HOOFDSTUK 5: Digitale audio

§1. Geluid

geluidsgolf

- wat
  - drukwisselingen in bepaald medium (geen luchtledige)
  - longitudinaal (in richting van energie)

- eigenschappen
  - golflengte (en dus frequentie)
    - bepaald toonhoogte
    - toonhoogte vastgelegd (vb. la = x Hz)
    - octaven: steeds frequentie x2 (vb. la' = 2x Hz)
  - amplitude
    - bepaald sterkte van geluidsgolf
      - maximale druk in een punt
    - relatief t.o.v. referentiedruk (0 dB zie verder)
  - hebben typische golfeigenschappen
    - reflectie (= weerkaatsten op oppervlakten)
    - diffractie (= buigen rond object)
    - van richting veranderen bij overgang tussen mediums

- analoge geluidsgolf
  - weergave als goniometrische functie <-> eigenlijk longitudinale golf
  - druk in een punt meten in functie van de tijd
    - $f(t)$
  - gevolg
    - geluid gedefinieerd adhv 1 variabele (A), afhankelijke van 1 variabele (t)
    - <-> beelden: afhankelijk van 2 variabelen (x,y)
    - <-> video: afhankelijk van 3 variabelen (x,y,t)

- geluid vs. spanning, vermogen, …
  - geluidsignaal als spanning V
    - grafiek van geluidsgolf gewoon interpreteren als grafiek van spanning
    - dit is wat gebeurt in een microfoon
      - daar omzetting van druksignaal naar elektrisch signaal

  - vermogen V^2 van een geluidssignaal
    - reden: spanning fluctueert rond 0
      - => op lange termijn spanning altijd 0
    - daarom kwadrateren
      - => dan maat voor intensiteit

speciale geluidssignalen

goed geluidssignaal

geluidsgolf als som van sinussen (en cosinussen) met termen:

gehele veelvouden van de grondfrequentie als frequentie = ‘harmonieken’

elk met hun frequentiecomponenten (coëfficiënten bij die term)

complexe geluidsgolf als som van de harmonieken

bandgelimiteerd (‘braaf’) geluidssignaal

fouriergetransformeerde heeft beperkte bandbreedte

dus frequenties van signaal tussen f_1 en f_2

digitale audio vs. digitale video

digitale video

zowel opnemen als weergeven kan digitaal én analoog

vb. kijken naar discrete pixels

volledig digitaal systeem mogelijk

digitale audio

zowel opnemen als weergeven steeds analoog

vb. uit digitale of analoge waarden een analoge geluidsgolf maken

enkel intern digitaal opslaan/bewerken/verwerken

<-> I/O altijd analoog

decibel (dB)

wat?

geen eenheid!

logaritmische schaal om verhoudingen aan te duiden

vooral van vermogens

meestal één v/d twee als constante referentiewaarde

eigenschappen

0 dB is verhouding 1 (dus gelijkheid, dus de referentiewaarde zelf)

+10 dB -> verhouding 10 (dus 10x groter dan referentiewaarde)

+20 dB -> verhouding 100 (dus 100x groter dan de referentiewaarde)

formules

$L = 10 \cdot \log_{10} \frac{I_1}{I_0}$ dB of m.a.w. $\frac{I_1}{I_0} = 10^{L/10}$

met I_1 een (vermogens)grootheid en met I_0 de referentiewaarde

veelgebruikte toepassing: geluidsniveau

verhouding van de geluidsintensiteit t.o.v. geluidsdrempel (net hoorbaar geluid)

$L = 10 \cdot \log_{10} \frac{J}{J_0}$ dB of $L = 20 \cdot \log_{10} \frac{p}{p_0}$

met J de geluidsintensiteit en p de geluidsdruk ($J = p^2$)

§2. Digitaliseren van geluid

digitaliseren van de geluidsgolf

principe

beide assen digitaliseren of discrediteren door ze te samplen

tijd en amplitude

'samplen' of bemonsteren

hoeveelheid meten in bepaalde intervallen

meestal van gelijke grootte

meestal gebruikt voor tijd

<-> kan voor alle grootheden

ook gewoon 'samplen' genoemd

sampling frequentie (Hz) geeft grootte intervallen aan

tijd-as

'samplen'

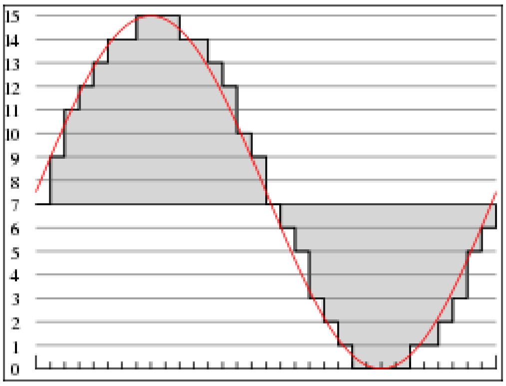
doel: maar op bepaalde tijdstippen waarde meten <-> alle tijdstippen

amplitude-as

'kwantiseren'

doel: maar bepaalde kleine 'categorieën' amplitudes overhouden <-> alle waarden

gevolg



parameters

welke sampling rate (tijdas) [1]

welke nauwkeurigheid voor kwantisatie en uniform verdeeld? [2]

bepalen sampling rate [1]

Nyquist stelling

gegeven een goed en bandgelimiteerd audiosignaal

volledig analoog signaal te reconstrueren adhv discreet aantal tijdstippen

ALS sampling rate >= 2 x intervalbreedte van frequenties

of sampling rate >= 2 x ($f_2 – f_1$)

praktisch: oogpunt signaal (of sampler maker)

gegeven een bepaald bandgelimiteerd signaal

heeft bepaalde bandbreedte

welke sampler mijn signaal volledig reconstrueren

=> al diegene met als sampling rate minimaal 2x mijn bandbreedte

die grens = Niquist rate

vb. bandbreedte signaal 50Hz => maak sampler van (minimum) 100Hz

praktisch: oogpunt sampler (of sampler gebruiker)

gegeven een sampler

heeft bepaalde sampling rate

welke signalen kan mijn sampler volledig reconstrueren

=> al diegene met als bandbreedte maximaal ½ x mijn sampling rate

die grens = Niquist frequency

vb. sampling rate 10kHz => alle signalen met bandbreedte onder 5kHz OK

SNR: Signal to Noise Ratio (Signaal-ruisverhouding)

signaal vs. ruis

signaal = alles wat we willen horen

ruis = al de rest

doel: signaal >>> ruis

Signal to Noise Ratio

verhouding ertussen (in dB)

maat voor kwaliteit van signaal (hoe hoger, hoe beter)

berekening

$$SNR = 10 \cdot \log_{10} \frac{V_{signal}^2}{V_{noise}^2} = 20 \cdot \log_{10} \frac{V_{signal}}{V_{noise}}$$

met V de spanning (of druk)

met V^2 het vermogen of de intensiteit van het signaal

SQNR: Signal to Quantization Noise Ratio

kwantisatie geeft afrondingsfout

dus ook ruis (want iets dat we niet willen horen)

afhankelijk van aantal bits N om waarden op te slaan

want maximaal 2^N verschillende waarden

quantisation noise (kwantisatieruis)

= verschil tussen echte (analoge) waarde op een (gesampled) tijdstip

en waarde van dichtste intervalwaarde

maximaal half interval

Signal to Quantization Noise Ratio

verhouding tussen signaal en kwantisatieruis (in dB)

maat voor kwaliteit van kwantisatie (hoe hoger, hoe beter)

berekening

$$SQNR = 10 \cdot \log_{10} \frac{V_{signal}^2}{V_{quan\_noise}^2} = 20 \cdot \log_{10} \frac{V_{signal}}{V_{quan\_noise}}$$

met V de spanning (of druk)

met V^2 het vermogen of de intensiteit van het signaal

lineaire en niet-lineaire kwantisatie [2]

situering

lineaire kwantisatie is zeer dom!

want bij stille volumes quantization noise veel groter waargenomen

zie SQNR met constante $V_{quan\_noise}^2$

dan kleinere verhouding bij kleine signaalintensiteit

=> daar fijner kwantiseren

principe: gelijke fouten moeten overeenkomen met gelijke perceptie (respons)

signaalruimte en responsruimte

verband tussen signaalruimte en responsruimte

$$\mathrm{d}r = k \cdot \frac{1}{s}\, \mathrm{d}s$$

met r respons en s absolute prikkel (constante k)

en geïntegreerd

$$r = k \cdot \ln s + C$$

of anders gesteld

$$r = k \cdot \ln {}^{s}/_{s_0}$$

met s_0 laagste prikkel die respons veroorzaakt ($s = s_0 \rightarrow r = 0$)

systeem

eerst signaal omzetten van signaalruimte naar responsruimte

via A-law of µ-law

daar uniform kwantiseren!

audio filtering

inputkant

doel: zorgen dat elk signaal zeker kan gesampled worden

band-pass filter toepassen om bandbreedte te beperken tot correct interval

(zie Nyquist frequentie)

outputkant

probleem: discontinue functie bij Fouriertranformatie oneindig grote frequenties

wiskundig altijd zo!

< digitale opslag

low-pass filter toepassen om alleen lagere frequenties door te laten

specificaties van verschillende audio applicaties

specificaties

	Telefoon	**FM Radio**	**CD**	**DVD**
sample rate (kHz)	8	22	44	192 (max)
bits per sample	8	16	16	24 (max)
# channels	1	2	2	6 (max)
data rate (kB/sec)	8	88	176	1200 (max)
frequentieband (kHz)	0.2-3.4	0.02-11	0.005-20	0-96 (max)

opmerkingen

data rate = samples per sec x bits per sample x #channels

vb. FM Radio: 22000 1/s x 16 bits x 2 = 704000 bps = 88000 B/s = 88 kB/s

§3. Coderen van geluid

algemeen

opnemen van geluid

1. samplen + niet-uniforme kwantisatie
2. transformatie
   - input data omzetten naar andere representatie
   - doel: makkelijker en betere compressie
2. coderen
   - codewoord aan elk outputniveau toekennen
   - 2 soorten: vaste lengte of variabele lengte (vb. Huffman codering)

lossy- en lossless audio coding

enkel over extra (compressie)verlies bij coderen <-> steeds verlies bij digitaliseren

PCM: Pulse Code Modulation

gewoon waarden opslaan (geen transformatie)
wat we in deel 2 gebruikt hebben

differentiële codering

uitgangspunt: verschil tussen waarde en vorige meestal klein
indien signaal beetje consistent in de tijd
= temporele redundantie
principe: sla verschil op met vorige ipv steeds absolute waarden
gevolg 1: verschilwaarden kleiner dan absolute waarden
=> kleinere representatie
gevolg 2: meer gepiekt histogram met maximum rond 0
=> meer voordeel bij variabele lengte codering

Lossless Predictive Coding

principe

voorspelt waarden op een vaste wijze (adhv vorige waarden)
slaat fouten tegenover voorspelling op

systeem

inputwaarden f_n komen binnen
voorspelde waarde $\widehat{f_n} = F(f_{n-1}, f_{n-2}, f_{n-3}, \ldots)$ bepalen

$\widehat{f_n} = f_{n-1}$ (vb. gelijk aan vorige, eenvoudigst)
$\widehat{f_n} = \sum_{k=1}^{2\text{tot}4} a_{n-k} f_{n-k}$ (vb. lineaire voorspeller in functie van enkele vorige)

fout e_n tussen echte en voorspelde waarde berekenen en encoderen

$e_n = f_n - \widehat{f_n}$

latere verliesloze reconstructie mogelijk

$f_n = \widehat{f_n} + e_n$
ook als geen temporele redundantie of slechte schatter (enkel minder compressie)

DPCM: Differential PCM

principe

we passen lossless predictive coding toe

=> we krijgen reeks verschillen

deze waarden gaan we ook nog eens kwantiseren!

DUS nog eens extra verlies invoeren!

systeem

inputwaarden f_n komen binnen

voorspelde waarde $\widehat{f_n} = F(\tilde{f}_{n-1}, \tilde{f}_{n-2}, \tilde{f}_{n-3}, \dots)$ bepalen

fouten e_n berekenen

$e_n = f_n - \widehat{f_n}$

kwantiseren naar foutwaarden $\tilde{e}_n$ en coderen

$\tilde{e}_n = Q[e_n]$

latere verliesloze reconstructie NIET mogelijk

$\tilde{f}_n = \widehat{f_n} + \tilde{e}_n$

dus verlieshebbende codering

<-> nog andere compressiemethoden mogelijk

DM: Delta Modulation

principe

vereenvoudigde versie van DPCM (zeer snel)

kwantiseren naar twee waarden $\widetilde{e_n}$

voor stijgend en dalend

1 bit voor foutwaarde

systeem

gelijkheid voorspellen (met aangepaste lossy versie!!)

$\widehat{f_n} = \tilde{f}_{n-1}$

fouten e_n berekenen

$e_n = f_n - \widehat{f_n} = f_n - \tilde{f}_{n-1}$

kwantiseren naar foutwaarden $\tilde{e}_n$ en coderen

$\tilde{e}_n = +k \text{ als } e_n > 0 \quad \text{en} \quad \tilde{e}_n = -k \text{ anders}$

met k een constante

latere verliesloze reconstructie NIET mogelijk

$\tilde{f}_n = \widehat{f_n} + \tilde{e}_n$

voorbeeld

$f_i = \{10,11,13,15\} \quad \text{en} \quad k = 4$

$\tilde{f}_i = \{10,14,10,14\}$ dus nog relatief dicht bij echte waarden slechte situaties

- bijna constant signaal bij relatief grote k -> geblokt
- snel stijgende signalen (veel sneller dan k) -> kan niet volgen
  - oplossing: zeer veel samplen (vele keren Nyquist rate)

adaptive DM

- uitganspunt:
  - k laten aanpassen aan stijlheid van de curve
  - zo slechte gevallen proberen wegwerken
- gevolg:
  - kleine fluctuaties (±ct) -> kleine k -> geen grote blokken
  - grote fluctuaties (stijl) -> grote k -> sneller meekunnen