- [Riederer, 1998] Riederer, K. A. J. (1998). Head-related transfer function measurements. Master’s thesis, Helsinki University of Technology.
- [Saberri et al., 1991] Saberri, K., Dostal, L., Sadralodabai, T., Bull, V., ja Perrot, D. R. (1991). Free-field release from masking. *Journal of the Acoustical Society of America*, 90(3):1355–1370.

- [Sunier ja Sunier, 2000] Sunier, J. ja Sunier, D. (2000). The binaural source FAQ page. <http://www.binaural.com/binfaq.html>.
- [Theile, 1986] Theile, G. (1986). On the standardization of the frequency response of high-quality studio headphones. *Journal of the Audio Engineering Society*, 34(12).
- [Theile ja Plenge, 1977] Theile, G. ja Plenge, G. (1977). Localization of lateral phantom sources. *Journal of the Audio Engineering Society*, 25(4):196–200.
- [Wenzel et al., 1993] Wenzel, E. M., Arruda, M., Kistler, D. J., ja Wightman, F. L. (1993). Localization using nonindividualized head-related transfer functions. *Journal of the Acoustical Society of America*, 94(1):111–123.
- [Wenzel ja Foster, 1993] Wenzel, E. M. ja Foster, S. H. (1993). Perceptual consequences of interpolating head-related transfer functions during spatial synthesis. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 102–105.
- [Wightman ja Kistler, 1999] Wightman, F. L. ja Kistler, D. J. (1999). Resolution of front-back ambiguity in spatial hearing by listener and source movement. *Journal of the Acoustical Society of America*, 105(5):2841–2853.
- [Zhang et al., 1998] Zhang, M., Tan, K.-C., ja Er, M. H. (1998). Three-dimensional sound synthesis based on head-related transfer functions. *Journal of the Audio Engineering Society*, 46(10):836–844.

A Esimerkkejä lähdekoodista

Koko auralisaattorin lähdekoodi sisältää n. 3300 riviä assembler- sekä 500 riviä C-lähdekoodia. Kernel-ohjelman osuus tästä on n. 1300 riviä. Melko suuri osa muusta assembler-koodista sisältää auralisaattorin ohjauksessa tarvittavia päätösketjuja ja haarautumisia. Tärkeimmät signaalinkäsittelyoperaatiot ovat toteutukseltaan varsin tiiviitä.

Seuraavassa on esitetty joitakin osia auralisaattorin ohjelmistosta. Kappaleessa A.1 esitetään HRTF-suodatuksessa tarvittavien kahden taajuusriippumattoman viiveen ja minimivaiheisen suodattimen toteutus ja kappaleessa A.2 esitetään ristiinkuulumisen eston IIR-suodattimien lähdekoodi. IIR-suodattimien tehokasta toteutusta SHARCilla on käsitelty lähteessä [Analog Devices, 1994, s. 100–110]. HRTF-suodattimien rutiini ajetaan kahdeksan kertaa joka näytejaksolla.

Kappaleessa A.3 esitetään näytteiden käsittelyn VBAP-algoritmissa suorittava rutiini. Lähdekoodi suorittaa kaikki näytteiden käsittelyyn keskeytyksen aikana tarvittavat toiminnot. Uudet panorointikertoimet lasketaan aina tarvittaessa erillisen C-kielisen algoritmin avulla [Pulkki, 1999a]. Esitetty lähdekoodi on esimerkeistä kaikkein pisin ja vähiten optimoitu, mutta sen suorittamiseen kuluu silti käytettävissä olevista 900 kellojaksosta alle 250.

A.1 HRTF-suodatin

```
hrtf_filter:
/*      HRTF FIR-filter
      Filtering parameters:
          dm(i0) = delay line
          pm(i8) = HRTFs, left and right coefficients interlaced
          f8 = left output
          f9 = right output
          m1 = m8 = 1;

      Registers used:
          f0 = left delay line sample
          f1 = right " " "
          f4 = current filter coefficient
          f8 = left accumulated product
          f9 = right " " "
          f12 = left temporary product
```



```

        f13 = right      "      "
        dm(i1) = delayed left input
        dm(i2) = delayed right input
        m2 = left HRTF delay (in samples)
        m3 = right HRTF delay

Return values:
        f8 = left previous output + new output
        f9 = right      "      "      "      " */

/* frequency independent delays */
        i1 = i0;                /* left delay line pointer */
        m2 = pm(i8, m8);        /* load delay length */
        i2 = i0;                /* right delay line pointer */
        m3 = pm(i8, m8);
        modify(i1, m2);        /* move index pointers */
        modify(i2, m3);

/* load first samples and coefficient and clear registers */
        r12 = r12 xor r12, f0 = dm(i1, m1);
        r13 = r13 xor r13, f1 = dm(i2, m1), f4 = pm(i8, m8);

        lcntr = TAPS - 1, do fmacs until lce;
        /* left */
        f12 = f0 * f4, f8 = f8 + f12, f0 = dm(i1, m1), f4 = pm(i8, m8);
        /* right */
fmacs:    f13 = f1 * f4, f9 = f9 + f13, f1 = dm(i2, m1), f4 = pm(i8, m8);

        /* last macs */
        f12 = f0 * f4, f8 = f8 + f12, f4 = pm(i8, m8);
        rts (db), f8 = f8 + f12;
        f13 = f1 * f4, f9 = f9 + f13;
        f9 = f9 + f13;

```

A.2 Ristiinkuulumisen esto

```

crosstalk:
/* Cascade of canonical form 2nd order IIR filters
One biquad:
        w(n) = x(n) + a1*w(n-1) + a2*w(n-2)
        y(n) = w(n) + b1*w(n-1) + b2*w(n-2)

Filtering parameters:
        dm(i3) = index to w(n) delay line
        dm(i4) = index to w(n) delay line
        pm(i11) = coefficients sum: a21, a11, b21, b11, a22, a12,..., gain
                                dif: a21, a11, b21, b11, a22, a12,..., gain

        f8 = left input
        f9 = right input

```

```

        m1 = m8 = 1

Return values:
        f8 = left cross talk canceled signal
        f9 = right " " " " */

        f8 = f9 + f8, f9 = f9 - f8; /* shuffler structure */
        r12 = r12 xor r12, f2 = dm(i3, m1), f4 = pm(i11, m8);
/* sum filter biquads */
        lcntr = BIQUADS, do quads1 until lce;
        /* a2*w(n-2), x(n)+0 or y(n), w(n-1), a1 */
        f12 = f2 * f4, f8 = f8 + f12, f3 = dm(i3, m1), f4 = pm(i11, m8);
        /* a1*w(n-1), x(n)+[a2*w(n-2)], new w(n-2), b2 */
        f12 = f3 * f4, f8 = f8 + f12, dm(i4, m1) = f3, f4 = pm(i11, m8);
        /* b2*w(n-2), new w(n), w(n-2), b1 */
        f12 = f2 * f4, f8 = f8 + f12, f2 = dm(i3, m1), f4 = pm(i11, m8);
        /* b1*w(n-1), w(n)+[b2*w(n-1)], new w(n-1), a2 for next biquad */
quads1:    f12 = f3 * f4, f8 = f8 + f12, dm(i4, m1) = f8, f4 = pm(i11, m8);
        f8 = f8 + f12;
        f8 = f8 * f4; /* cascade gain */

        r12 = r12 xor r12, f4 = pm(i11, m8);
/* difference filter biquads, works like sum biquads */
        lcntr = BIQUADS, do quads2 until lce;
        f12 = f2 * f4, f9 = f9 + f12, f3 = dm(i3, m1), f4 = pm(i11, m8);
        f12 = f3 * f4, f9 = f9 + f12, dm(i4, m1) = f3, f4 = pm(i11, m8);
        f12 = f2 * f4, f9 = f9 + f12, f2 = dm(i3, m1), f4 = pm(i11, m8);
quads2:    f12 = f3 * f4, f9 = f9 + f12, dm(i4, m1) = f9, f4 = pm(i11, m8);
        f9 = f9 + f12, modify(i3, m0);
        jump ch_out (db);
        f9 = f9 * f4; /*cascade gain */
        f9 = f8 + f9, f8 = f8 - f9; /* shuffler structure */

```

A.3 VBAP

```

vbap:
/* VBAP panning of input samples
Parameters:
        none: gets its parameters from global variables */

/* switch to alternate registers */
bit set mode1 SRRFL | SRRFH | SRD1L | SRD1H | SRD2L | SRD2H;
push sts; /* sts is not pushed automatically on spr0I interrupt */
i4 = output;
i5 = vbap_gain;
i6 = vbap_ls;
r0 = r0 xor r0; /* zero outputs */
dm(output) = r0;
dm(output + 1) = r0;

```

```
dm(output + 2) = r0;
dm(output + 3) = r0;
dm(output + 4) = r0;
dm(output + 5) = r0;
dm(output + 6) = r0;
dm(output + 7) = r0;

r0 = dm(rx_buf + 1);    /* codec 0, left channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain);
f0 = f0 * f1;

r0 = dm(rx_buf + 2);    /* codec 0, right channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain + 1);
f0 = f0 * f1;

r0 = dm(rx_buf + 4);    /* codec 1, left channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain + 2);
f0 = f0 * f1;

r0 = dm(rx_buf + 5);    /* codec 1, right channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain + 3);
f0 = f0 * f1;

r0 = dm(rx_buf + 7);    /* codec 2, left channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain + 4);
f0 = f0 * f1;

r0 = dm(rx_buf + 8);    /* codec 2, right channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain + 5);
f0 = f0 * f1;

r0 = dm(rx_buf + 10);   /* codec 3, left channel */
f0 = float r0;
call pan (db);
f1 = dm(channel_gain + 6);
f0 = f0 * f1;

r0 = dm(rx_buf + 11);   /* codec 3, right channel */
f0 = float r0;
```

```
    call pan (db);
    f1 = dm(channel_gain + 7);
    f0 = f0 * f1;

    f0 = dm(output);          /* convert to integer */
    r0 = fix f0;
    dm(tx_buf + 1) = r0;     /* and write to DAC */
    f0 = dm(output + 1);
    r0 = fix f0;
    dm(tx_buf + 2) = r0;
    f0 = dm(output + 2);
    r0 = fix f0;
    dm(tx_buf + 4) = r0;
    f0 = dm(output + 3);
    r0 = fix f0;
    dm(tx_buf + 5) = r0;
    f0 = dm(output + 4);
    r0 = fix f0;
    dm(tx_buf + 7) = r0;
    f0 = dm(output + 5);
    r0 = fix f0;
    dm(tx_buf + 8) = r0;
    f0 = dm(output + 6);
    r0 = fix f0;
    dm(tx_buf + 10) = r0;
    f0 = dm(output + 7);
    r0 = fix f0;
    dm(tx_buf + 11) = r0;

    rti (db);
    pop sts;
    bit clr mode1 SRRFL | SRRFH | SRD1L | SRD1H | SRD2L | SRD2H;

pan:
/* The actual panning
Parameters:
    f0 = new sample
    i4 = output vector
    i5 = address to VBAP gain vector
    i6 = address to VBAP loudspeaker vector */

    f1 = dm(i5, 1);          /* first loudspeaker VBAP gain */
    f1 = f0 * f1, r2 = dm(i6, 1); /* loudspeaker number */
    m4 = r2;
    f2 = dm(m4, i4);        /* sum new samples to the loudspeaker's output */
    f1 = f1 + f2;
    dm(m4, i4) = f1;

    f1 = dm(i5, 1);        /* second loudspeaker */
    f1 = f0 * f1, r2 = dm(i6, 1);
```

```
m4 = r2;
f2 = dm(m4, i4);
f1 = f1 + f2;
dm(m4, i4) = f1;

f1 = dm(i5, 1);          /* last loudspeaker */
f1 = f0 * f1, r2 = dm(i6, 1);
m4 = r2;
f2 = dm(m4, i4);
rts (db);
f1 = f1 + f2;
dm(m4, i4) = f1;
```