

Kuulon malleihin perustuva audiokoodaus

Lähteet: Kahrs, Brandenburg, (Editors). (1998). "Applications of digital signal processing to audio and acoustics". Kluwer Academic.
Bernd Edler. (1997). "Low bit rate audio tools". MPEG meeting.



Sisältö:

- Johdanto
 - vaatimuksia koodekeille
 - lähdekoodaus vs. kuulon malleihin perustuva k.
 - koodauslaadun mittaaminen
- Faktoja psykoakustiikasta
- Kuulon malleihin perustuvan koodauksen kokonaiskuva
- Selostus koodaustyökaluista
 - suodinpankit
 - kuulon mallit
 - kvantisointi ja koodaus
 - stereokoodaus
- Todellisia koodausjärjestelmiä
 - MPEG-1, MPEG-2 BC
 - MPEG-2 AAC
 - Vertailua
 - MPEG-4

1 Johdanto

- Tiedonsiirron kaistanleveys kasvaa jatkuvasti, mutta kaistan tarve kasvaa vielä enemmän
 - tarve kompressiotekniikoille
- Audiokoodauksen sovelluksia
 - digitaaliset radiolähetykset
 - audioraidat digitaalisen videon yhteydessä (mm. digi-TV ja DVD)
 - musiikin tallennus ja kovalevyäänitys, kannettavat soittimet
 - audion siirto Internetissä ja modeemi/ISDN/ADSL-linjan yli sekä langattomasti

Johdanto

1.1 Vaatimuksia audiokoodekille

- Kompression tehokkuus: äänenlaatu vs. bittinopeus
- Absoluuttinen saavutettavissa oleva laatu
 - yleensä vaatimuksena on:
"säätämällä bittinopeuden riittäväksi dekodatusta signaalissa ei enää kuule eroa originaaliin signaaliin CD:llä tai DAT-nauhalla"
- Kompleksisuus
 - laskennallinen kompleksisuus: päätekijä kotitietokoneilla
 - muistivaatimukset: päätekijä dedikoiduille piireille suunnitellussa
 - enkooderin vs. dekodeerin kompleksisuus
 - enkooderi on yleensä huomattavasti kompleksisempi
 - enkoodaus voidaan joissain sovelluksissa tehdä off-line, eli ei reaaliajassa

Johdanto

Vaatimuksia audiokoodekille

- Algoritminen viive
 - riippuen sovelluksesta, viive joko on tai ei ole tärkeä kriteeri
 - erittäin tärkeä kaksisuuntaisessa kommunikaatiossa (~20 ms vielä ok)
 - ei lainkaan tärkeä puhtaasti tallennussovelluksessa, koodekin ei tällöin tarvitse olla edes kausaalinen (järjestyksessä ajan yli toimiva)
 - jonkin verran tärkeä digitaalisessa radio/TV-lähetyskäsä (~ 100 ms)
- Muokattavuus
 - tietty kohta alkuperäisestä audiosignaalista voidaan poimia esiin *koodatusta* bittivirrasta
 - vaatii että dekodaus voi alkaa mistä tahansa kohdasta bittivirtaa
- Virhesietoisuus
 - herkkyys yksittäisille tai purskevirheille siirtokanavassa
 - virhesietoisuus voidaan saavuttaa virheenkorjauskoodeilla, mutta se vaatii redundanssin (ja siten bittinopeuden) lisäämistä

1.2 Lähdekoodaus vs. kuulon malleihin perustuva koodaus

- Yleensä signaalit siirtyä tietyn laatusina, mutta ei täysin identtisinä originaaliin signaaliin verrattuna
- Kompressiota voidaan saavuttaa poistamalla
 - redundantti informaatio, joka voidaan rekonstruoida vastaanottimessa
 - irrelevantti informaatio, jolla ei ole merkitystä vastaanottajalle
- **Lähdekoodaus:** painopiste redundantin informaation poistossa
 - puheenkoodaus: malli ihmisen ääniväylästä määrittelee *mahdolliset signaalit*, jolloin vain ääniväylän parametrit tarvitsee siirtää
 - toimii varsin heikosti geneerisessä audiokoodauksessa
 - mitkä tahansa signaalit ovat mahdollisia, ja niitä voidaan jopa kutsua musiikiksi
- **Kuulon malleihin perustuva koodaus:** painopiste kuulon kannalta irrelevantin informaation poistossa
 - minimoidaan koodauksessa syntyvien vääristymien kuultavuus

Lähdekoodaus vs. kuulon malleihin perustuva koodaus

- Puhe- ja audiosignaalit ovat koko lailla erilaisia
 - koodauksen yhteydessä sanaa "audiosignaali" käytetään yleensä viittaamaan signaaleihin, jotka eivät ole puhetta, vaikka puhesignaalit tietysti ovat audiota (äänisignaaleja) nekin
- Tyypillisesti audiosignaalien
 - näytteistystaajuus on korkeampi
 - dynaaminen alue on laajempi
 - tehospektri vaihtelee enemmän
 - korkea laatu on kuulijalle keskeisempi kuin puhesignaalien
 - stereo- ja monikavanaäänentoisto tulee kyseeseen
- Lähdekoodaus on edistynyt nopeasti
 - puhesignaaleille saavutetaan hyvä koodauslaatu huomattavasti matalammilla bittinopeuksilla kuin audiosignaaleille

1.3 Koodauslaadun mittaaminen

- Signaalien häviöllinen koodaus aiheuttaa väistämättä jonkin verran vääristymiä alkuperäiseen signaaliin
- Vääristymän määrää voidaan mitata
 - subjektiivisilla kuuntelutesteillä, käyttäen esimerkiksi *mean opinion score (MOS)* –mittaa aistimukselliseen samanlaisuuden mittana
 - tärkein ja luotettavin tapa arvioida koodauslaatua
 - yksinkertaisilla objektiivisilla kriteereillä, kuten signaali-kohina – suhde mittana alkuperäisen ja rekonstruoidun signaalin välillä
 - komplekseilla kriteereillä, kuten objektiivisilla aistimuksellisen samankaltaisuuden mittareilla, jotka huomioivat ihmisen kuuloaistin tunnetut ominaisuudet (esim. peittoilmiön)
- Mean opinion score (MOS) –mitta
 - koehenkilöt arvostelevat koodatun äänen *N*-askeleisella skaalalla
 - kaksi adjektivein määriteltyä viisiaskelista skaalaa on käytössä (signaalin laatua / kahden signaalin erilaisuutta kuvailevat)
 - testin lopullinen tulos (MOS) on keskiarvo koehenkilöiden arvioista

Koodauslaadun mittaaminen

- MOS on laajalti käytetty. Sillä on kuitenkin myös haittoja
 - tulos vaihtelee ajan ja koehenkilöiden mukaan
 - täysin samaa testitulosta on vaikea toistaa
 - tulos riippuu vahvasti valituista testisignaaleista
 - tyypillinen audiomateriaali vs. kriittiset testisignaalit
- ISO / MPEG –testit
 - kolme signaalia A, B, C
 - A on alkuperäinen (=referenssi), B ja C ovat referenssi ja testattava järjestelmä
 - koehenkilöt eivät tiedä B- ja C-signaaleista kumpi on kumpi
 - koehenkilöiden täytyy päättää onko B vai C referenssi, ja arvostella sitten toisen signaalin laatu
- Muitakin yllä mainitun tyyppisiä testejä on
 - esim. ABX-testi: A on referenssi, B on testisignaali, X on A tai B
 - A:n ja B:n välillä on eroa, mikäli koehenkilö pystyy tunnistaa X:n

2 Faktoja psykoakustiikasta (kertausta)

- Pääkysymys kuulomalleihin perustuvassa koodauksessa:
 - kuinka paljon kohinaa signaalin voidaan lisätä ilman että se on kuultavissa
- Vastaus löytyy *psykoakustiikasta*
 - psykoakustiikka tutkii suhteita akustisten tapahtumien ja niiden tuottamien kuuloaistimusten välillä
- Tärkein avainsana koodauksessa on ”*peittoilmiö*”
 - = maskaus (engl. masking)
 - peittoilmiö kuvaa tilannetta jossa hiljaisempi mutta selvästi kuultava signaali muuttuu kuulumattomaksi äänekkäämmän signaalin vaikutuksesta
 - peittovaikutus riippuu sekä äänien spektrirakenteesta että niiden vaihtelusta ajan funktiona

Faktoja psykoakustiikasta

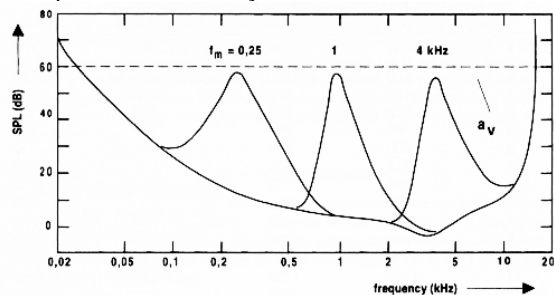
2.1 Peitto taajuustasossa

- Malli ihmisen kuulojärjestelmän taajuusanalyysistä
 - taajuustaso voidaan jakaa kriittisiin kaistoihin
 - peittovaikutus on voimakas, mikäli taajuuskomponentit ovat lähempänä kuin kriittisen kaistan etäisyydellä toisistaan
 - Bark asteikko: taajuusasteikko joka saadaan täyttämällä kuuloalue vierekkäisillä kriittisillä kaistoilla, ja numeroimalla ne 1:stä 26:een
- Kapeakaistainen kohinasignaali kykenee peittämään siniääneksen tehokkaammin (pienemmällä äänipainetasolla) kuin siniäänes kohinan
- *Peittokynnys* tarkoittaa peittoäänen aiheuttamaa noususta kuulokynnystasoa
 - ääniä, joiden taso jää peittokynnyksen alapuolelle ei kuulla
 - peittokynnys hiljaisuudessa = kuulokynnys

Faktoja psykoakustiikasta

Peitto taajuustasossa

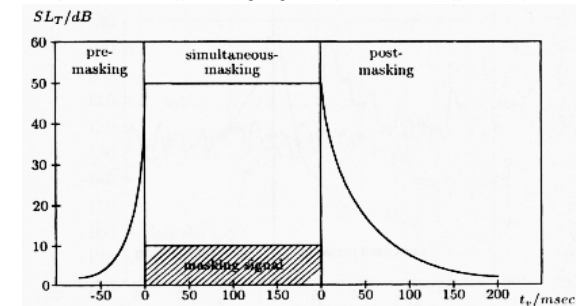
- Kuva: peittokynnyksiä [Herre95]
 - peittoääni: kapeakaistaista kohinaa 250 Hz, 1 kHz, 4 kHz
 - leviämiskäyrä: peittovaikutus ulottuu peittoäänen lähiympäristöön
 - leviää enemmän kohti korkeita taajuuksia
- Peiton additiivisuus (kompleksit signaalit)
 - yhteinen peittovaikutus on yleensä *enemmän* kuin osien summa



Faktoja psykoakustiikasta

2.2 Peitto aikatasossa

- *Eteenpäin* peitto (= jälkipeitto, post-masking)
 - signaalin peittovaikutus ulottuu ajanhetkiin sen loppumisen *jälkeen*
- *Taaksepäin* peitto (= esipeitto, pre-masking)
 - signaalin peittovaikutus ulottuu ajanhetkiin *ennen* sen alkamista
- Kaaviokuva esipeitosta (pre-) ja jälkipeitosta (post-) [Sporer98]

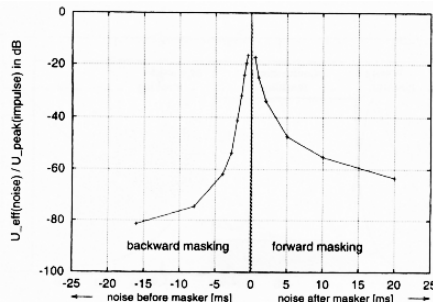


Faktoja psykoakustiikasta

Peitto aikatasossa

Audiokoodaus 13
DA / Klapuri

- Kuva: peittovaikutuksen mittaustuloksia [Spille92]
 - *aikapeitto ei ulotu kovin kauas*
 - samanaikaisten äänten taajuuspeitto on merkittävämpi ilmiö
 - peittokynnys lähestyy kynnystä hiljaisuudessa, jos aikaero signaalien välillä on enemmän kuin 16 ms, ja jo 2 ms aikaerolla peittokynnys on 25 dB alempana kuin impulssin aikana



Faktoja psykoakustiikasta

2.3 Kuuntelijoiden väliset erot

- Taustaoletus kuulomallien käytössä koodauksessa on, että ihmisten kuulokyvyissä ei ole suuria eroja
- Oletus on enemmän tai vähemmän totta
 - absoluuttinen kuulokynnys: vaihtelee jopa tietylle henkilölle eri aikoina, koodekkeiden täytyy olettaa erittäin hyvä kuulo
 - peittokynnys taajuustasossa: vaihtelut ovat melko pieniä
 - peitto aikatasossa: suuria eroja, kuuntelija voi harjaantua kuulemaan esikaikuja

2.4 Johtopäätös

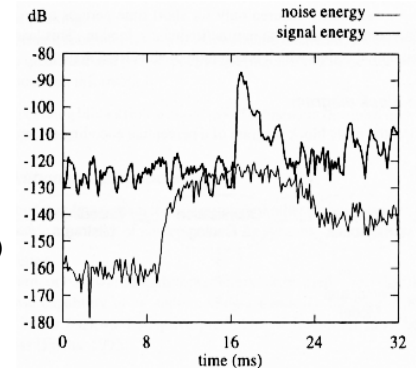
- Kuulon tutkimus ei ole millään tavalla loppuun kaluttu aihe
 - yksinkertaisia malleja voidaan rakentaa helposti, ja niillä saavutetaan jo kohtuullisen hyviä koodaustuloksia
 - kehittyneempiä koodekkeja (kuulomalleja) suunniteltaessa nykyisen psykoakustisen tietämyksen rajat tulevat hyvin nopeasti vastaan

Faktoja psykoakustiikasta

Peitto aikatasossa

Audiokoodaus 14
DA / Klapuri

- *Esikaiku* (engl. *pre-echo*): Jos koodekin tuottamat häiriöt leviävät ajallisesti siten että ne edeltävät itse signaalin tason muutosta, syntyvää kuultavaa artefaktia sanotaan esikaikuksi
 - yleinen ongelma, sillä koodekkeissa käytettävät suodinpankit aiheuttavat ajallista leviämistä
- Kuva: esimerkki esikaikusta
 - alempi käyrä (kohinasignaali) paljastaa analyysi-ikkunan muodon



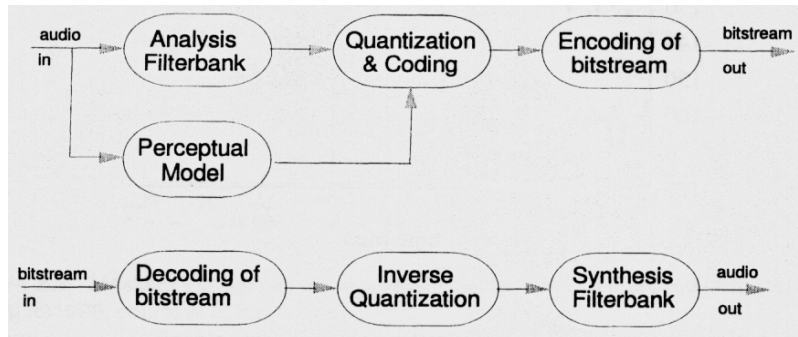
3 Kuulon malleihin perustuvan koodauksen kokonaiskuva

Audiokoodaus 16
DA / Klapuri

- *Perusajatuksena on peittää kvantisointikohina signaalista riippuvan kuulokynnyksen (peittokynnys) alle*
- Peittovaikutuksen mallintaminen
 - tärkeimmät peittovaikutukset voidaan kuvailla taajuustasossa
 - toisaalta peittovaikutuksen stationäärisyys ulottuu ajallisesti vain noin 15 ms etäisyydelle (ks. aikapeitto)
- Seuraus:
 - kuulon malleihin perustuva audiokoodaus on parhaiten tehtävissä aika-taajuustasossa
 - yhteinen koodekkeiden perusrakenne

3.1 Peruslohkokaavio

- Kuva: Lohkokaavio kuulon malleihin perustuvasta koodausjärjestelmästä
 - yllä: enkooderi
 - alla: dekooderi



Peruslohkokaavio

- Suodinpankki
 - käytetään hajottamaan sisääntuleva signaali alikaistoihin / spektrikomponentteihin aika-taajuustasossa
- Kuulomalli
 - yleensä tarkastelee sisääntulevaa signaalia, ei suodinpankin ulostuloja
 - parempi resoluutio sekä aika- että taajuustasossa
 - laskee signaaliriippuvan peittokynnyksen käyttäen psykoakustisia sääntöjä
- Kvantisointi ja koodaus
 - spektrikomponentit kvantisoidaan ja koodataan
 - tavoitteena on pitää kvantisointikohina peittokynnyksen alapuolella
- Kehysten pakkaus
 - bittivirran formatoija kokoaa bittivirran, joka tyypillisesti koostuu koodatusta äänidatasta ja oheisinformaatiosta

3.2 Lisätyökaluja

- Neljän välttämättömän osan lisäksi käytetään lukuisia muita tekniikoita parantamaan koodauksen tehokkuutta
- Adaptiivinen ennustus eteen- tai taaksepäin
 - parempi redundanssin poisto erityisesti stationäärisille signaaleille
- Ajallinen kohinan muokkaus
 - suodatetaan osia spektristä siten että voidaan kontrolloida kvantisointikohinan ajallista muotoa
- Stereokoodaus
 - M/S stereokoodaus: vaihdellaan vasen/oikea ja summa/erotus esitysmuotojen välillä
 - intensiteetti stereokoodaus: korkeilla taajuuksilla vaiheinformaatio voidaan hylätä. Lähetetään vain yksi tarkka energiaspektri ja sen ohessa skaalaustekijät kummallekin kanavalle. Käy myös n:lle kanavalle.
 - stereoennustus: yhden kanavan näytteistä ennustetaan muille kanaville
- Spektrin tasoitus
 - käytetään parantamaan kvantisoinnin ja koodauksen tehokkuutta
 - LPC-analyysillä normalisoidaan spektrin verhoikäyrä

3.3 Aistimuksellinen entropia

- Aistimuksellinen (engl. *perceptual*) entropia
 - käytetään määrittelemään pienin datanopeus, joka tarvitaan enkoodaamaan audiosignaali ilman aistittavaa eroa
 - **kuvaa signaalin kuuloaistille välttämää informaatiomäärää**
- Voidaan laskea

$$PE = \frac{1}{N} \sum_{f=f_l}^{f_u} \max \left[0, \log_2 \left(\frac{|signal(f)|}{threshold(f)} \right) \right]$$

missä N on taajuuskomponenttien määrä välillä $f_l - f_u$ (20Hz–20kHz), $signal(f)$ on taajuuskomponentin amplitudi, ja $threshold(f)$ on estimoitu peittokynnys taajuudella f

- Määritelmä tukeutuu signaaliriippuvan kuulokynnyksen (peittokynnys) laskettavuuteen

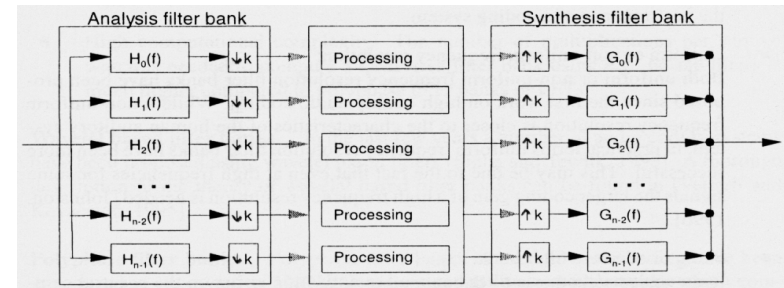
4 Selostus koodaustyökaluista

- Seuraavassa käydään yksityiskohtaisemmin läpi audiokoodekin eri osia
 - nykyisissä järjestelmissä käytössä olevia suodinpankkeja → määrää koodekin perusrakenteen
 - kuulon malleja → koodausjärjestelmän algoritminen ydin
 - kvantisointi- ja koodaustyökaluja → toteuttaa itse datan vähentämisen enkooderissa
- Lisätyökaluista käsitellään lyhyesti
 - stereokoodausta
 - ajallista ennustamista

Selostus koodaustyökaluista

4.1 Suodinpankit

- Suodinpankki määrää koodekin perusrakenteen
- Kuvan esimerkki: staattisen n -kanavaisen analyysi/synteesi suodinpankin lohkokaavio [Herre95]
 - kanavien näytteistystaajuutta lasketaan tässä kertoimella k → kaistanleveydet samoja → tasavälisen taajuuserottelun pankki
 - jos $k = n$, puhutaan *kriittisen näytteistykseen* suodinpankista



Selostus koodaustyökaluista

Suodinpankit: parametreja

- Taajuusresoluutio: kaksi päätyyppiä
 - matalan resoluution suodinpankit (esim. 32 alikaistaa), joita usein kutsutaan *alikaistakoodekeiksi*
 - tällöin yleensä kvantisoidaan ajallisesti peräkkäisissä lohkoissa
 - korkean resoluution suodinpankit (esim. 512 alikaistaa), joita usein kutsutaan *muunnoskoodekeiksi*
 - tällöin yleensä kvantisoidaan yhdistämällä vierekkäisiä taajuusviivoja
 - matemaattisesti kaikki audiokoodekeissa käytettävät muunnokset voidaan nähdä suodinpankkeina
 - periaatteellinen erottelu alikaistakoodekkien ja muunnoskoodekkien välillä ei ole järkevää
- Täydellisesti rekonstruoivat suodinpankit
 - mahdollistavat alkuperäisen signaalin häviöttömän rekonstruoinnin analyysi/synteesi järjestelmässä, mikäli kvantisointia ei käytetä
 - yksinkertaistaa koodausjärjestelmän muiden osien suunnittelua
 - yleensä käytetään joko täydellisesti tai melkein täydellisesti rekonstruoivia suodinpankkeja

Selostus koodaustyökaluista

Suodinpankit: parametreja

- Prototyyppi-ikkuna (aikatason kehyksen ikkunointi)
 - erityisesti matalilla bittinopeuksilla analyysi / synteesi prototyyppi-ikkunan ominaisuudet ovat ratkaiseva suorituskytekykijä
- Tasavälinen vai epätasavälinen taajuuserottelu
 - epätasavälinen taajuusresoluutio on lähempänä ihmiskuulon ominaisuuksia (kriittiset kaistat jakautuvat log. taajuusasteikolle)
 - käytännössä tasavälisen taajuuserottelun suodinpankit ovat olleet menestyksekkäämpiä (helpottaa teknistä toteutusta)
- Staattinen vai adaptiivinen suodinpankki
 - kvantisointivirhe leviää ajallisesti synteesi-ikkunan koko pituudelle
 - esikaiku-efekti voidaan välttää mikäli suodinpankki ei ole staattinen vaan vaihtelee eri aika-/taajuusresoluutioiden välillä
 - esimerkki: adaptiivinen ikkunan vaihtelu, jossa kytketään lyhyempään analyysi-ikkunaan transienttimaisissa muutoskohdissa

Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

Audiokoodaus 25
DA / Klapuri

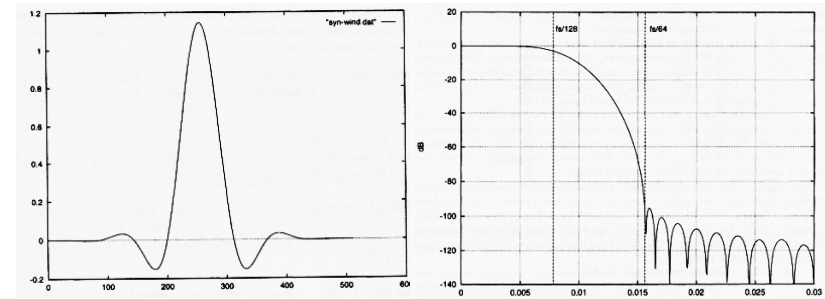
- Quadrature mirror filter (QMF) suodinpankit
 - puumainen rakenne, jossa kukin suodatin jakaa kahteen osaan
 - voidaan suunnitella siten että kaistat muistuttavat kriittisiä kaistoja
 - epätäydellinen rekonstruktio, mutta täydellinen aliasoinnin kumoaminen
 - *heikkouksia*: epätäydellinen rekonstruktio, pitkä algoritminen viive (jopa 250 ms, johtuu peräkkäisistä vaiheista), laskennallinen kompleksisuus
- Wavelet-pohjaiset suodinpankit (joitakin kokeiluja tehty)
- Monivaihesuodinpankit (engl. *polyphase filter banks*)
 - prototyypisuodattimen suunnittelu on joustavaa
 - laskennallisesti suhteellisen kevyitä
 - MPEG-1 audio: 511-tappinen prototyypisuodin, erittäin jyrkkä vaste (ks. kuvat seuraavalla sivulla)
 - järkevä kompromissi aikakäyttämisen ja taajuuserottelun välillä
 - *heikkous*: mahdollistaa vain tasavälisten suodinpankkien suunnittelun

Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

Audiokoodaus 26
DA / Klapuri

- Kuva: MPEG-1 Audion prototyypisuodatin [Sporer98]
 - polyphase suodinpankki
 - ikkunafunktio (vasemmalla) ja taajuusvaste (oikealla)



Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

Audiokoodaus 27
DA / Klapuri

- Fourier-muunnokseen perustuvat suodinpankit
 - diskreetti Fourier- (DFT) tai kosinimuunnos (DCT)
 - varhaiset audiokodekit käyttivät DFT:tä tai DCT:tä, lainaten ne suoraan kuvankoodauksesta
 - häiriöitä kehysten välisillä rajoilla vähennetään ikkunoinnilla ja overlap/add tekniikalla
 - nykyisin ikkunoidut muunnokset nähdään usein polyphase-suodinpankkirakenteena
 - aikaikkuna toimii prototyypisuodattimena
 - taajuusmuunnos moduloi suodatetun signaalin kantataajuudelle
 - viime aikoina kaikki korkean taajuuserottelun koodikit (muunnoskoodikit) käyttävät MDCT:tä DFT/DCT:n sijasta
 - MDCT = modified discrete cosine transform

Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

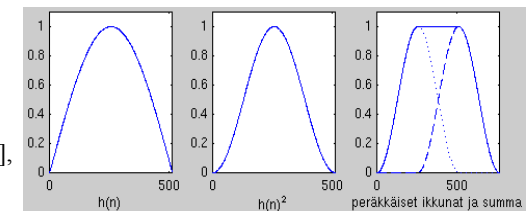
Audiokoodaus 28
DA / Klapuri

- Modified discrete cosine transform (MDCT)
 - tunnetaan myös nimellä *modulated lapped transform*
 - 1. *Aikatason laskostuminen* kumotaan suunnittelemalla ikkuna siten, että se täyttää täydellisen rekonstruktion ehdon:
$$h(i)^2 + h(i + N/2)^2 = 1, \quad i = 0, \dots, N/2 - 1,$$
missä N on ikkunan pituus
 - suomeksi sanottuna peräkkäisten ikkunoiden neliöt summautuvat ykköseen, mikäli niiden etäisyys toisistaan on \langle ikkunan pituus $\rangle / 2$
 - Miksi neliöt? Koska ikkunointi toistuu synteisipankissa → neliö.

- normaalisti käytetään sini-ikkunaa ja 50 % limittäisyyttä

$$h(i) = \sin[\pi(i+0.5)/N],$$

missä $i = 0, \dots, N-1$



Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

Audiokoodaus 29
DA / Klapuri

2. **Muunnos** tehdään DCT:llä, jota on modifioitu aikasiirtokomponentilla

$$X_t(m) = \sum_{k=0}^{N-1} h(k)x_t(k) \cos\left[\frac{\pi}{2N}(2k+1+M)(2m+1)\right]$$

missä N on kehyksen pituus, $M = N/2$ on taajuuskehyksen pituus, $h(k)$ on aikaikkuna, $x_t(k)$ ovat näytteet kehyksessä t , ja $X_t(m)$ on taajuusmuunnos

- **aikasiirtokomponentin idea:** ikkunan vasemman ja oikean puoliskon aikatason laskostumiset voidaan hoidella toisistaan riippumatta
- vrt. normaali DCT, joka vertailun vuoksi voidaan kirjoittaa

$$X_t(m) = \sum_{k=0}^{N-1} h(k)x_t(k) \cos\left[\frac{\pi}{2N} \cdot 2k \cdot 2m\right]$$

- kriittinen näytteistys: aika-taajuuskomponenttien lukumäärä on sama muunnoksen sisään- ja ulostulossa
 - yhdistää kriittisen näytteistysen hyvään taajuuserotteluun
- yleensä käytetään 128 – 2048 tasavälistä "taajuuskaistaa"

Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

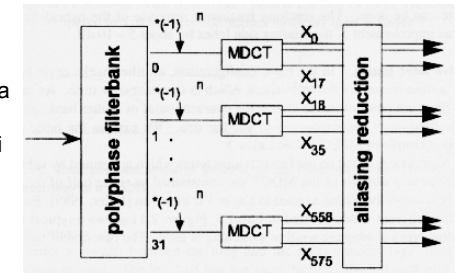
Audiokoodaus 30
DA / Klapuri

Hybridi-suodinpankit

- kostuu kaskadissa olevista erityyppisistä suodinpankeista
- suunnittelun joustavuus lisääntyy, mutta niin myös kompleksisuus
- esimerkki:
 - sisääntuleva signaali jaetaan ensin neljään kaistaan QMF-pankilla
 - jokainen kaista jaetaan edelleen 64 –128:een taajuuskanavaan
 - aikaresoluutio on 21 ms matalille ja 2.7 ms korkeimmille taajuuksille

– Kuva: MPEG-1 Layer 3 [Herre95]

- 32-kaistainen polyphase-suodinpankki (yhteensopiva Layerien 1 ja 2 kanssa)
- jaetaan edelleen 18:aan tai 6:een taajuusviivaan (ikkunan vaihtelu, engl. *window switching*, ks. seuraava sivu)



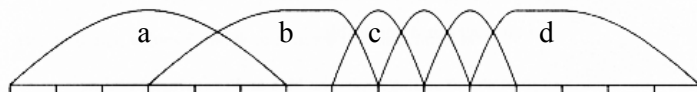
Selostus koodaustyökaluista

Käytössä olevia suodinpankeja

Audiokoodaus 31
DA / Klapuri

Adaptiiviset suodinpankit

- peruskonfiguraatiossa aika-taajuuserottelu pysyy vakiona
- adaptiivista ikkunan vaihtoa käytetään esimerkiksi MPEG-1 Layer 3:ssa
- Kuva: esimerkkisekvenssi eri ikkunatyypeillä
 - pitkä ikkuna: normaali ikkunatyyppi stationäärisille signaalikohdille
 - alkuikkuna: varmistaa aikatason aliasoinnin kumoamisen sille osalle joka menee päällekkäin lyhyen ikkunan kanssa
 - lyhyt ikkuna: saman mallinen kuin (a), mutta 1/3 pituudeltaan
→ aikaresoluutio paranee 4 ms:iin (192 vs. 576 taajuusviivaa)
 - loppuikkuna, sama tehtävä kuin alkuikkunalla



Selostus koodaustyökaluista

4.2 Kuulon mallit

Audiokoodaus 32
DA / Klapuri

- Kuulomalli muodostaa koodausjärjestelmän algoritmisen ytimen
- Useimmat koodausstandardit määrittelevät vain siirtoformaatin
 - mahdollistaa muutokset ja parannukset kuulomalliin vielä sen jälkeen kun standardi on fiksattu
 - esim. MiniDisc-formaatti esiteltiin 1991; löi läpi paljon myöhemmin
- **Kuulomallin päätehtävä** enkooderissa on tuottaa arvioita kulloinkin hyväksyttävästä kvantisointikohinan määrästä
- Lisätehtäviä ovat
 1. kontrolloida adaptiivista ikkunan vaihtoa, mikäli sellainen on käytössä
 2. kontrolloida *bittisäilö*-puskuria, mikäli sellainen on käytössä
 3. kontrolloida stereokoodauksen menetelmien käyttöä

Kuulon mallit: triviaali esimerkki

- Yksinkertaisimmillaan voidaan käyttää staattista mallia
 - tutkimalla signaali-riippuvia peittokäyriä eri taajuusalueilla saadaan selville tarvittava worst-case signaali-kohina suhde eri kaistoille
 - fiksataan lukema $SNR_{\text{worst}(i)}$ kullekin kaistalle i
 - bittiallokoinnin strategia:
 - $n_{\text{bits}}(i) = SNR_{\text{worst}(i)} / 6.02\text{dB}$
 - eli bittien lukumäärä kaistalle i saadaan worst-case signaali-kohina -suhteesta ko. kaistalle
 - kertausta kvantisoinnista: kukin lisäbitti parantaa SNR:ää 6.02 dB
- Kehittyneemmät mallit yrittävät estimoida aikariippuvaa signaali-maski suhdetta (signal-to-mask ratio SMR) kullekin kaistalle
 - tietämys peittoilmiön additiivisuudesta ja aikakäyttäytymisestä on varsin rajallinen
 - ei ole olemassa ”oikeaa” psykoakustista mallia

Kuulon mallit: tonaalisuusmitta

- Yksi tapa johtaa parempi estimaatti peittokynnykselle on tehdä ero tilanteiden välillä joissa kohina peittää siniäänestä tai päinvastoin
- Kompleksisille signaaleille ajasta t ja taajuudesta ω riippuva tonaalisuusmitta $v(t, \omega)$ johtaa parhaaseen arvioon peittokynnyksestä
- Tonaalisuutta voidaan mitata esimerkiksi yksinkertaisella polynomisella ennustajalla:
 - käytetään ennustukseen tietyn taajuuskomponentin magnitudia ja vaihetta kahdessa peräkkäisessä kehyksessä. Magnitudille esim.:
 $r(t, \omega)$:magnitudi ajanhetkellä t ja taajuudella ω
 $r_{\text{pred}}(t, \omega) = r(t-1, \omega) + [r(t-1, \omega) - r(t-2, \omega)]$:ennustettu arvo
 - etäisyys ennustetun ja todellisen arvon välillä kertoo ennustettavuudesta (”kaosmitta”)
 - kaosmitta voidaan projisoida tonaalisuusmitaksi: mitä ennustettavampi, sitä tonaalisempi

Kuulon mallit: MPEG-1 kuulomalli 2

- Taajuustason esitysmuoto
 - käytetään FFT:tä ja 1024:n näytteen mittaista Hanning-ikkunaa
 - suodinpankin ulostulosta ei saada helposti magnitudia ja vaiheita, joita tarvitaan tonaalisuuden estimointiin
- Magnitudispektristä lasketaan energia e_b kullakin 1/3 kriittisellä kaistalla. Tämän jälkeen spektriä konvoloidaan siten että piikkimäiset energiahaiput leviävät ihmiskuulon peittoilmiön leviämisen tapaan
- Tonaalisuuden t_b estimointi kullakin kaistalla b perustuu edellä mainittuun yksinkertaiseen polynomiseen ennustajaan
- Tonaalisuus vaikuttaa peittokynnykseen kaistalla b siten, että vaadittava signaali-kohina suhde on seuraava:

$$SNR_b = \max[MINVAL_b, t_b \times TMN_b + (1 - t_b) \times NMT_b]$$
 missä $MINVAL_b$ on kaistakohtainen vakio minimiarvo, ja TMN_b ja NMT_b ovat mittoja äänneksen kyvyllä peittää kohinaa ja päinvastoin
- Aikapeitto huomioidaan jos edellinen kehys oli selväsi äännekkäämpi
- Kuten arvata saattaa, esitys tässä on huomattavasti yksinkertaistettu

4.3 Kvantisointi ja koodaus

- Kvantisointi ja koodaus toteuttavat itse datan vähentämisen enkooderissa
- Suunnitteluvaihtoehtoja
 - kvantisointi: useimmat järjestelmät käyttävät tasavälistä kvantisointia. MPEG-1 ja MPEG-2 käyttävät epätasavälistä eksponentiaalista kvantisointia
 - koodaus: kvantisoidut spektrikomponentit lähetetään joko sellaisenaan, tai entropiakoodattuina sanoina
 - kvantisoinnin ja koodauksen kontrollirakenteet (tyypilliset kaksi):
 1. bittiallokointi (suora rakenne): bittiallokointialgoritmit ohjaa joko datan tilastolliset ominaisuudet tai kuulomalli. Bittiallokointi tehdään ennen kvantisointia
 2. kohinan allokointi (epäsuora rakenne): data kvantisoidaan – mahdollisesti kuulomallin mukaan. Kullekin komponentille käytetty bittimäärä voidaan laskea vasta kun kvantisointiprosessi on tehty

Kvantisointi ja koodaus: työkaluja

- Block companding (= "block floating point")
 - useita arvoja (peräkkäisiä näytteitä tai vierekkäisiä taajuusviivoja) normalisoidaan maksimiarvoon
 - maksimiarvo on blokille yhteinen skaalaustekijä ("eksponentti")
 - kaikki arvot blokin sisällä kvantisoidaan sitten siten että kvantisointiaskeleen määrää blokille allokoitujen bittien määrä

- Epätasavälinen skalaarikvantisointi

- implementoi "oletus" kohinamuodon kvantisointiaskeletta säätäen
- isommat arvot kvantisoidaan epätarkemmin kuin pienet arvot
- *esimerkiksi* MPEG-1 Layer 3:ssa ja MPEG-2 AAC:ssa näin:

$$r_{quant}(i) = \text{round} \left[\left(\frac{|r(i)|}{quant} \right)^{0.75} - 0.0946 \right]$$

missä $r(i)$ on alkuperäinen arvo, $r_{quant}(i)$ on kvantisoitu arvo, $quant$ on kvantisointiaskele, ja round pyöristää lähimpään kokonaislukuun

Kvantisointi ja koodaus: työkaluja

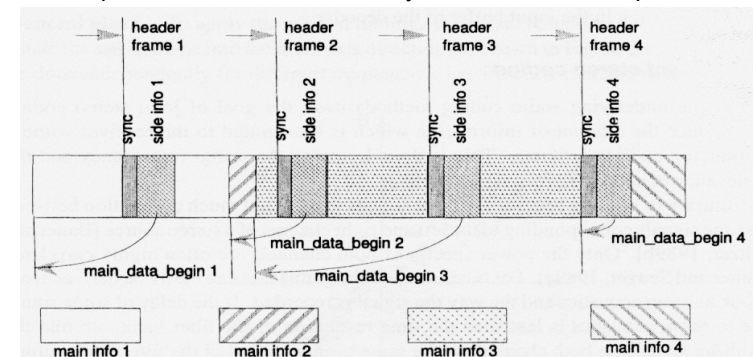
- Vektorikvantisointi
 - ei kvantisoida yksittäisiä näytteitä, vaan $n:n$ näyteen ryhmiä
 - rajataan alueita n -ulotteisesta avaruudesta, ei lukusuoralta
 - käytössä useimmissa puhe- ja videokoodausjärjestelmissä
 - esim. TWIN-VQ (oli ehdolla yhdeksi MPEG-4 audiokoodekiksi)
- Kohinan allokointi
 - ei eksplisiittistä bittiallokointia
 - taajuuskaistojen skaalaustekijöitä käytetään värittämään kvantisointikohinaa. Iteratiivinen algoritmi kohinan allokointiin:
 1. kvantisoidaan data
 2. lasketaan realisoitunut kvantisointikohina vähentämällä rekonstruoitu signaali originaalista ei-kvantisoidusta signaalista
 3. vahvistetaan signaalin tasoa niillä taajuuskaistoilla, joilla kvantisointikohina ylittää peittökynnyksen. Tämä vastaa kvantisointiaskeleen pienentämistä ko. kaistoille
 4. tarkistetaan iteraation päättymisehto (vahvistus ei ollut tarpeen, tai muu syy), muussa tapauksessa palataan kohtaan (1) käyttäen muutettuja signaaliarvoja

Kvantisointi ja koodaus: työkaluja

- Lyhytaikainen bufferointi
 - mahdollistaa paikallisesti vaihtelevan bittinopeuden
 - tavoite: tasoittaa pois paikalliset vaihtelut bittien tarpeessa
 - **bittisäilö** (engl. *bit reservoir*): bufferointitekniikka, joka täyttää tämän tarpeen
 - bittimäärä per kehyks ei ole enää vakio, vaan vaihtelee, kuitenkin siten että pidemmän aikavälin keskiarvo pysyy vakiona
 - toteutuneen bittinopeuden kumulointunut *poikkeama* tavoitteesta (keskiarvosta) rajoitetaan tiettyyn maksimiarvoon. Poikkeama on aina negatiivinen, eli todellinen bittinopeus ei saa ylittää siirtokanavan kapasiteettia
 - lisäbittejä tarvittaessa ne otetaan bittisäilöstä, ja seuraavat kehykset koodataan sitten hieman pienemmällä bittimäärällä, jotta bittisäilöön saadaan taas säästettyä bittejä
 - aiheuttaa ylimääräisen viiveen dekooderissa

Kvantisointi ja koodaus: työkaluja

- Kuva (sekavahko): esimerkki bittisäilö-tekniikasta (layer 3)
 - huomaa että lisäbittejä sijoitetaan aikaisempiin kehyksiin, joihin tilaa on saatu säästettyä, ei tuleviin kehyksiin. Näin toteutunut bittinopeus ei missään vaiheessa ylitä siirtokanavan kapasiteettia



Selostus koodaustyökaluista

4.4 Stereokoodaus

Audiokoodaus 41
DA / Klapuri

- Tavoitteena on jälleen vähentää informaatiomäärää aiheuttamatta kuultavia häviöitä
- Mahdollistuu poistamalla stereosignaalien redundanssia ja tiettyjen stereo-ominaisuuksien irrelevanttisuutta
- Redundantti informaatio
 - vastoin yleistä luuloa, vasemman ja oikean kanavan aikasignaalit eivät yleensä korreloi paljon
 - vain kanavien *tehospektrit ovat yleensä vahvasti korreloivia*
- Irrelevantti informaatio
 - ihmiskuulon kyky lokalisoida äänilähteiden avaruudellista sijaintia heikkenee korkeilla taajuuksilla
 - *korkeilla taajuuksilla avaruudellinen hahmottaminen perustuu pääasiassa intensiteettieroihin* kullakin taajuudella

Selostus koodaustyökaluista

Stereokoodauksen sudenkuoppia

Audiokoodaus 42
DA / Klapuri

- Joissakin tapauksissa stereokoodauksen vaatima bittinopeus on suurempi kuin kahden monokanavan koodauksessa
 - tietyt koodausvirheet, jotka peittyvät yksittäisellä kanavalla, tulevat kuuluviksi kun soitetaan kaksi monokanavaa
 - ns. binauraalinen (kaksikorvainen) ero peittökynnyksessä
 - havaitaan erityisesti matalilla taajuuksilla
- Presedenssiefekti
 - tietyissä tilanteissa äänilähteet lokalisoidaan pelkästään ensin saapuneen aaltorintaman tulosuunnan perusteella
 - koodaustekniikat voivat johtaa vääristyneeseen stereokuvaan

Selostus koodaustyökaluista

M/S–stereokoodaus

Audiokoodaus 43
DA / Klapuri

- Normalisoidut vasemman ja oikean kanavan summa- ja erotussignaalit siirretään itse kanavien sijasta
- Painopiste redundantin informaation poistossa
- Täydellinen rekonstruktio onnistuu dekooderissa
 - muunnos $L+R \rightarrow M+S$ ei hukkaa informaatiota
- *Saavutettu bittisäästö riippuu rankasti itse signaalista*
 - vaihtelee 50 %:sta (vasen kanava on oikean kopio) 0 %:iin

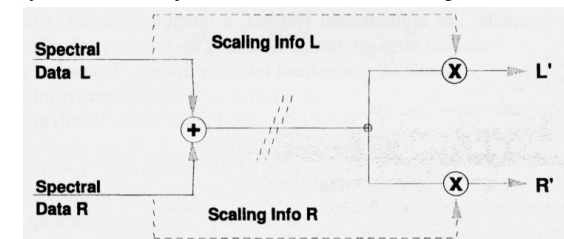
Selostus koodaustyökaluista

Intensiteetti–stereokoodaus

Audiokoodaus 44
DA / Klapuri

- Jokaisella alikaistalla säilytetään vain yksi intensiteettispektri
 - tilainformaatio siirretään koodaamalla omat skaalaustekijät vasemmalle ja oikealle kanavalle
- Toimii hyvin korkeilla taajuuksilla
 - tärkeimmät spatiaaliset johtolangat siirtyvät, yksityiskohtia saattaa puuttua
 - pidetään vähemmän häiritsevänä kuin muita koodausvirheitä
- Painopiste irrelevantin informaation poistossa
 - 50 % datan vähennys korkeilla taajuuksilla, noin 20 % koko signaalille

- Kuva: intensiteetti stereokoodauksen peruseriaate [Herre95]



Selostus koodaustyökaluista

4.5 Ennustus

Audiokoodaus 45
DA / Klapuri

- Parantaa redundanssin poistoa lähes stationäärisille signaaleille
- MPEG-2 AAC (Advanced Audio Coding)
 - kaksitappinen taaksepäin adaptiivinen ennustaja
 - suurempi määrä ennustustermejä lisäksi liikaa siirrettävää sivuinformaatiota
 - ennustusta kytketään päälle ja pois koodaushyödyn varmistamiseksi

5 Todellisia koodausjärjestelmiä

5.1 MPEG Audio

Audiokoodaus 46
DA / Klapuri

- MPEG (Moving Picture Experts Group) standardisoi videon ja audion kompressiotekniikoita
 - Kolme matalan bittinopeuden audiokoodausstandardia on valmistunut
 - MPEG-1 Audio
 - suunniteltu soveltumaan useisiin tarpeisiin: tallentamiseen, digitaalisiin radiolähetysiin, ja lähetyksiin tietoverkossa
 - järjestelmä koostuu kolmesta eri tasosta (engl. *layers* 1, 2, 3)
 - bittinopeustavoitteet: 192 kbit/s (layer 1), 128 kbit/s (layer 2), 64 kbit/s (layer 3)
 - tukee näytteistystaajuuksia 32 kHz, 44.1 kHz, ja 48 kHz
 - moodeja: mono, stereo, dual (erilliset kanavat), joint stereo (stereokoodaus)
 - MPEG-2 Backwards Compatible coding (BC)
 - MPEG-1:n kanssa taaksepäin yhteensopiva *monikanavainen* koodaus
 - koodaus myös matalammilla näytteistystaajuuksilla (16 kHz asti)
 - ei uusia koodausalgoritmeja
 - MPEG-2 Advanced Audio Coding (AAC)
 - toisen sukupolven audiokodekki geneeriseen stereo- ja monikanavasignaalien koodaukseen, mukaanlukien 5.1 ja 7.1 konfiguraatiot
 - ei taaksepäin yhtensopiva MPEG-1:n kanssa
- MPEG-2 tuo kaksi erilaista ja isoa lisäystä MPEG-1 koodekkien rinnalle

Todellisia koodausjärjestelmiä

5.2 MPEG – ensimmäinen sukupolvi

Audiokoodaus 47
DA / Klapuri

MPEG-1 tasot 1 ja 2:

- Tavallinen monivaihesuodinpankki
 - 32 alikaistaa, fiksattu aika-/taajuusresoluutio (taso 1: koko ajan 8 ms aikakehys)
 - 511 tappinen prototyypisuodin
- Psykoakustinen malli määrää bittiallokoinnin
- Kvantisointi käyttäen blokki kompondointia
 - kvantisoidaan 12:n näytteen lohkoissa, maksimiarvo skaalataan 1:een
 - skaalaustekijät kvantisoidaan eksponentiaalisesti, 2 dB välein
 - 6:lla bitillä → 64 eri arvoa → 2 dB x 64 → ≈120 dB dynaamiseksi alue
 - skaalatut arvot kvantisoidaan tasavälisesti
- Tason 2 parannuksia verrattuna tasoon 1
 - yksittäisiä pieniä lisäkeinoja bittiallokoinnin yms. pakkaamiseen
 - erilainen aikakehysjako (24 ms Layer 2:ssa, 8 ms Layer 1:ssä)
 - bittiallokointi optimoidaan iteratiiviseen tapaan siten, että signaali-maski suhde saadaan parhaaksi mahdolliseksi

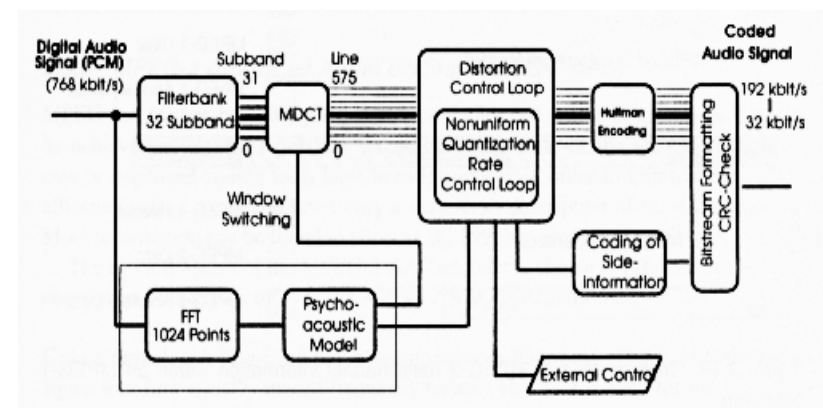
Todellisia koodausjärjestelmiä

MPEG – ensimmäinen sukupolvi

Audiokoodaus 48
DA / Klapuri

MPEG-1 taso 3 (.mp3)

- Kuva: lohkokkaavio MPEG-1 tason 3 enkooderista



Todellisia koodausjärjestelmiä

MPEG – ensimmäinen sukupolvi

MPEG-1 taso 3 (.mp3)

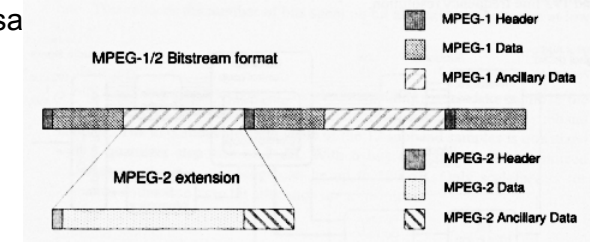
- Piirteitä tasoista 1 ja 2
- Lisää koodaustehokkuutta saavutettu
 - paremman taajuuserottelun avulla
 - koodattujen arvojen häviöttömän Huffman-koodauksen avulla
- Hybridi-suodinpankki, vaihteleva aika-/taajuusresoluutio
 - kytkeytyy kolmen vaihtoehdon välillä: 576, 216, tai 192 taajuusviivaa, selostettiin adaptiivisten suodinpankkien yhteydessä
- Muita koodaustyökaluja
 - erilainen kvantisointi, edelleen epätasavälinen kuten tasoissa 1/2
 - kvantisointikohinan kontrolli "analysis by synthesis"-periaatteella
 - stereokoodaustekniikoita
 - tukee M/S- ja intensiteetti-stereokoodauksen yhdistelmää

Todellisia koodausjärjestelmiä

MPEG – ensimmäinen sukupolvi

MPEG-2 taaksepäin yhteensopiva monikanavakoodekki

- Ominaisuuksia
 - kehysformaatti on identtinen MPEG-1 bittivirran kanssa
 - lisäkanavat siirretään MPEG-1:n "ancillary data"-kentässä
 - MPEG-1 dekooderi osaa downmiksata 5-kanavaisen informaation
 - kolme versiota, tasot 1, 2 ja 3, jotka vastaavat MPEG-1:stä
- Kuva: MPEG-2 monikanavainformaation siirto MPEG-1 bittivirrassa



Todellisia koodausjärjestelmiä

5.3 MPEG-2 AAC

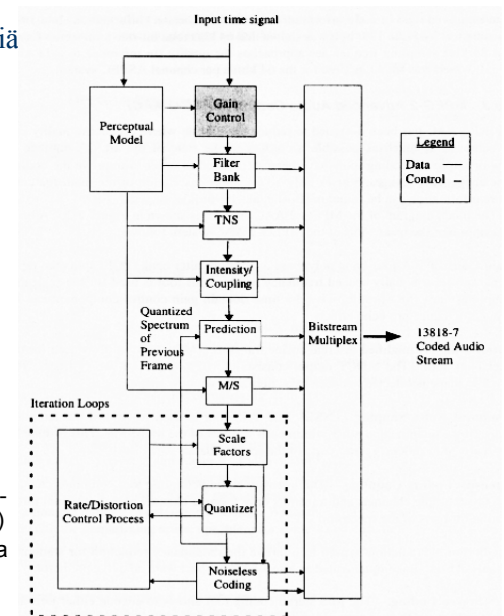
- State-of-the-art geneerinen audiokoodekki
 - Suunnittelutavoite: laskea bittinopeutta mahdollisimman paljon kuitenkin säilyttäen "broadcast" laadun
 - saavutetaan matalahkolla bittinopeudella (64 kbit/s per kanava, tai vähempi)
 - Useita uusia koodaustyökaluja tehokkuuden parantamiseksi
 - joissain sovelluksissa AAC:llä voidaan pakata stereosignaali matalammillaan 16 kbit/s bittinopeudella
 - Kompleksisuusnäkökohtia
 - eri sovelluksissa tarvitaan erilaisia koodekin kompleksisuus vs. laatu kompromisseja
 - AAC määrittelee kolme eri kompleksisuusprofiilia
 - pääprofiili
 - matalan kompleksisuuden profiili (esim. ennustus ei ole käytössä, koska ei ole käytettävissä paljon muistia)
 - skaalautuvan näytteistystaajuuden profiili
- AAC yleensäkin: hyvä koodaushyöty kohtuullisella kompleksisuudella

Todellisia koodausjärjestelmiä

MPEG-2 AAC

Lohkokaavio:

- Tason säätö
 - optionaalinen ominaisuus (käytössä skaalautuvan näytteistystyksen profiilissa)
 - neljään tasaväliseen kaistaan jakava monivaihesuodinpankki
 - aikataason signaalin tasoa voidaan säätää esikaiku-efektien vähentämiseksi
- Suodinpankki
 - MDCT: 1024 tai 128 taajuusviivaa (aikaresoluutio parhaimmillaan 2.7 ms)
 - adaptiivinen: kaksi erilaista ikkunamuotoa



Todellisia koodausjärjestelmiä

MPEG-2 AAC

Audiokoodaus 53

DA / Klapuri

... Lohkokaavio

- Temporal noise shaping (ks. kohta 3.2)
 - tehdään säätämällä vahvistuksia eri taajuuskaistoille
- Intensiteettikoodaus useille kanaville (ks. kohta 4.4)
- M/S-stereokoodaus (ks. 4.4)
- Ennustus
 - toisen asteen taaksepäin adaptiivinen ennustaja (ks. 4.5)
 - kullekin spektrikomponentille 16 kHz taajuuksiin asti
- Skaalaustekijät
 - spektri jaetaan useisiin spektrikomponenttien ryhmiin, jotka jakavat yhden skaalaustekijän
- Kvantisointi
 - epätasavälinen kvantisointi, askeleen koko 1.5 dB
- Huffman-koodausta käytetään koodaamaan kvantisoitu spektridata
 - häviötön koodaus

Todellisia koodausj.

5.4 Vertailua

- Lähde: [MPEG-4 dokumentti]
- Tulokset ovat keskenään vertailukelpoisia vain kunkin paksuilla vaakaviivoilla erotetun ryhmän sisällä (kanavamäärä vaihtuu)
- AAC tuottaa broadcast-laatua viisikanavaisesti 320 kbit/s

Coding tool	Number of channels	Total bit rate	Typical subjective quality
AAC	5	320 kb/s	4.6
1995 Backward Compatible MPEG-2 Layer II	5	640 kb/s	4.6
AAC	2	128 kb/s	4.8
AAC	2	96 kb/s	4.4
MPEG-2 Layer II	2	192 kb/s	4.3
MPEG-2 Layer III	2	128 kb/s	4.1
AAC	1	24 kb/s	4.2
Scalable: CELP base and AAC enhancement	1	6 kb/s base, 18 kb/s enh.	3.7
Scalable: Twin VQ base and AAC enhancement	1	6 kb/s base, 18 kb/s enh.	3.6
AAC	1	18 kb/s	3.2
G.723	1	6.3 kb/s	2.8
Wideband CELP	1	18.2 kb/s	2.3

6 MPEG-4 Audio

Audiokoodaus 55

DA / Klapuri

» Luettavuuden helpottamiseksi MPEG-4 käsitellään omassa kappaleessaan, vaikka se loogisesti kuuluisikin edellisen alle

- Standardi saatiin valmiiksi 2000
- **Koostuu perheestä koodausalgoritmeja**, jotka tähtäävät eri bittinopeuksiin ja eri sovelluksiin
- Poistaa kuilun puheenkoodauksen, kuulon malleihin perustuva koodauksen, ja signaalien synteessin väliltä
 - bittinopeudet vaihtelevat välillä 2 kbit/s – yli 128 kbit/s per kanava
- Ominaisuuksia
 - skaalautuvuus erilaisille päätelaitteille
 - objektipohjaisuus: päätelaite voi valita äänivirrasta vain osia dekodattavaksi, ja voi uudelleen miksata tai manipuloida synkronisoituja ääniobjekteja
 - hyvä koodaustehokkuus (bittinopeus vs. laatu) saavutetaan valitsemalla kullekin signaalille sopiva koodausalgoritmi

MPEG-4 Audio

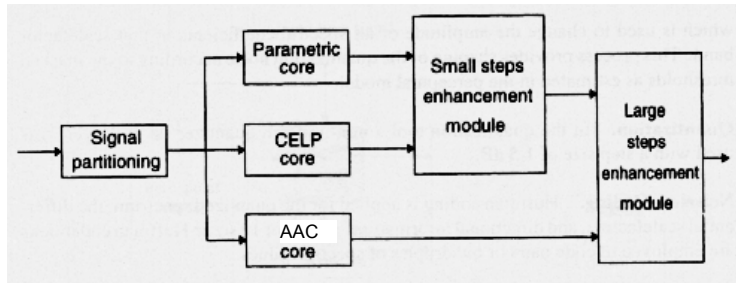
Audiokoodaus 56

DA / Klapuri

- Taustamotivointia – koodauksen dilemma:
 - Yleiskäyttöiset **kuulon malleihin** perustuva audiokoodekit
 - saavutetaan hyvä laatu kaikentyyppisille audiosignaaleille bittinopeudella 64 kbit/s (MPEG-2 AAC)
 - perustuu kuulon kannalta irrelevantin informaation poistamiseen
 - puhetta tai yksittäisen soittimen ääntä *ei* voida koodata muita signaaleja ratkaisevasti pienemmällä bittinopeudella (AAC: puhekin vaatii ≥ 24 kbit/s)
 - Puhekoodekit, esim. ITU-T
 - saavutetaan hyvä laatu puhesignaaleille yli 4 kbit/s nopeuksilla
 - perustuu pääasiassa signaalin **lähteen**, puheentuottoelimestön, **mallintamiseen**
 - ei oikein sovellu musiikkisignaalien koodaamiseen
 - Mahdollisia ratkaisuja
 1. rajoitetaan tukemaan vain tiettytyypistä audiomateriaalia
 2. määritellään mielekäs yhdistelmä erilaisia koodekkityyppejä
- joku valinta täytyy tehdä

MPEG-4 Audio

- Kuva: MPEG-4:n skaalautuva konfiguraatio
 - *parametrinen ydin*: hyvin matalan bittinopeuden koodekki, joka perustuu äänen parametriseseen esitystapaan
 - *CELP-ydin*: puhekoodekki (Code Excited Linear Predictive coding)
 - *AAC-ydin*: aika/taajuus –esitysmuotoon perustuva kuulomalliin perustuva koodekki, *MPEG-2 AAC*
 - *enhancement*: valinnaista lisäinformaatiota, joka lisätään bittivirtaan koodatun minimi-informaation lisäksi (skaalautuva laatu)



MPEG-4 Audio

- MPEG-4 tukee edellä mainittujen koodekkien ja niiden yhdistelmien käyttämistä
- Saavutetut bittinopeudet
 - puhesignaalit: 2 – 24 kbit/s
 - dekooderissa syntetisoitu puhe (siirretään tekstimuodossa mahdollisten kontrolliparametrien kera): 200 – 1200 bit/s
 - kompleksiset musiikkisignaalit: 6 – yli 128 kbit/s (monikanavatuki)
 - ei-kompleksiset musiikkisignaalit (synteettinen audio): ≥ 4 kbit/s
 - siirretään nuotti- ja kontrolli-informaatio, sekä tietyllä kielellä määritellyt instrumenttiäännet
 - kokonaiset äänimaisemat voidaan koota eri algoritmeilla koodatuista osista

MPEG-4 Audio

- Skaalautuva audiokoodaus
 - pelkästään osa bittivirrasta riittää, jotta voidaan dekodata ja generoida merkityksellinen audiosignaali
 - bittivirrassa on erikseen koodattuna minimi-informaatio, ja sen lisäksi "enhancement" osia, jotka yhdistämällä laatua voidaan parantaa
 - tavoite: multicast on tehokkaampaa kuin simulcast — on tehokkaampaa siirtää samaa bittivirtaa kaikille vastaanottajille, joista jokainen valitsee sen osan kuin kykenee dekoodaamaan/esittämään vastaanottimessaan
- Objektipohjaisuus
 - ääniä voidaan mikrata ja tuottaa päätelaitteessa, audio-objekti voidaan sijoittaa avaruudellisesti haluttuun paikkaan
 - binääriformaatti äänimaisemien määrittelyyn (engl. *binary format for scene description*): käytetään selostamaan miten ääniobjektit miksataan, kunkin efektit, yms. post-produktio
 - sisältöön perustuva interaktiivisuus (MPEG-4:n perusfilosofioita)
 - esim. virtuaaliodellisuudessa äänilähteestä voi kävellä kauemmaksi
 - valikoiva dekooodaus: äänisignaalin eri osat voidaan koodata erikseen, ja silti synkronisoida yhteen. Tämä mahdollistaa signaalin eri osien koodaamisen eri algoritmeilla, ja valikoivan dekoodaamisen

MPEG-4 Audio

6.1 CELP–puhekoodekki

- Puheinformaatio on moduloituneena spektrin karkeaan muotoon
 - foneemit erottovat toisistaan karkean spektrimuotonsa perusteella
 - korostuneita taajuusalueita nimitetään *formanteiksi*
 - spektrin hienorakenne ei muuta puheen foneemisäilytystä
 - tieto äänteen soinnillisuudesta ja äänenkorkeus välitetään erikseen
- Lähdemalli
 - perustuu malliin ihmisen puheentuottoelimistöstä. Tietyn tyyppiset signaalit ovat "mahdollisia", ei esim. äärettömän nopeat laajat muutokset
 - soinnillinen / soinniton heräte + ääniväylä, jota mallinnetaan suodattimella
 - muuttaen ääniväylän muotoa (esim. kielen ja leuan asentoa) foneemi vaihtuu
 - ks. seuraava sivu
- Kuulomalli
 - kvantisointikohinan spektrin muodon pitäisi olla samanlainen sisääntulevan signaalin kanssa
 - implementoitu spektriä painottavalla suodattimella
- Bittinopeudet
 - CELP: puhe 4 kbit/s (kelvollinen laatu) – 24 kbit/s (erittäin hyvä laatu)

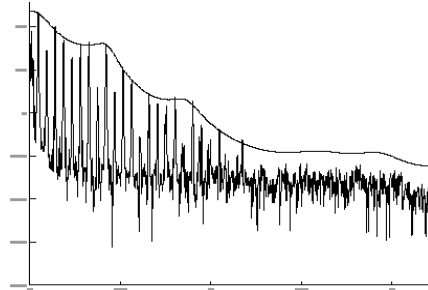
MPEG-4 Audio CELP-puhekoodekki

Audiokoodaus 61
DA / Klapuri

- Matala-asteinen suodatin sopii hyvin spektrin karkean muodon mallintamiseen, siis koodaamaan foneemitiedon
 - käytetään ns. ennustavia suotimia (LPC: linear predictive coding)
 - käytännössä käytetään lähes aina IIR-suodattimia, joissa on vain napoja (ei lainkaan nollia, osoittaja = 1)
 - suodatin on muotoa $H(z) = 1 / A(z)$, missä $A(z)$ on z :n polynomi
 - tämä johtuu siitä, että korostuneiden taajuusalueiden, formattien, ajatellaan välittävän olennaisen foneemi-informaation

- estimoitavan suodattimen parametrimäärää saadaan pienemmäksi

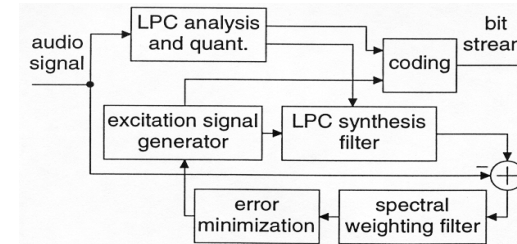
- Kuva: signaalinpätkän magnitudispektri, sekä siitä LPC-analyysillä estimoitu spektrin karkea muoto



MPEG-4 Audio CELP-puhekoodekki

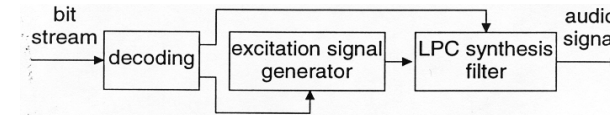
Audiokoodaus 62
DA / Klapuri

- Enkooderin lohkokaavio [Edler97]



- Dekooderi

- herätesignaalin generointi perustuu koodikirjaan

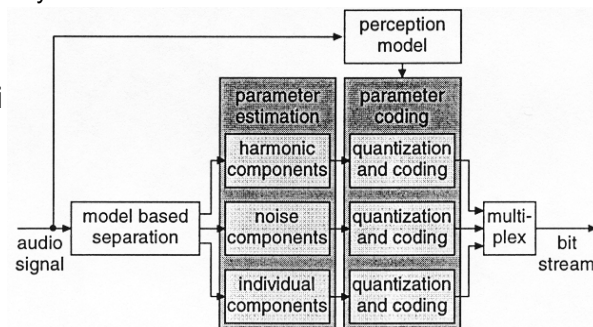


MPEG-4 Audio 6.2 Parametrinen koodekki

Audiokoodaus 63
DA / Klapuri

- Lähdemallit
 - harmonisia ääniä
 - kohinaa
 - yksittäisiä sinikomponentteja
 - yhdistelmä yllä olevista

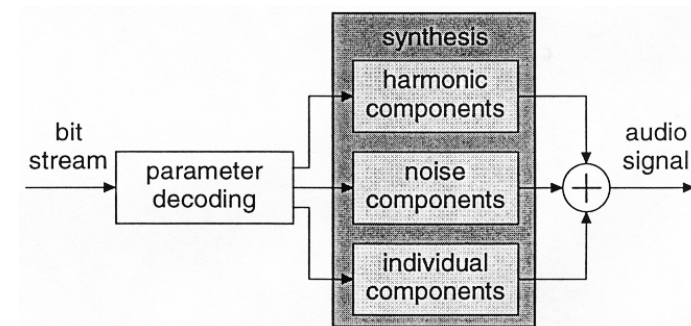
- Kuva: enkooderi [Edler97]



MPEG-4 Audio Parametrinen koodekki

Audiokoodaus 64
DA / Klapuri

- Kuva: dekooderi [Edler97]



MPEG-4 Audio

Parametrinen koodekki

Audiokoodaus 65
DA / Klapuri

- Kaksi toimintamoodia
 - harmonisia ja kohinakomponentteja
 - soveltuu puheenkoodaukseen 2–4 kbit/s bittinopeuksilla
 - yksittäisiä sinikomponentteja
 - vähemmän kompleksisten musiikkisignaalien koodaukseen, (esim. yksittäinen instrumentti) nopeuksilla 4–16 kbit/s
 - yhdistelmä molemmista moodeista
 - syntaksi tukee, määritellään transiitikohta
 - automaattinen moodin valitsin
- Parametrisen koodekin mahdollistamia lisätoiminnallisuuksia
 - toistonopeuden muuntelu vaihtamalla synteessin ajastusta
 - äänenkorkeuden muuntelu
 - skaalaamalla taajuusparametreja
 - puheelle spektrin karkea muoto pidetään paikallaan, jolloin pään (ääniväylän) dimensioiden aistimuksellinen kutistuminen voidaan välttää
- Bittinopeudet: puheelle 2–4 kbit/s, ei-kompleksinen musiikki 4–16 kb/s

MPEG-4 Audio

6.3 Koodaustekniikoiden yhdistely

Audiokoodaus 66
DA / Klapuri

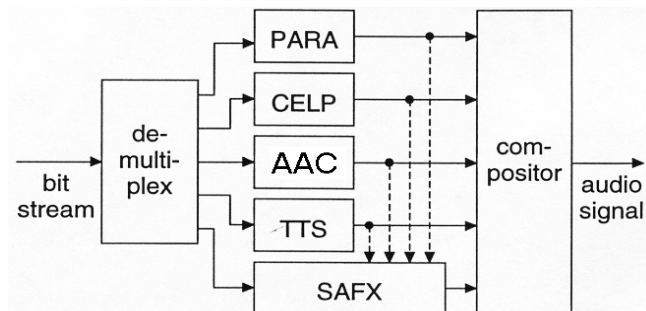
- Kohdesignaalina esim. puhe + taustamusiikki
- Signaalin osien erillinen koodaus voi parantaa suorituskykyä
- Eri osat oltava erillisinä signaaleina
 - äänilähteiden separointi: sisäänmenevästä polyfonisesta signaalista erotellaan kukin äänilähde omaksi signaalikseen
 - automaattinen analyysi ja separointi onnistuu nykytekniikalla ainoastaan hyvin yksinkertaisissa tapauksissa
 - signaalin eri osat on pidettävä erillään tuotantovaiheessa
 - koodekin valinta kullekin signaalille voi tapahtua käsin tai automaattisesti

MPEG-4 Audio

6.4 MPEG-4 dekooderi

Audiokoodaus 67
DA / Klapuri

- Aiempien MPEG-standardien tapaan vain bittivirta ja dekooderi on määriteltä
 - enkooderien optimointi tulevaisuudessa käy mahdolliseksi
- Kuva: MPEG-4 dekooderi [Edler97]



MPEG-4 Audio

Dekooderi

Audiokoodaus 68
DA / Klapuri

- Dekooderissa näkyy kaksi uutta lohkoa
 - TTS (text-to-speech synthesis): tekstistä puheeksi syntetisointi
 - SAFX: structured audio and effects
- Näiden abstraktiotaso on niin korkea, ettei tarvitse valita/määritellä mitään tiettyä koodekkityyppiä
 - tekstin syntetisointiin tarvittavat parametrit ovat yleisesti tiedossa
 - musiikkiparametrit tiedossa (MIDI)
- Huomaa: vaikka MPEG määrittelee vain dekooderin, määrää esim. puhekoodekin bittivirran määrittely kuitenkin *koodekkityypin*, samoin geneeriselle koodekille
 - näin ei ole kahdessa yllä mainitussa tapauksessa, kyse on korkeammasta, semanttisen tason esitysmuodosta

MPEG-4 Audio Dekooderi

Audiokoodaus 69
DA / Klapuri

- MPEG-4 tukee ns. synteettisen ja luonnollisen koodauksen hybridiä
 - dekooderissa voidaan yhdistää korkean tason semanttisesta tiedosta syntetisoitua ja koodattua ääntä
- Ääniä voidaan miksata, tuottaa ja avaruudellisesti lokalisoida terminaalissa
 - binääriformaatti äänimaisemien määrittelyyn mahdollistaa selostuksen miksausesta, efekteistä, yms. post-produktiosta
 - omia efektejä voi määritellä SAFX:n orkestrointikielellä

MPEG-4 Audio

Audiokoodaus 70
DA / Klapuri

6.5 Tekstistä puheeksi synteesi

- Standardi sisältää määrittelyn prosodisista parametreista, joilla puheeseen saadaan ääntämyksellistä elävyyttä yms.
 - puheen nopeus ja äänenkorkeus
 - puhujan ikä ja sukupuoli voidaan määrittellä
 - eri kielille ja aksenteille on omat koodinsa
 - kansainvälisesti kattava joukko foneemeja, joille on kullekin omat koodinsa
 - huulisynkronisointi kuvan kanssa on mahdollista
 - kasvojen animaatioparametrit voidaan välittää

MPEG-4 Audio

Audiokoodaus 71
DA / Klapuri

6.6 Strukturoitu audio (SA)

- Audion koodaus korkealla semanttisella tasolla
- SA-dekooderi: muusiikin ja ääniefektien synteesi
- Äänidata
 - *structured audio orchestra language*
 - **instrumenttiäänten** synteesimenetelmien kuvauskieli (vrt. Csound, NetSound)
 - mitään yhtä synteesimenetelmää ei ole valittu, vain niiden kuvauskieli
 - käytettävissä peruselementtejä, kuten oskillaattoreita ja suotimia
 - *structured audio sample bank format*
 - ääniä voidaan esittää myös aaltomuotodatan avulla
 - perustana ns. wavetable sampling synteesi (syntetisaattoreissa käytetty)
 - suunnattu halvoille ja yksinkertaisille päätelaitteille
- Nuotti- ja kontrollidata: *structured audio score language*
 - formaatti, jolla **nuottidata ja äänten kontrolliparametrit** esitetään
 - käytetään ohjaamaan ääniä, jotka on määritetty yllä mainituilla tavoilla
 - vrt. MIDI, joka on myös otettu mukaan vaihtoehtoiseksi ja yksinkertaisemmaksi kuvausformaatiksi

MPEG-4 Audio

Audiokoodaus 72
DA / Klapuri

6.7 Avoimia kysymyksiä

- Enkooderien optimointi jatkuu
 - valittujen koodekkityyppien puitteissa edelleen mahdollista, koska vain dekooderi on speksattu
- Optimaalisen koodaustekniikan valinta automaattisesti kulloisellekin signaalille
- Automaattinen signaalien separointi, jotta voidaan käyttää sopivaa koodekkia kullekin ääniobjektille

Kokonaiskuva MPEG:stä

- MPEG-1
 - VHS-laatu
 - video-CD
 - mp3 audio
- MPEG-2
 - DVD-laatu
 - DVD, digi-TV boxit
 - AAC audio
- MPEG-4
 - skaalautuva laatu
 - skaalautuva jakelu: matkapuhelin vs. satelliitti-TV
- (MPEG-7: sisällön kuvaus, MPEG-21 jakelu: oikeuksien hallinta)

7 Onko MPEG:n ulkopuolella elämää?

- Kehittyneimmät kuulomalleihin perustuvat audiokoodekit on rakennettu MPEG Audion viitekehysessä
- Eri yhtiöiden tuotteissa on omia koodekkivariaatioitaan
 - näistä kerrotaan myös ”kuluttajan audiotekniikkaa”-luennon yhteydessä
 - MiniDisc: adaptive transform acoustic coding (ATRAC)
 - Digital Compact Cassette: precision adaptive subband coding (PASC)
 - Dolby: AC-3 -koodaus
 - AT&T: perceptual audio coder (PAC)

Onko MPEG:n ulkopuolella elämää?

- Uudempia koodekkeja:
- MP3Pro: perustuu MPEG 1 layer 3:een
 - lisäksi korkeat taajuudet koodataan karkeasti ja rekonstruoidaan käyttäen myös matalampia taajuuksia
- Ogg Vorbis: kehitetty eteenpäin MPEG 1 layer 2:sta
 - avoimen lähdekoodin koodekki
 - aika/taajuus-esitys: MDCT
 - vektorikvantisointi
 - taustakohinamalli välttää useille koodekeille tyypilliset ”liverrys”-artefaktat matalilla bittinopeuksilla
- Windows Media Audio
 - aika/taajuus-esitys: MDCT, 5 eri ikkunan pituutta
- Noudattelevat suurinpiirtein samoja peruslinjoja kuin MPEG-koodekit