Kvantizacija

Linearni kvantizator – L-PCM (pulse-code modulation) – A/D konverter:

Ulazni signal:	x
Kvantizirani signal:	x_q
Broj kvantizacijskih nivoa:	N
Razmak kvantizacijskih nivoa:	Δ
Rezolucija kvantizatora [bit]:	b

Kvantizirani signal:

$$x_q = \Delta \cdot round\left(\frac{x}{\Delta}\right)$$

Ulazna dinamika ADCa [V]:

$$A = N \cdot \Delta$$

Granulacijska pogreška kvantizatora za danu ulaznu dinamiku:

$$\Delta = A \cdot 2^{-b}$$

Ulazna dinamika preko bitova je

$$A = 2^b \cdot \Delta$$

Signal kao slučajni proces:

Realizacija slučajnog procesa X: $x[n]$

Realizacija procesa pogreške: $e[n] = x[n] - x_q[n]$

$$|e| \leq \Delta / 2$$

Gustoća razdiobe procesa X: $f_x(X)$ *pdf*

Gustoća razdiobe greške e: $f_e(e) = \dfrac{1}{\Delta}$

Varijanca signala: $\sigma_x^2 = E(x - E(x))^2$

Za uniformni signal: $\sigma_x^2 = E(x^2) = \int_{\Phi} x(\Phi)^2 f_{\Phi}(\Phi)\, d(\Phi)$

$$f_{\Phi}(\Phi) = pdf$$

Varijanca kvantizacijske pogreške: $\sigma_e^2 = E(e - E(e))^2$

Za uniformnu grešku: $\sigma_e^2 = E(e^2) = \int_{e} e^2 f_e\, de = \dfrac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} e^2\, de = \dfrac{1}{\Delta} \dfrac{e^3}{3} \Bigg|_{-\Delta/2}^{\Delta/2} = \dfrac{\Delta^2}{12}$

Kvantizacijski omjer signal-šum (SQNR) za zadani broj bitova:

$$SQNR(b) = 10 \log_{10} \frac{\sigma_x^2}{\sigma_e^2} = 6.02 \cdot b + 10 \log_{10} = 6.02 \cdot b + 10 \log_{10} \frac{12 \cdot \sigma_x^2}{A^2} \quad \text{[dB]}$$

Kvantizacijska greška se smanjuje za 6 dB za svaki bit rezolucije!

Limitiranje:

Signal ima efektivnu vrijednost σ_x , za limit unutar dinamike amplituda mora biti ispod $\frac{A}{2}$.

Tjemeni faktor za efektivnu vrijednost: $\delta = \frac{A/2}{\sigma_x}$

Ulazna dinamika bez zasićenja: $A = 2\,\delta\,\sigma_x$

Kvantizacijski korak bez zasićenja: $\Delta = 2\,\delta\,\sigma_x\,2^{-b}$

Direktan SQNR za signal: $SQNR(b) = 20\,b\log_{10}2 + 10\log_{10}\frac{3}{\delta^2}$

Kodiranje:

- source coding se temelji na svojstvima signala
- entropy coding se temelji na statistici uzoraka signala

Izvor signala: $S=\{s_{1,} s_{2,} s_{3,} \ldots\}$
Vjerojatnost simbola: p_i
Količina informacije za simbol: $\log_2 \frac{1}{p_i}$
Entropija – količina informacije koju izvor generira (prosječna duljina simbola)

$$H(S)=-\sum_{i=1}^{n} p_i \log_2 p_i$$

- kodovi varijabilne duljine imaju kraće kodne riječi za vjerojatnije znakove
- prefiksni kodovi nemaju nijednu kodnu riječ koja je početak neke druge

Huffmanov kod

Prefiksni i varijabilno dug kod, dodjeljuje kraće kodne riječi vjerojatnijim znakovima

Algoritam

1. Sortiraj poznate znakove po vjerojatnosti
2. Uzmi dva najmanje vjerojatna simbola, grupiraj stablom u jedan, zbroji im vjerojatnosti i makni ih iz algoritma
3. Ponovi postupak dok se ne istroše svi znakovi i dok korijen nema vjerojatnost 1
4. Dodijeli kodne riječi svakom listu; gornja podgrana svake grane ima vrijednost 0, a donja 1; svaki list grane ima kodnu riječ koja je niz 0 i 1 kojim se od korijena došlo do njega

Ako je entropija cjelobrojna, to je prosječna duljina kodne riječi.
Ako entropija nije cjelobrojna, prosječna duljina kodne riječi je $H(S)+1$

Aritmetičko kodiranje

Cijela poruka se enkodira u dugi binarni broj

Algoritam

1. Uzmi skup ulaznih simbola i pridodaj im $ kao završni znak, te svima pridijeli interval proporcionalan vjerojatnosti tako da im unija spada u [0, 1>
2. Kodiraj simbol tako da njegov interval particioniraš u istim omjerima kao i početni [0, 1>, da bi ponovno dobio proporcionalne dijelove intervala
3. Ponavljaj prošli korak dok ne enkodiraš do zaključno znaka $
4. Kodirana poruka je broj iz krajnjeg intervala dobivenog odabirom $ - enkodiraj broj tako da ga zapišeš kao binarni "decimalni" broj

Vjerojatnost poruke $p_m=\prod_{i}^{s_i=\$} p_i$

Entropija poruke: $H(S)=-\log_2 p_m$

Kodiranje radi kompresije

Kodiranje s konačnom entropijom: kako da poruka stane u kanal uz visoku kvalitetu?
Kodiranje s konačnom distorzijom: kako minimizirati širinu kanala za danu kvalitetu?

ECSQ - entropy-constrained scalar quantization

Korak kvantizacije: $\Delta = \frac{A}{N}$

Kvantizacijski razredi: $C_i = [\Delta i - \Delta/2, \quad \Delta i + \Delta/2\rangle$

Rekonstrukcija - zamjena koda *i* pripadnim centroidom

$$x_{qi} = \Delta \cdot i$$

Vjerojatnost pojedinog simbola *i*:

$$p_I(i) = \int_{\Delta i - \Delta/2}^{\Delta i + \Delta/2} f_X(x) dx$$

Entropija je najveća za proces uniformne gustoće
Distorzija unutar kvantizacijskog razreda:

$$D = \sum_i D_i = \sum_i \int_{\Delta i - \Delta/2}^{\Delta i + \Delta/2} f_X(x)(x - x_{qi})^2 dx$$

USQ (uniform scalar quantization)

$$f_X = \frac{1}{\Delta N}$$

$$p_I = \int_{\Delta i - \Delta/2}^{\Delta i + \Delta/2} \frac{1}{\Delta N} dx = \frac{1}{\Delta N} \int_{\Delta i - \Delta/2}^{\Delta i + \Delta/2} dx = \frac{1}{\Delta N} \Delta = \frac{1}{N}$$

$$max(H(i)) = \log_2(N)$$

Distorzija: $D = \frac{\Delta^2}{12} = \sigma_e^2$

Varijanca ulaznog procesa: $\sigma_x^2 = \frac{(\Delta N)^2}{12}$

$$SQNR(N) = 6.02\, H(I)$$

Za smanjenje distorzije za 6 dB potrebno je povećati entropiju za 1 bit.
Izlazna entropija ne-uniformnog procesa je uvijek manja od entropije uniformnog.

HR-ECSQ – High-Resolution ECSQ

Za dovoljno uske kvantizacijske razrede pdf se može aproksimirati vrijednošću u centroidu razreda!

Uniformni kvantizator Δ je UVIJEK najbolje rješenje!

$$H(I)=h(X)-\frac{1}{2}\log_2(12D),\ D=\frac{\Delta^2}{12}$$

Diferencijalna entropija je mjera kompleksnosti signala:

$$h(X)=-\int_x f_X(x)\log f_X(x)dx$$

Ukupni SQNR: $SQNR=20\log_{10}2\,H(I)+SQNR_0$ [dB]

Offset veze, $SQNR_0$ ovisi o ulaznoj diferencijalnoj entropiji i varijanci:

$$SQNR_0=20\log_{10}(\sqrt{12\cdot\sigma_x^2}\,2^{-h(X)})$$

Veći offset uz istu izlaznu entropiju daje bolju kvalitetu!

Kako odrediti korak HR-ECSQ kvantizatora (Δ_{min}) za zadanu ograničenu entropiju *H (I)*?

$$h(X)=-\int_x f_X(x)\log f_X(x)dx$$

$$\Delta_{min}=2^{h(X)-H(I)}$$

Kako odrediti potrebnu entropiju $H(I)_{min}$ i korak Δ_{min} za zadanu minimalnu kvalitetu $SQNR_{min}$?

$$h(X)=-\int_x f_X(x)\log f_X(x)dx$$

$$SQNR_0=20\log_{10}(\sqrt{12\cdot\sigma_x^2}\,2^{-h(X)})$$

$$H(I)_{min}=\frac{SQNR_{min}-SQNR_0}{20\log_{10}2}$$

$$\Delta=2^{h(X)-H(I)_{min}}$$

Kvantizacijski korak za zadanu izlaznu entropiju H(I) za uniformni proces σ_x je:

$$\Delta_{unif}=\sqrt{12\,\sigma_x^2}\,2^{-H(I)}$$

Linearna predikcija

Cilj je na osnovu poznatog vektorskog procesa ***X*** predvidjeti skalarni proces *Y* (uz određene zavisnosti). Uz primjenu prediktora procjenujemo Y i kodiramo samo razlike!

Poznati vektorski proces dimenzije p	$\vec{\boldsymbol{x}}$
Skalarni proces koji procjenjujemo	y
Procjena skalarnog procesa	$\tilde{y}$
Pogreška procjene	$e=y-\tilde{y}$
Prediktor s minimalnom varijancom greške	$\vec{\boldsymbol{\alpha}}$
Red prediktora (dimenzija prediktora)	p
Broj uzoraka nad kojima se vrši predikcija	N

Princip rada prediktora:

$$\tilde{y}=\left[\vec{\boldsymbol{\alpha}}^T\right]\cdot\left[\begin{array}{c} \\ \vec{\boldsymbol{x}} \\ \\ \end{array}\right]$$

Kovarijancijska simetrična matrica opisuje međuovisnosti između svih parova komponenata ulaznog procesa:

$$\vec{\boldsymbol{\Phi}}_{xx}=E([\vec{\boldsymbol{x}}\cdot\vec{\boldsymbol{x}}^T])=\frac{1}{N}[\vec{\boldsymbol{x}}\cdot\vec{\boldsymbol{x}}^T]$$

Korelacijska matrica opisuje korelacije ciljnog procesa i komponenti izvora ***X***:

$$\vec{\boldsymbol{\Psi}}_{xy}=E[\vec{\boldsymbol{x}}\cdot y]=\frac{1}{N}[\vec{\boldsymbol{x}}\cdot y]$$

Optimalni linearni prediktor:

$$\bar{\boldsymbol{\alpha}}=\boldsymbol{\Phi}_{xx}{}^{-1}\cdot\boldsymbol{\Psi}_{xy}$$

Prediktor reda *p* korelira *p* koraka unazad. Povećanje reda smanjuje ukupnu pogrešku

Prediktor kao filter

Linearni prediktor je FIR (finite-impulse response) stabilni filter *P(z):*

$$P(z)=\sum_{k=1}^{p}\alpha_k z^{-k}$$

Inverzni filter je IIR (infinite-impulse response) filter-rekonstruktor:

Filter greške: $A(z)=1-P(z)$

Rekonstruktor: $H(z)=\dfrac{1}{A(z)}=\dfrac{1}{\sum_{k=1}^{p}\alpha_k z^{-k}}$

H(z) je IIR filter – stabilan je jedino ako su svi polovi unutar jedinične kružnice, tj. ako su sve nultočke po modulu manje od 1.

Autokorelacija

Za osiguravanje stabilnosti IIR rekonstruktora koristi se metoda autokorelacije. *R(j)* je autokorelacija $x[n]$ sa uzorcima na indeksima za korak *j*.

$$R(j)=\sum_{m=0}^{N-1-j} x[m]x[m+j]$$

Formira se matrica iz koje se izračuna stabilni IIR rekonstruktor.

$$\begin{bmatrix} R(0) & R(1) & R(2) & \cdots & R(p-1) \\ R(1) & R(0) & R(1) & \cdots & R(p-2) \\ R(2) & R(1) & R(0) & \cdots & R(p-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R(p-1) & R(p-2) & R(p-3) & \cdots & R(0) \end{bmatrix} \cdot \begin{bmatrix} \bar{\alpha}_1 \\ \bar{\alpha}_2 \\ \bar{\alpha}_3 \\ \vdots \\ \bar{\alpha}_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ R(3) \\ \vdots \\ R(p) \end{bmatrix}$$

Vrste LPC kodera

OLP (Open-Loop Prediction) koder

Izvorni proces korišten je sličan, ali ne isti kao rekonstrukcijski.
Pogreška nastaje pri kvantizaciji $e[n] \rightarrow \hat{e}[n]$ i razlici u predikciji dekodera i enkodera.
Međutim, šum se također rekonstruira s parametrima glasa, pa se "skriva" iza signala (proces noise shapinga)

CLP (Closed-Loop Prediction) koder

Originalni proces je dostupan na obje strane koristi se predkvantizirani signal koji se lokalno rekonstruira i tvore isti proces predikcije. Međutim, do noise maskinga ne dolazi.

Adaptacija

Koder mora s vremenom slati ažurni prediktor:,to može napraviti kao:

FAP - forward-adaptive prediction (šalje se uz tok samog kodiranog signala)
BAP - backward-adaptive prediction (proces CLPa)

Kodiranje govora

Dugotrajna korelacija potrebna je za pojedine glasove (od 3 do 15 ms trajanja)
Kratkotrajna korelacija potrebna je unutar glasa (unutar 1 ms)
Formanti - rezonantne karakteristike glasa (frekvencije u pojedinom glasu)

Za analizu se koristi spektrogram
narrowband za dugotrajne korelacije
wideband za kratkotrajne korelacije

Princip rada LPC VoCodera:

1) Primijeni LPC na 20-30 ms glasa i nađi prediktor $P(z)$ reda p s koeficijentima $\vec{\alpha}_1$ do $\vec{\alpha}_p$
2) Izračunaj signal predikcijske pogreške $e[n]$, varijancu $\sigma_e^2(\bar{\boldsymbol{\alpha}})$ te faktor pojačanja $G=\sigma_e(\bar{\boldsymbol{\alpha}})$
3) Za pogrešku potraži periodičnost i ovisno o tome koristi sinus ili bijeli šum (autokorelacijom)
4) Kvantiziraj procijenjene parametre i ubaci uz poruku
5) Pomakni fokus za 10-20 ms na idući frame (framerate 50 do 100 fps)

Princip rada LPC VoDecodera:

1) Rekonstuiraj parametre
2) Formiraj segment signala jediničnim impulsima s razmakom periode ili uzorkom šuma
3) Pomnoži signal s G da bi se dobio sintetički $e[n]$ na ulazu H(z)
4) Propusti signal kroz $H(z)$ s parametrima koji su poslani ($\hat{\alpha}_k=Q(\bar{\alpha}_k)$)
5) Pusti segment sintetskog glasa i ponovi postupak za idući frame

OLP (Open-Loop Prediction) koder (ponovno)

Pogreška kvantizacije se prolaskom kroz H(z) na drugoj strani spektralno uobličuje (noise shaping) i zbog spektralne sličnosti je manje primjetna nego direktno kvantizirani glas (frequency masking). SQNR se ne povećava, ali se energija pogreške frekvencijski bolje raspodjeljuje.

CLP (Closed-Loop Prediction) koder (ponovno)

Pogreška rekonstrukcije je jednaka pogrešci kvantizacije, a lakše je kvantizirati grešku jer ima manju varijancu od signala. Kvaliteta se poboljšava samo smanjenjem energije pogreške, ne noise shapingom.

Prediction gain je poboljšanje SQNRa predviđenog u odnosu na kvantizirani signal.

$$PG=10\log_{10}\frac{\sigma_x^2}{\sigma_{ec}^2}\ [\mathrm{dB}]$$

Problem je što ne možemo odrediti grešku CLPa jer za prediktor trebamo kvantizator i obratno. Prediktor se stoga optimira za mod rada blizu OLPu, gdje je idealno $H(I)=\infty$.

$$PG_\infty=10\log_{10}\frac{\sigma_x^2}{\sigma_e^2}\ [\mathrm{dB}]$$