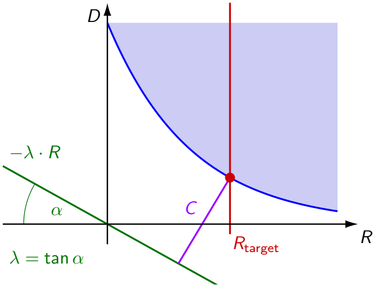
Optimal Scalar Quantization



Last Lectures: Lloyd Quantizer & Rate-Distortion Function

Lloyd Quantizer

- Minimizes distortion D for given number K of quantization intervals
- Design algorithm: Iterate between two optimization criterions
  - Centroid condition (MSE): $s'_k = \mathrm{E}\{\, S \,|\, S \in \mathcal{I}_k \,\}$
  - Nearest neighbor condition (MSE): $u_k = (s'_k + s'_{k-1})/2$

Last Lectures: Lloyd Quantizer & Rate-Distortion Function

Lloyd Quantizer

- Minimizes distortion D for given number K of quantization intervals
- Design algorithm: Iterate between two optimization criterions
  - Centroid condition (MSE): $s'_k = \mathrm{E}\{\, S \,|\, S \in \mathcal{I}_k \,\}$
  - Nearest neighbor condition (MSE): $u_k = (s'_k + s'_{k-1})/2$

Rate-Distortion Function

- Greatest lower bound for lossy source coding: $R(D) = \lim_{N\to\infty} \inf_{g_N:\ \delta_N(g_N)\le D} \dfrac{I_N(g_N)}{N}$
- Property of the source (no consideration of codes)

Last Lectures: Lloyd Quantizer & Rate-Distortion Function

Lloyd Quantizer

- Minimizes distortion D for given number K of quantization intervals
- Design algorithm: Iterate between two optimization criterions
  - Centroid condition (MSE): $s'_k = \mathrm{E}\{\, S \,|\, S \in \mathcal{I}_k \,\}$
  - Nearest neighbor condition (MSE): $u_k = (s'_k + s'_{k-1})/2$

Rate-Distortion Function

- Greatest lower bound for lossy source coding:
- Property of the source (no consideration of codes)

$$R(D) = \lim_{N\to\infty} \inf_{g_N:\; \delta_N(g_N)\leq D} \frac{I_N(g_N)}{N}$$

High-Rate Approximations

- Panter & Dite asymptote for Lloyd quantizer (MSE): $D_F(R) = \varepsilon_F^2 \cdot \sigma^2 \cdot 2^{-2R}$
- Shannon lower bound for rate-distortion function (MSE): $D_L(R) = \varepsilon_L^2 \cdot \sigma^2 \cdot 2^{-2R}$

Last Lectures: Lloyd Quantizer & Rate-Distortion Function

Lloyd Quantizer

- Minimizes distortion D for given number K of quantization intervals
- Design algorithm: Iterate between two optimization criterions
  - Centroid condition (MSE): $s'_k = \mathrm{E}\{\, S \,|\, S \in \mathcal{I}_k \,\}$
  - Nearest neighbor condition (MSE): $u_k = (s'_k + s'_{k-1})/2$

Rate-Distortion Function

- Greatest lower bound for lossy source coding:
- Property of the source (no consideration of codes)

$$R(D) = \lim_{N\to\infty} \inf_{g_N:\ \delta_N(g_N)\le D} \frac{I_N(g_N)}{N}$$

High-Rate Approximations

- Panter & Dite asymptote for Lloyd quantizer (MSE): $D_F(R) = \varepsilon_F^2 \cdot \sigma^2 \cdot 2^{-2R}$
- Shannon lower bound for rate-distortion function (MSE): $D_L(R) = \varepsilon_L^2 \cdot \sigma^2 \cdot 2^{-2R}$

➡ MSE distortion increase of Lloyd vs SLB: $\dfrac{D_F}{D_L}(R) = \dfrac{\varepsilon_F^2}{\varepsilon_L^2}$ ➡ Gauss: ≈ 2.72 (4.34 dB); Laplace: ≈ 5.20 (7.16 dB)

Rate-Distortion Efficiency of Scalar Quantizers

Distortion

- Quantifies deviation between original and reconstructed samples
- Typically: Additive distortion measures with $d(s, s')$ being the single-sample distortion

$$D = \mathrm{E}\{\, d(S, S')\,\} = \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s$$

Rate-Distortion Efficiency of Scalar Quantizers

Distortion

- Quantifies deviation between original and reconstructed samples
- Typically: Additive distortion measures with $d(s, s')$ being the single-sample distortion

$$D = \mathrm{E}\{\, d(S, S')\,\} = \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s$$

Bit Rate

- Average number of bits for coding quantization indexes q (or reconstructed values s')

$$R = \mathrm{E}\{\, \ell(S')\,\} = \sum_{\forall k} p_k \cdot \ell_k = \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

Rate-Distortion Efficiency of Scalar Quantizers

Distortion

- Quantifies deviation between original and reconstructed samples
- Typically: Additive distortion measures with $d(s, s')$ being the single-sample distortion

$$D = \mathrm{E}\{\, d(S, S') \,\} = \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s$$

Bit Rate

- Average number of bits for coding quantization indexes q (or reconstructed values s')

$$R = \mathrm{E}\{\, \ell(S') \,\} = \sum_{\forall k} p_k \cdot \ell_k = \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

Design of Scalar Quantizers

- Lloyd quantizer minimizes distortion, but ignores impact on bit rate (assumes fixed-length coding)

Rate-Distortion Efficiency of Scalar Quantizers

Distortion

- Quantifies deviation between original and reconstructed samples
- Typically: Additive distortion measures with $d(s, s')$ being the single-sample distortion

$$D = \mathrm{E}\{\, d(S, S')\,\} = \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s$$

Bit Rate

- Average number of bits for coding quantization indexes q (or reconstructed values s')

$$R = \mathrm{E}\{\, \ell(S')\,\} = \sum_{\forall k} p_k \cdot \ell_k = \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

Design of Scalar Quantizers

- Lloyd quantizer minimizes distortion, but ignores impact on bit rate (assumes fixed-length coding)

➡ Improved performance: **Consider bit rate in quantizer design**

Joint Minimization of Distortion and Bit Rate

Constrained Optimization Problem

- Optimization problem can be formulated as

$$\min \; D \qquad \text{subject to} \qquad R \le R_{\text{target}}$$

or, equivalently,

$$\min \; R \qquad \text{subject to} \qquad D \le D_{\text{target}}$$

Joint Minimization of Distortion and Bit Rate

Constrained Optimization Problem

- Optimization problem can be formulated as

$$\min \; D \qquad \text{subject to} \qquad R \leq R_{\text{target}}$$

or, equivalently,

$$\min \; R \qquad \text{subject to} \qquad D \leq D_{\text{target}}$$

Reformulation as Unconstrained Problem

- Typically, constrained optimization problems cannot be solved directly
- Use technique of Lagrange multipliers for reformulation as unconstrained problem

$$\min \; D + \lambda \cdot R$$

Joint Minimization of Distortion and Bit Rate

Constrained Optimization Problem

- Optimization problem can be formulated as

$$\min\ D \qquad \text{subject to} \qquad R \leq R_{\text{target}}$$

or, equivalently,

$$\min\ R \qquad \text{subject to} \qquad D \leq D_{\text{target}}$$

Reformulation as Unconstrained Problem

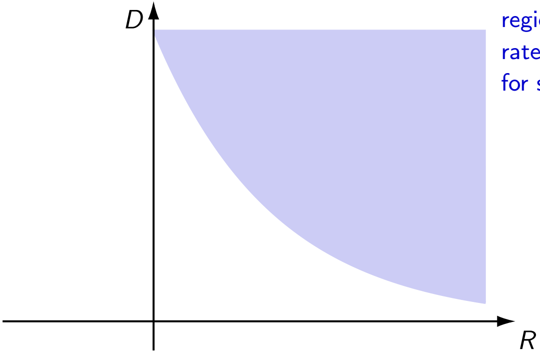
- Typically, constrained optimization problems cannot be solved directly
- Use technique of Lagrange multipliers for reformulation as unconstrained problem

$$\min\ D + \lambda \cdot R$$

➡ The parameter $\lambda > 0$ is called **Lagrange multiplier**

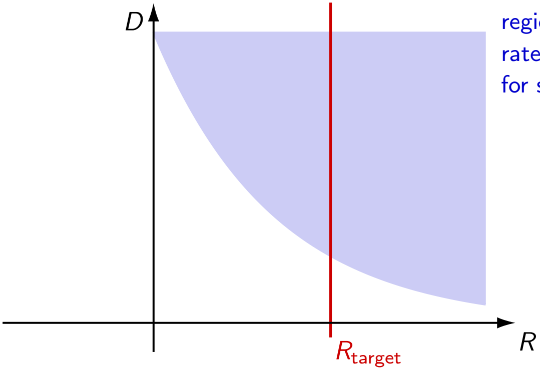
➡ Each value of λ corresponds to a rate constraint R_{target} (or distortion constraint D_{target})

Convex Optimization: Illustration



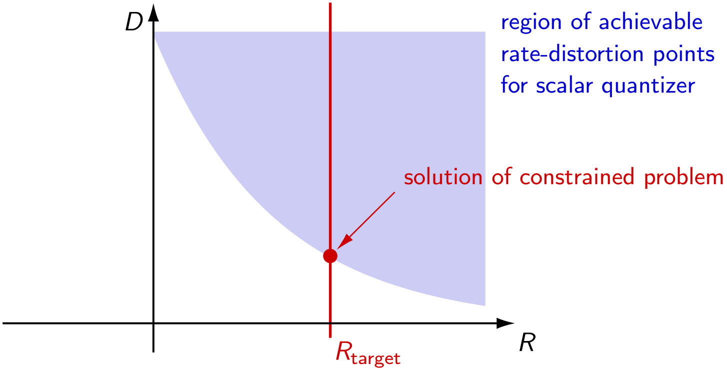
region of achievable rate-distortion points for scalar quantizer

Convex Optimization: Illustration

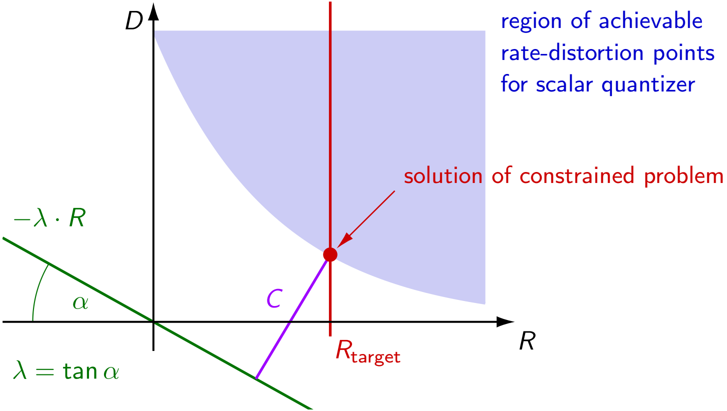


region of achievable rate-distortion points for scalar quantizer

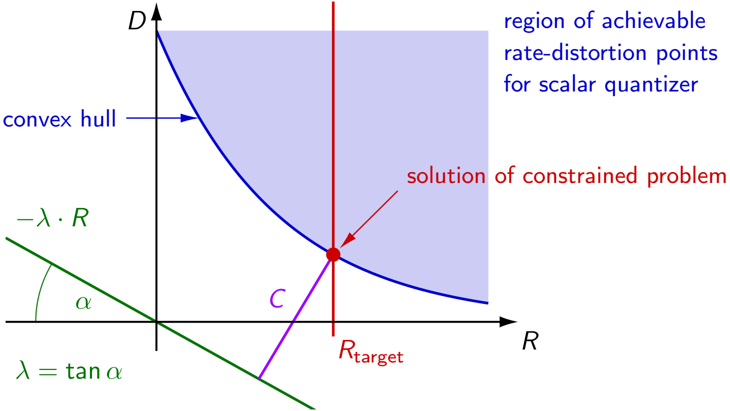
Convex Optimization: Illustration



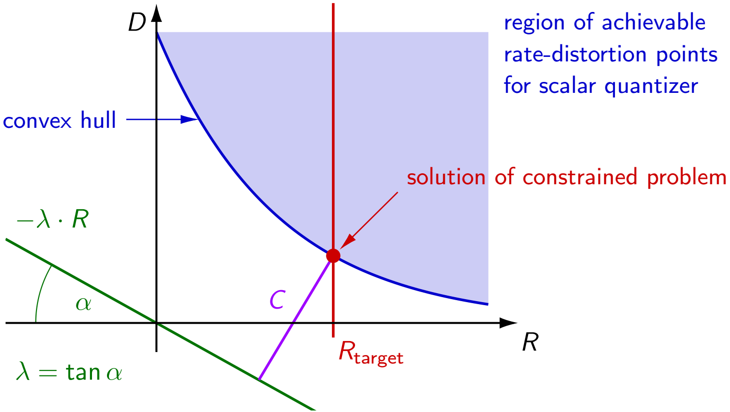
Convex Optimization: Illustration



Convex Optimization: Illustration

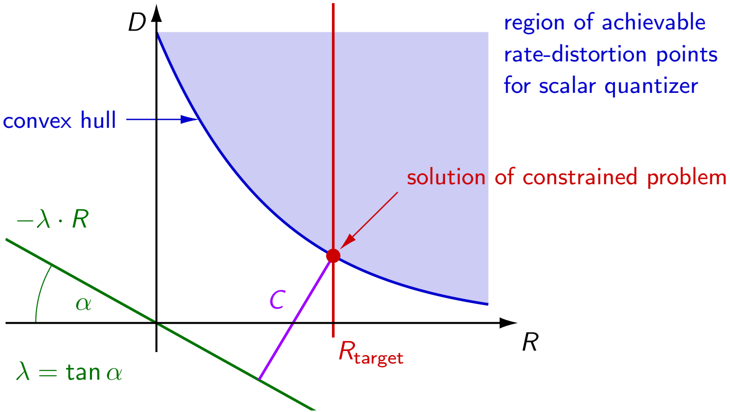


Convex Optimization: Illustration



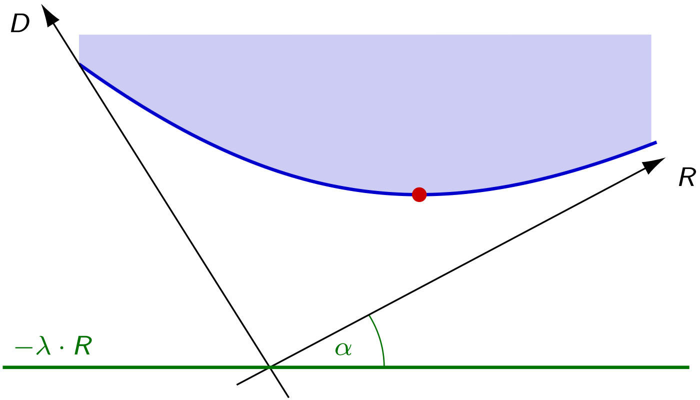
➡ Points on convex hull: Minimize distance C to line $D = -\lambda \cdot R$

Convex Optimization: Illustration

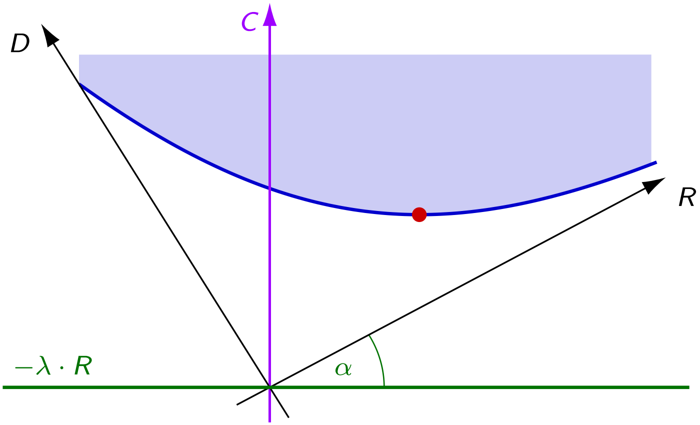


- ➡ Points on convex hull: Minimize distance C to line $D = -\lambda \cdot R$
- ➡ Geometrical interpretation: Rotate coordinate system by angle α

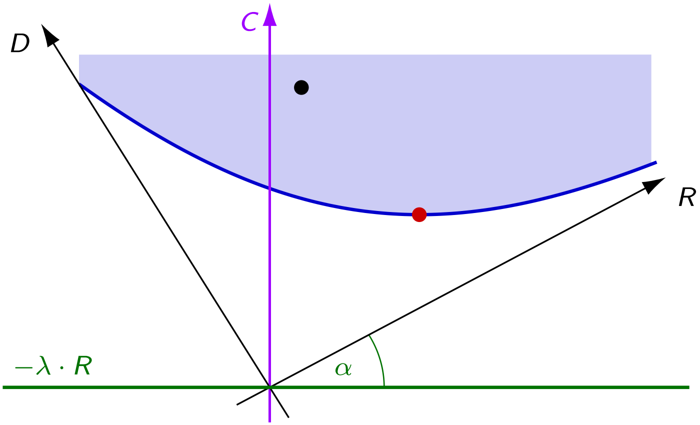
Convex Optimization: Lagrangian Formulation



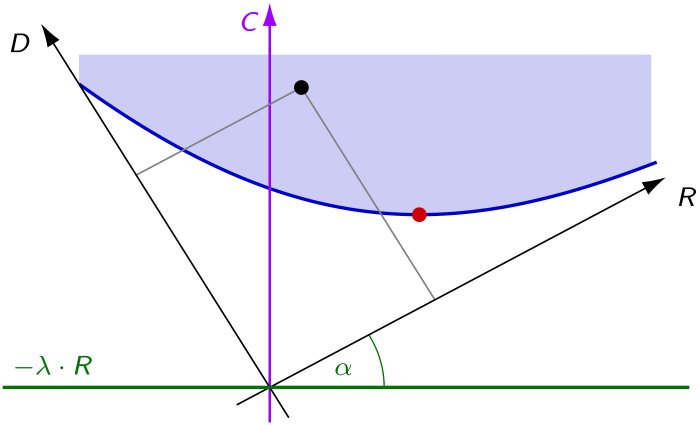
Convex Optimization: Lagrangian Formulation



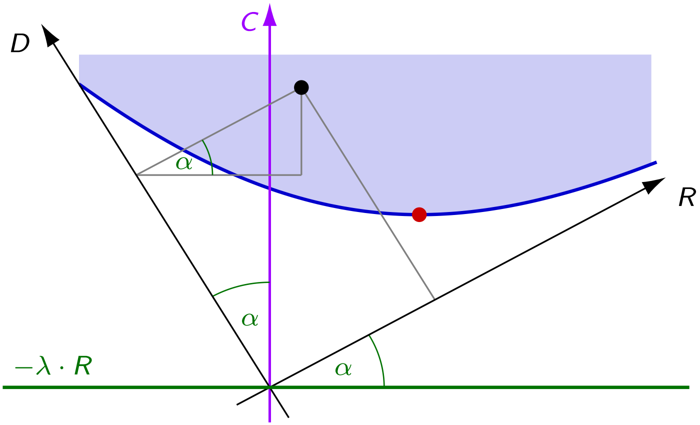
Convex Optimization: Lagrangian Formulation



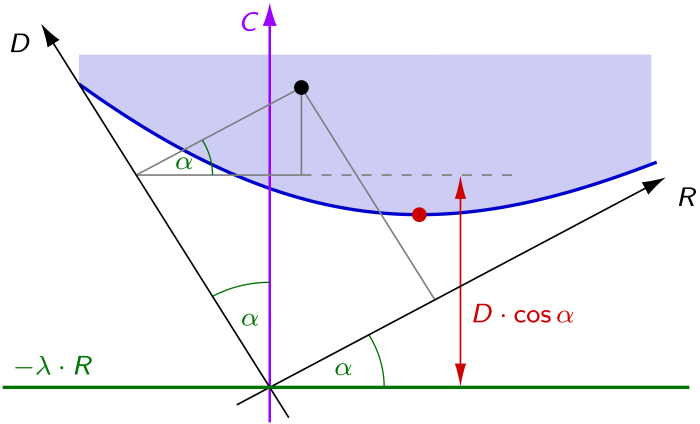
Convex Optimization: Lagrangian Formulation



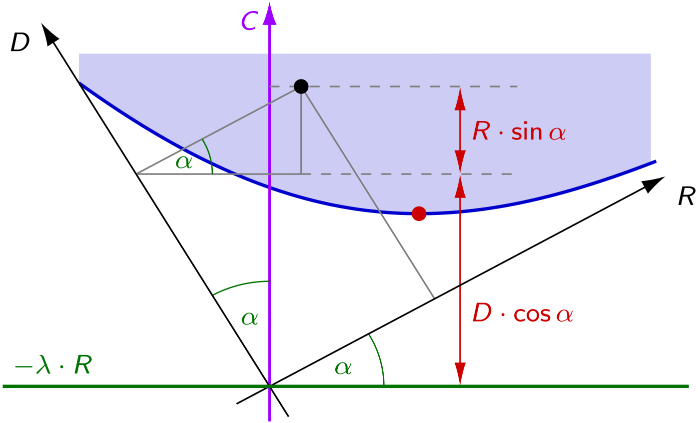
Convex Optimization: Lagrangian Formulation



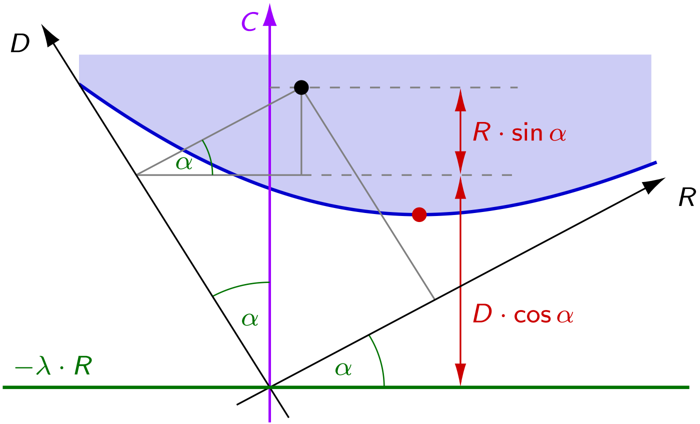
Convex Optimization: Lagrangian Formulation



Convex Optimization: Lagrangian Formulation

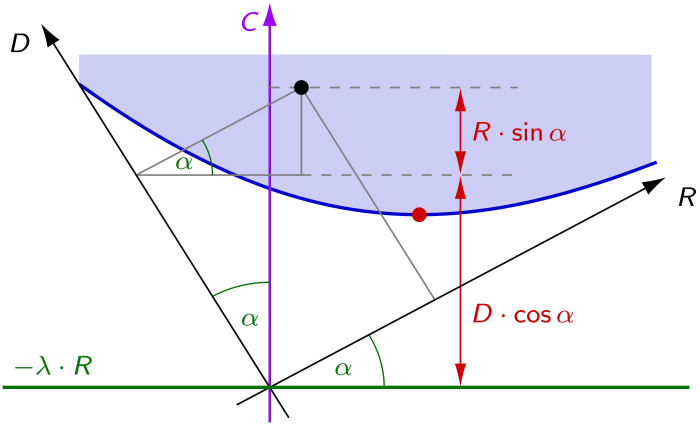


Convex Optimization: Lagrangian Formulation



➡ Minimize distance: $C = D \cdot \cos\alpha + R \cdot \sin\alpha$

Convex Optimization: Lagrangian Formulation



➡ Minimize distance: $C = D \cdot \cos\alpha + R \cdot \sin\alpha$

➡ Equivalent minimization: $J = D + \lambda \cdot R$ (note: Lagrange multiplier is given by $\lambda = \tan\alpha$)

Optimal Scalar Quantizer

Optimization Criterion

- Minimization of Lagrangian cost for some given Lagrange multiplier λ

$$J = D + \lambda \cdot R$$

Optimal Scalar Quantizer

Optimization Criterion

- Minimization of Lagrangian cost for some given Lagrange multiplier λ

$$
\begin{aligned}
J &= D + \lambda \cdot R \\
&= \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s + \lambda \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s \qquad \text{(any additive distortion measure)}
\end{aligned}
$$

Optimal Scalar Quantizer

Optimization Criterion

- Minimization of Lagrangian cost for some given Lagrange multiplier λ

$$\begin{aligned} J &= D + \lambda \cdot R \\ &= \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s + \lambda \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s \qquad \text{(any additive distortion measure)} \end{aligned}$$

➡ Each Lagrange multiplier $\lambda > 0$ yields solution of original constrained problem for one R_{target}

Optimal Scalar Quantizer

Optimization Criterion

- Minimization of Lagrangian cost for some given Lagrange multiplier λ

$$J = D + \lambda \cdot R$$
$$= \sum_{\forall k} \int_{u_k}^{u_{k+1}} d(s, s'_k)\, f(s)\, \mathrm{d}s + \lambda \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s \qquad \text{(any additive distortion measure)}$$

➡ Each Lagrange multiplier $\lambda > 0$ yields solution of original constrained problem for one R_{target}

Optimization Criterion for MSE Distortion

- Determine quantizer parameters (s'_k and u_k) and codeword lengths ℓ_k such that the Lagrangian cost $J = D + \lambda R$ for MSE distortion is minimized

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2\, f(s)\, \mathrm{d}s + \lambda \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

Optimal Decoder Mapping: Centroid Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

1 **Optimize reconstruction levels s'_k** for given decision thresholds u_k

Optimal Decoder Mapping: Centroid Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

1. **Optimize reconstruction levels s'_k** for given decision thresholds u_k
   - Note: Rate term does not depend on reconstruction levels s'_k

Optimal Decoder Mapping: Centroid Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

1. **Optimize reconstruction levels s'_k** for given decision thresholds u_k
   - Note: Rate term does not depend on reconstruction levels s'_k

   ➡ Same condition as for Lloyd quantizer

Optimal Decoder Mapping: Centroid Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 f(s)\, \mathrm{d}s \;+\; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

1 Optimize reconstruction levels s'_k for given decision thresholds u_k

- Note: Rate term does not depend on reconstruction levels s'_k

➡ Same condition as for Lloyd quantizer

➡ Centroid condition for MSE distortion

- Optimal reconstruction level s'_k is given by conditional mean

$$\boxed{s'_k = \mathrm{E}\{\, S \mid S \in \mathcal{I}_k \,\} = \int_{u_k}^{u_{k+1}} s\, f(s \,|\, s \in \mathcal{I}_k)\, \mathrm{d}s = \frac{1}{p_k} \int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s = \frac{\displaystyle\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\displaystyle\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}}$$

Optimal Lossless Mapping: Entropy Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

2 **Optimize codeword lengths ℓ_k** for given decision thresholds u_k

Optimal Lossless Mapping: Entropy Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

2 Optimize codeword lengths ℓ_k for given decision thresholds u_k

- Note: Distortion term does not depend on codeword lengths ℓ_k

Optimal Lossless Mapping: Entropy Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

2 Optimize codeword lengths ℓ_k for given decision thresholds u_k

- Note: Distortion term does not depend on codeword lengths ℓ_k

➡ Remember: Lossless coding theorem: $R \geq H(S')$

$$\sum_{\forall k} p_k \cdot \ell_k \geq -\sum_{\forall k} p_k \cdot \log_2 p_k \qquad \text{(equality if and only if } \ell_k = -\log_2 p_k\text{)}$$

Optimal Lossless Mapping: Entropy Condition

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

2 Optimize codeword lengths ℓ_k for given decision thresholds u_k

- Note: Distortion term does not depend on codeword lengths ℓ_k

➡ Remember: Lossless coding theorem: $R \geq H(S')$

$$\sum_{\forall k} p_k \cdot \ell_k \geq -\sum_{\forall k} p_k \cdot \log_2 p_k \qquad \text{(equality if and only if } \ell_k = -\log_2 p_k\text{)}$$

➡ **Entropy condition** (neglecting inefficiency of actual entropy coding)

$$\boxed{\ell_k = -\log_2 p_k = -\log_2 \left(\int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s \right)}$$

Optimal Encoder Mapping: Condition for Decision Thresholds

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

3 **Optimize decision thresholds** u_k for given reconstruction levels s'_k and codeword lengths ℓ_k

Optimal Encoder Mapping: Condition for Decision Thresholds

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

3 Optimize decision thresholds u_k for given reconstruction levels s'_k and codeword lengths ℓ_k

- Note: Each threshold u_k impacts only neighbouring intervals $\mathcal{I}_{k-1}$ and $\mathcal{I}_k$

Optimal Encoder Mapping: Condition for Decision Thresholds

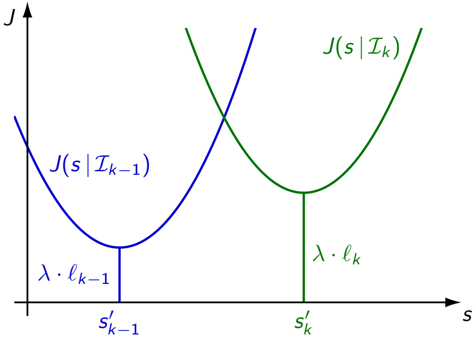
$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

3 Optimize decision thresholds u_k for given reconstruction levels s'_k and codeword lengths ℓ_k

- Note: Each threshold u_k impacts only neighbouring intervals $\mathcal{I}_{k-1}$ and $\mathcal{I}_k$

➡ Map each input value s to interval $\mathcal{I}_k$ that minimizes contribution to Lagrangian cost

$$J(s \,|\, \mathcal{I}_k) = (s - s'_k)^2 + \lambda \cdot \ell_k$$



Optimal Encoder Mapping: Condition for Decision Thresholds

$$J = \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - s'_k)^2 \, f(s) \, \mathrm{d}s \; + \; \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

3 Optimize decision thresholds u_k for given reconstruction levels s'_k and codeword lengths ℓ_k

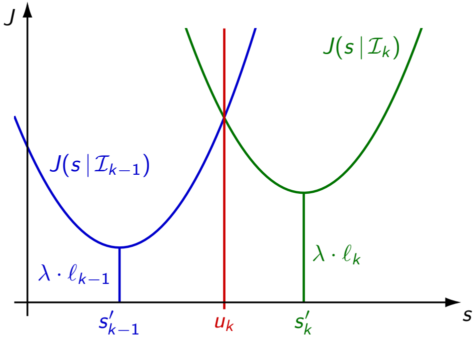
- Note: Each threshold u_k impacts only neighbouring intervals $\mathcal{I}_{k-1}$ and $\mathcal{I}_k$

➡ Map each input value s to interval $\mathcal{I}_k$ that minimizes contribution to Lagrangian cost

$$J(s \,|\, \mathcal{I}_k) = (s - s'_k)^2 + \lambda \cdot \ell_k$$

➡ At decision threshold u_k, we require

$$J(u_k \,|\, \mathcal{I}_{k-1}) = J(u_k \,|\, \mathcal{I}_k)$$



Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$J(u_k \,|\, \mathcal{I}_{k-1}) \;=\; J(u_k \,|\, \mathcal{I}_k)$$

Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$
\begin{aligned}
J(u_k \,|\, \mathcal{I}_{k-1}) &= J(u_k \,|\, \mathcal{I}_k) \\
(u_k - s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= (u_k - s'_k)^2 + \lambda \cdot \ell_k
\end{aligned}
$$

Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$
\begin{aligned}
J(u_k \,|\, \mathcal{I}_{k-1}) &= J(u_k \,|\, \mathcal{I}_k) \\
(u_k - s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= (u_k - s'_k)^2 + \lambda \cdot \ell_k \\
u_k^2 - 2u_k s'_{k-1} + (s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= u_k^2 - 2u_k s'_k + (s'_k)^2 + \lambda \cdot \ell_k
\end{aligned}
$$

Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$
\begin{aligned}
J(u_k \,|\, \mathcal{I}_{k-1}) &= J(u_k \,|\, \mathcal{I}_k) \\
(u_k - s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= (u_k - s'_k)^2 + \lambda \cdot \ell_k \\
u_k^2 - 2u_k s'_{k-1} + (s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= u_k^2 - 2u_k s'_k + (s'_k)^2 + \lambda \cdot \ell_k \\
2u_k(s'_k - s'_{k-1}) &= (s'_k)^2 - (s'_{k-1})^2 + \lambda\,(\ell_k - \ell_{k-1})
\end{aligned}
$$

Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$
\begin{aligned}
J(u_k \,|\, \mathcal{I}_{k-1}) &= J(u_k \,|\, \mathcal{I}_k) \\
(u_k - s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= (u_k - s'_k)^2 + \lambda \cdot \ell_k \\
u_k^2 - 2u_k s'_{k-1} + (s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= u_k^2 - 2u_k s'_k + (s'_k)^2 + \lambda \cdot \ell_k \\
2u_k(s'_k - s'_{k-1}) &= (s'_k)^2 - (s'_{k-1})^2 + \lambda\,(\ell_k - \ell_{k-1}) \\
2u_k(s'_k - s'_{k-1}) &= (s'_k - s'_{k-1})(s'_k + s'_{k-1}) + \lambda\,(\ell_k - \ell_{k-1})
\end{aligned}
$$

Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$
\begin{aligned}
J(u_k \,|\, \mathcal{I}_{k-1}) &= J(u_k \,|\, \mathcal{I}_k) \\
(u_k - s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= (u_k - s'_k)^2 + \lambda \cdot \ell_k \\
u_k^2 - 2u_k s'_{k-1} + (s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= u_k^2 - 2u_k s'_k + (s'_k)^2 + \lambda \cdot \ell_k \\
2u_k(s'_k - s'_{k-1}) &= (s'_k)^2 - (s'_{k-1})^2 + \lambda\,(\ell_k - \ell_{k-1}) \\
2u_k(s'_k - s'_{k-1}) &= (s'_k - s'_{k-1})(s'_k + s'_{k-1}) + \lambda\,(\ell_k - \ell_{k-1})
\end{aligned}
$$

➜ **Modified nearest neighbour condition for MSE distortion**

$$
\boxed{u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)}
$$

Optimal Encoder Mapping: Modified Nearest Neighbour Condition

- Optimal encoding mapping for MSE distortion

$$
\begin{aligned}
J(u_k \,|\, \mathcal{I}_{k-1}) &= J(u_k \,|\, \mathcal{I}_k) \\
(u_k - s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= (u_k - s'_k)^2 + \lambda \cdot \ell_k \\
u_k^2 - 2u_k s'_{k-1} + (s'_{k-1})^2 + \lambda \cdot \ell_{k-1} &= u_k^2 - 2u_k s'_k + (s'_k)^2 + \lambda \cdot \ell_k \\
2u_k(s'_k - s'_{k-1}) &= (s'_k)^2 - (s'_{k-1})^2 + \lambda\,(\ell_k - \ell_{k-1}) \\
2u_k(s'_k - s'_{k-1}) &= (s'_k - s'_{k-1})(s'_k + s'_{k-1}) + \lambda\,(\ell_k - \ell_{k-1})
\end{aligned}
$$

➡ **Modified nearest neighbour condition for MSE distortion**

$$
\boxed{u_k = \frac{1}{2}\,(s'_{k-1} + s'_k) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)}
$$

➡ **Threshold is shifted towards the reconstruction level with the longer codeword**

Entropy-Constrained Lloyd Quantizer: Minimization of Lagrangian Cost

Necessary Conditions for Optimality (MSE distortion)

1. Centroid condition

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

2. Entropy condition

$$\ell_k = -\log_2 p_k$$

3. Modified nearest neighbor condition

$$u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)$$

Entropy-Constrained Lloyd Quantizer: Minimization of Lagrangian Cost

Necessary Conditions for Optimality (MSE distortion)

1. Centroid condition

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

2. Entropy condition

$$\ell_k = -\log_2 p_k$$

3. Modified nearest neighbor condition

$$u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)$$

Design of optimal entropy-constrained scalar quantizers

- In general: Cannot be derived in closed form

➡ Iterative algorithm similar to Lloyd algorithm

Entropy-Constrained Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:

- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Entropy-Constrained Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$

Entropy-Constrained Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)$$

Entropy-Constrained Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)$$

3. Update the reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$ according to

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\,\mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s} \qquad \text{and} \qquad \ell_k = -\log_2\left(\int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s\right)$$

Entropy-Constrained Lloyd Algorithm for Given Pdf (MSE Distortion)

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \frac{1}{2}\left(s'_{k-1} + s'_k\right) + \frac{\lambda}{2}\left(\frac{\ell_k - \ell_{k-1}}{s'_k - s'_{k-1}}\right)$$

3. Update the reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$ according to

$$s'_k = \frac{\int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s} \qquad \text{and} \qquad \ell_k = -\log_2\left(\int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s\right)$$

4. Repeat the previous two steps until convergence

Entropy-Constrained Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:
- a sufficiently large realization $\{s_n\}$ of the considered source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$

Entropy-Constrained Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:
- a sufficiently large realization $\{s_n\}$ of the considered source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$
2. Associate all samples of the training set $\{s_n\}$ with one of quantization intervals $\mathcal{I}_k$

$$q(s_n) = \arg\min_{\forall k} \; (s_n - s'_k)^2 + \lambda \cdot \ell_k$$

Entropy-Constrained Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:
- a sufficiently large realization $\{s_n\}$ of the considered source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$
2. Associate all samples of the training set $\{s_n\}$ with one of quantization intervals $\mathcal{I}_k$

$$q(s_n) = \arg\min_{\forall k} \; (s_n - s'_k)^2 + \lambda \cdot \ell_k$$

3. Update the reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$ according to

$$s'_k = \frac{1}{N_k} \sum_{n:\; q(s_n)=k} s_n \qquad \text{and} \qquad \ell_k = -\log_2\left(\frac{N_k}{N}\right)$$

where N_k is the number of samples associated with $\mathcal{I}_k$ and N is the total number of samples

Entropy-Constrained Lloyd Algorithm for a Training Set (MSE Distortion)

Given is:

- a sufficiently large realization $\{s_n\}$ of the considered source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial set of reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$
2. Associate all samples of the training set $\{s_n\}$ with one of quantization intervals $\mathcal{I}_k$

$$q(s_n) = \arg \min_{\forall k} \; (s_n - s'_k)^2 + \lambda \cdot \ell_k$$

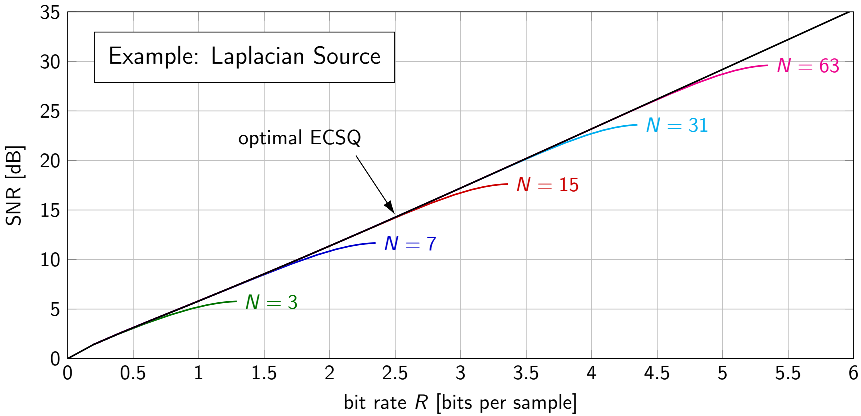
3. Update the reconstruction levels $\{s'_k\}$ and codeword lengths $\{\ell_k\}$ according to

$$s'_k = \frac{1}{N_k} \sum_{n:\; q(s_n)=k} s_n \qquad \text{and} \qquad \ell_k = -\log_2\left(\frac{N_k}{N}\right)$$

   where N_k is the number of samples associated with $\mathcal{I}_k$ and N is the total number of samples

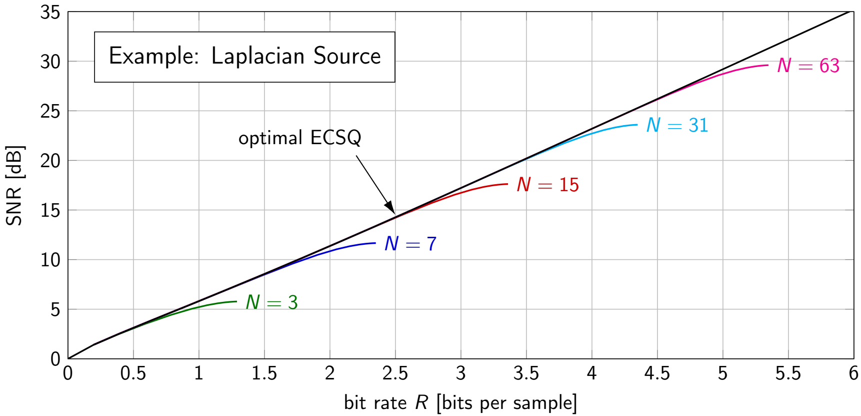
4. Repeat the previous two steps until convergence

Number of Initial Intervals for Entropy-Constrained Lloyd Algorithm



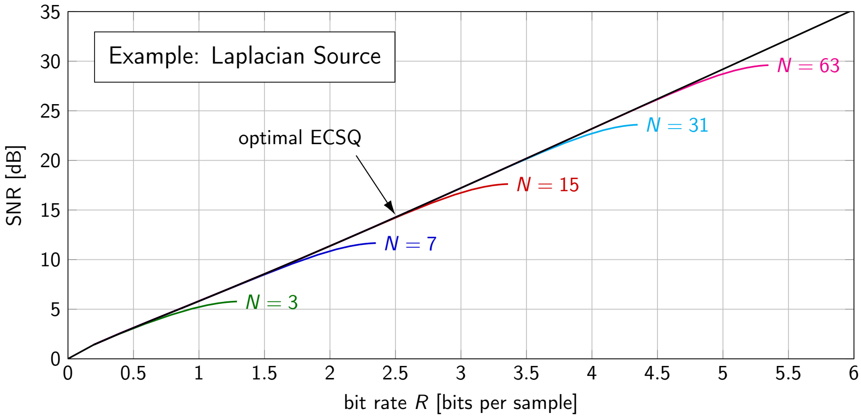
- Too small number of intervals leads to sub-optimal design

Number of Initial Intervals for Entropy-Constrained Lloyd Algorithm



- Too small number of intervals leads to sub-optimal design
- EC Lloyd algorithm removes intervals during iterations (probabilities get smaller and smaller)

Number of Initial Intervals for Entropy-Constrained Lloyd Algorithm



- Too small number of intervals leads to sub-optimal design
- EC Lloyd algorithm removes intervals during iterations (probabilities get smaller and smaller)

➜ **Use large number of intervals in initialization**

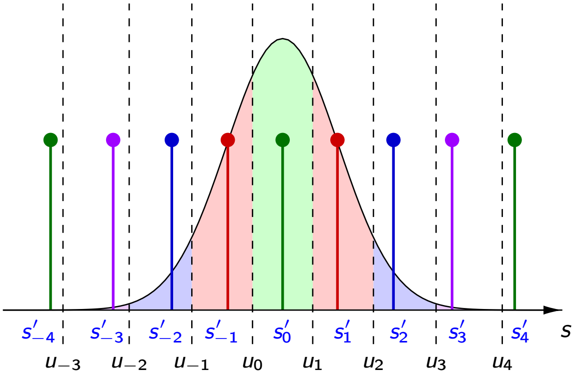
Example: EC Lloyd Algorithm for Gaussian Source

Gaussian Source

- Zero mean $\mu = 0$
- Unit variance $\sigma^2 = 1$

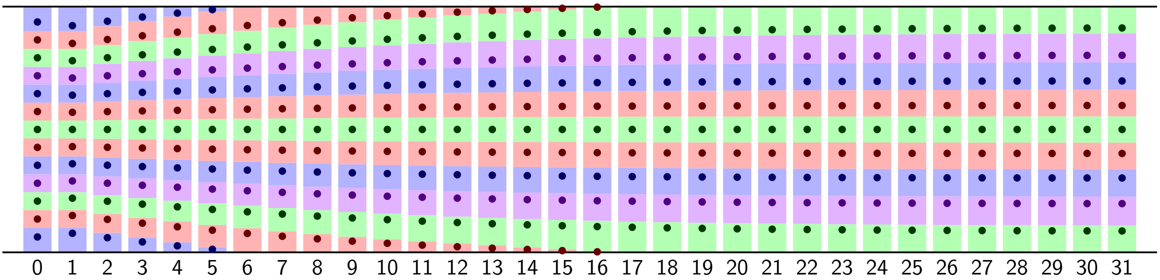
EC Lloyd Quantizer for 2 bits per sample

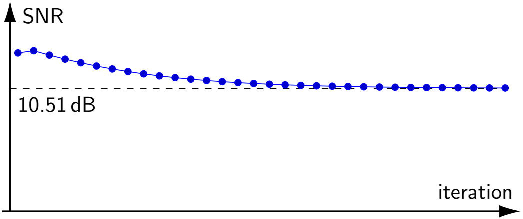
- Decision thresholds:
$$\begin{aligned} u_{0/1} &= \pm 0.538 \\ u_{-1/2} &= \pm 1.623 \\ u_{-2/3} &= \pm 2.743 \\ u_{-3/4} &= \pm 3.926 \end{aligned}$$
- Reconstruction levels:
$$\begin{aligned} s'_0 &= 0.000 \\ s'_{\pm 1} &= \pm 0.980 \\ s'_{\pm 2} &= \pm 1.981 \\ s'_{\pm 3} &= \pm 3.029 \\ s'_{\pm 4} &= \pm 4.148 \end{aligned}$$

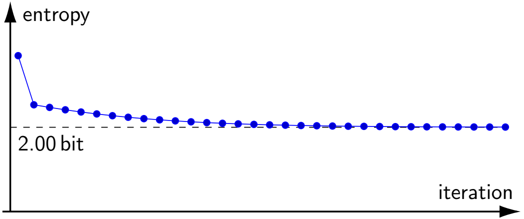


$$\begin{aligned} R &= 2.00 && \text{(rate = entropy)} \\ D &= 0.089 \\ \text{SNR} &= 10.51\,\text{dB} && \text{(RD bound} = 12.04\,\text{dB)} \end{aligned}$$

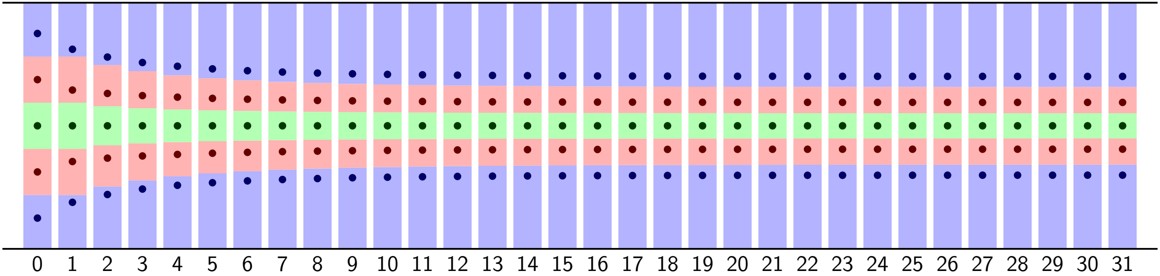
Example: Convergence of EC Lloyd Algorithm for Gaussian Source

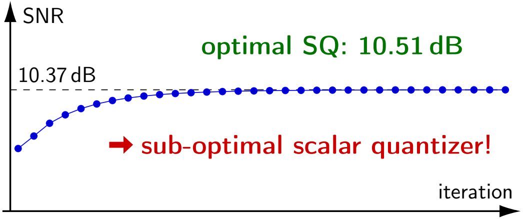


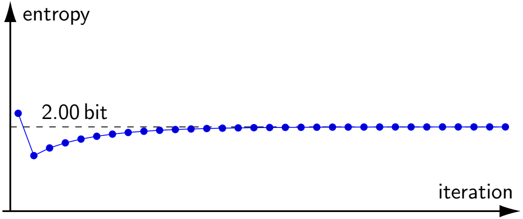




Example: EC Lloyd with Insufficient Initial Intervals (Gaussian Source)

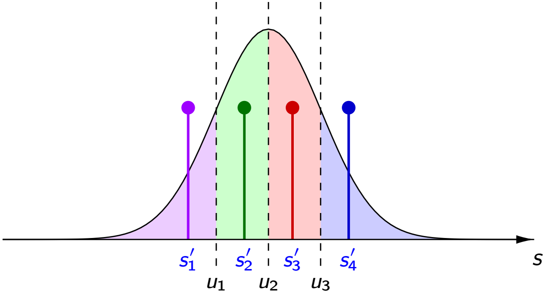






Example: EC Lloyd vs Lloyd at Same Entropy (Gaussian)

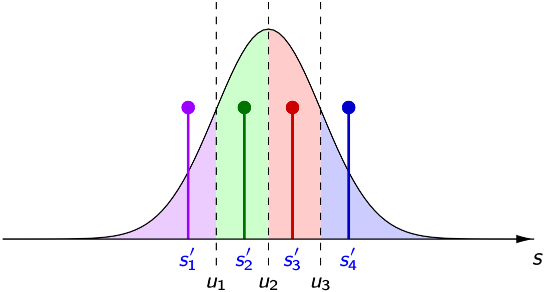
Lloyd Algorithm



$$
\begin{aligned}
K &= 4 \qquad\qquad (R_{\mathrm{FL}} = 2.0)\\
H &= 1.911\\
D &= 0.117\\
\mathrm{SNR} &= 9.30\ \mathrm{dB}
\end{aligned}
$$

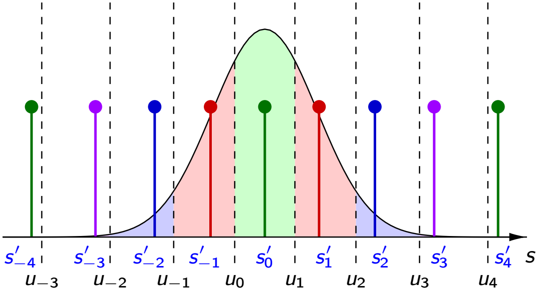
Example: EC Lloyd vs Lloyd at Same Entropy (Gaussian)

Lloyd Algorithm



$$K = 4 \qquad (R_{FL} = 2.0)$$
$$H = 1.911$$
$$D = 0.117$$
$$\text{SNR} = 9.30\text{ dB}$$

Entropy-Constrained Lloyd Algorithm



$$\lambda = 0.1393$$
$$H = 1.911$$
$$D = 0.101 \quad \Rightarrow \quad \text{factor 0.86 smaller}$$
$$\text{SNR} = 9.98\text{ dB} \quad \Rightarrow \quad 0.68\text{ dB better}$$

Example: EC Lloyd Algorithm for Laplacian Source

Laplacian Source

- Zero mean $\mu = 0$ and unit variance $\sigma^2 = 1$

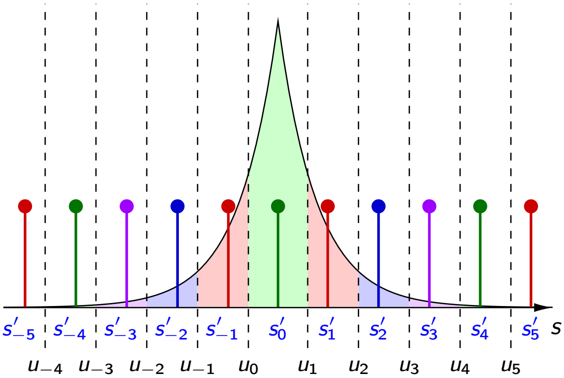
EC Lloyd Quantizer for 2 bits per sample

- Decision thresholds:

$$\begin{aligned} u_{0/1} &= \pm 0.540 \\ u_{-1/2} &= \pm 1.465 \\ u_{-2/3} &= \pm 2.390 \\ u_{-3/4} &= \pm 3.315 \\ u_{-4/5} &= \pm 4.240 \\ &\cdots \end{aligned}$$

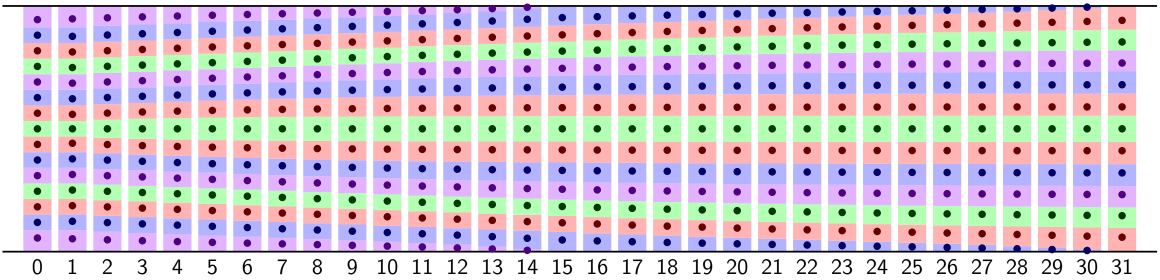
- Reconstruction levels:

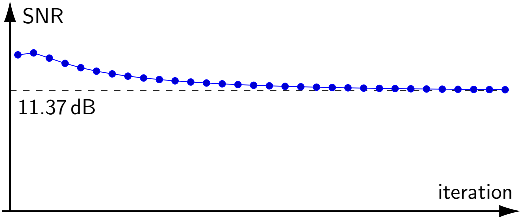
$$\begin{aligned} s'_0 &= 0.000 \\ s'_{\pm 1} &= \pm 0.905 \\ s'_{\pm 2} &= \pm 1.830 \\ s'_{\pm 3} &= \pm 2.755 \\ s'_{\pm 4} &= \pm 3.681 \\ s'_{\pm 5} &= \pm 4.606 \\ &\cdots \end{aligned}$$

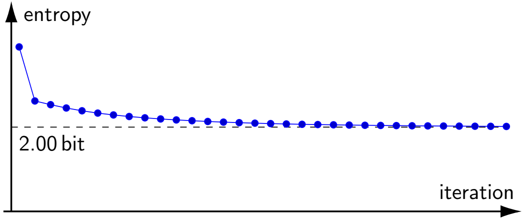


$$\begin{aligned} R &= 2.00 & &(\text{rate} = \text{entropy}) \\ D &= 0.073 \\ \text{SNR} &= 11.37\,\text{dB} & &(\text{SLB} = 12.67\,\text{dB}) \end{aligned}$$

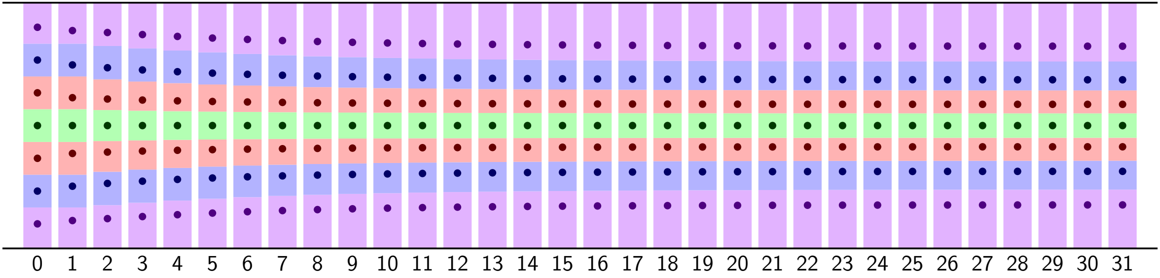
Example: Convergence of EC Lloyd Algorithm for Laplacian Source

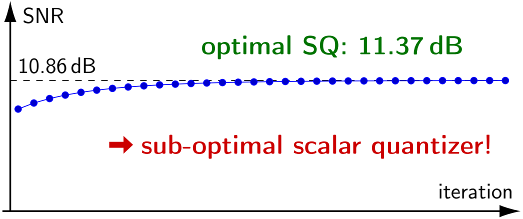


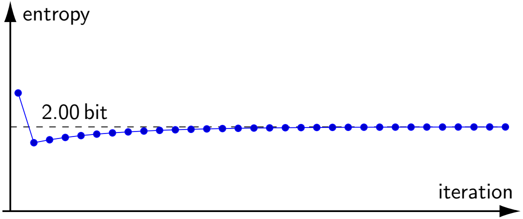




Example: EC Lloyd with Insufficient Initial Intervals (Laplacian Source)

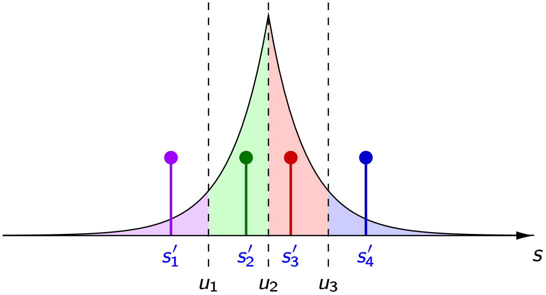






Example: EC Lloyd vs Lloyd at Same Entropy (Laplace)

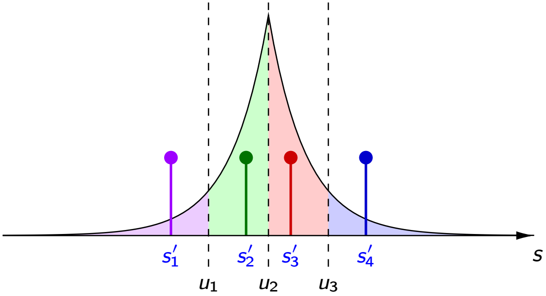
Lloyd Algorithm



$$\begin{aligned} K &= 4 \qquad\qquad (R_{\mathrm{FL}} = 2.0) \\ H &= 1.728 \\ D &= 0.176 \\ \mathrm{SNR} &= 7.54\ \mathrm{dB} \end{aligned}$$

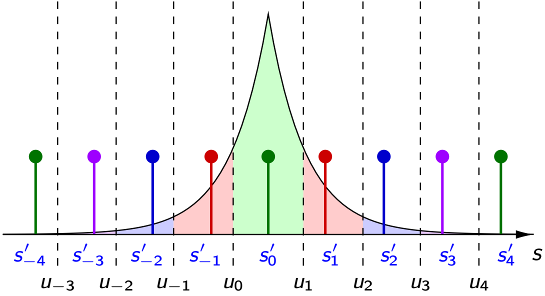
Example: EC Lloyd vs Lloyd at Same Entropy (Laplace)

Lloyd Algorithm



$$
\begin{aligned}
K &= 4 \qquad (R_{\text{FL}} = 2.0)\\
H &= 1.728\\
D &= 0.176\\
\text{SNR} &= 7.54\text{ dB}
\end{aligned}
$$

Entropy-Constrained Lloyd Algorithm



$$
\begin{aligned}
\lambda &= 0.1350\\
H &= 1.728\\
D &= 0.104 \quad \Rightarrow \quad \text{factor 0.59 smaller}\\
\text{SNR} &= 9.83\text{ dB} \quad \Rightarrow \quad \text{2.29 dB better}
\end{aligned}
$$

Review: MSE Distortion for Centroid Quantizers at High Rates

High-Rate Approximation

- High rates: Pdf $f(s)$ is nearly constant inside each quantization interval

$$f(s) \approx \frac{p_k}{\Delta_k} = \frac{p_k}{u_{k+1} - u_k} \qquad \Longrightarrow \qquad p_k \approx f(s'_k) \cdot \Delta_k$$

Review: MSE Distortion for Centroid Quantizers at High Rates

High-Rate Approximation

- High rates: Pdf $f(s)$ is nearly constant inside each quantization interval

$$f(s) \approx \frac{p_k}{\Delta_k} = \frac{p_k}{u_{k+1} - u_k} \qquad \Longrightarrow \qquad p_k \approx f(s'_k) \cdot \Delta_k$$

MSE Distortion for Centroid Quantizers at High Rates

- When considering Lloyd quantizers, we derived

$$\boxed{D = \frac{1}{12} \sum_{\forall k} p_k \, \Delta_k^2}$$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \to 0$)

$$R \;=\; H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k$$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \to 0$)

$$R \;=\; H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k$$

$$\big[\, p_k = f(s'_k)\,\Delta_k \,\big]$$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \to 0$)

$$R = H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k$$

$$\big[\, p_k = f(s'_k)\,\Delta_k \,\big] \qquad = -\sum_{\forall k} p_k \left(\log_2 f(s'_k) + \log_2 \Delta_k \right)$$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \to 0$)

$$
\begin{aligned}
R &= H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k \\
\big[\, p_k = f(s'_k)\,\Delta_k \,\big] \qquad &= -\sum_{\forall k} p_k \left(\log_2 f(s'_k) + \log_2 \Delta_k \right) \\
&= -\sum_{\forall k} f(s'_k)\, \log_2 f(s'_k)\, \Delta_k - \sum_{\forall k} p_k \log_2 \Delta_k
\end{aligned}
$$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \to 0$)

$$R \;=\; H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k$$

$\big[\, p_k = f(s'_k)\,\Delta_k \,\big]$

$$= -\sum_{\forall k} p_k \left(\log_2 f(s'_k) + \log_2 \Delta_k \right)$$

$$= -\sum_{\forall k} f(s'_k)\, \log_2 f(s'_k)\,\Delta_k - \sum_{\forall k} p_k \log_2 \Delta_k$$

$\big[\, \Delta_k \to 0 \,\big]$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \rightarrow 0$)

$$
\begin{aligned}
R &= H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k \\
\left[\, p_k = f(s_k')\,\Delta_k \,\right] \qquad &= -\sum_{\forall k} p_k \left(\log_2 f(s_k') + \log_2 \Delta_k \right) \\
&= -\sum_{\forall k} f(s_k')\, \log_2 f(s_k')\, \Delta_k - \sum_{\forall k} p_k \log_2 \Delta_k \\
\left[\, \Delta_k \rightarrow 0 \,\right] \qquad &= -\int_{-\infty}^{\infty} f(s)\, \log_2 f(s)\, \mathrm{d}s - \frac{1}{2} \sum_{\forall k} p_k \log_2 \Delta_k^2
\end{aligned}
$$

Entropy of Quantization Indexes at High Rates

Average Bit Rate for Optimal Entropy Coding

- Approximation for high bit rates ($\Delta_k \to 0$)

$$
\begin{aligned}
R &= H(S') = -\sum_{\forall k} p_k \cdot \log_2 p_k \\
\big[\, p_k = f(s'_k)\,\Delta_k \,\big] \qquad &= -\sum_{\forall k} p_k \Big(\log_2 f(s'_k) + \log_2 \Delta_k \Big) \\
&= -\sum_{\forall k} f(s'_k)\, \log_2 f(s'_k)\, \Delta_k - \sum_{\forall k} p_k \log_2 \Delta_k \\
\big[\, \Delta_k \to 0 \,\big] \qquad &= -\int_{-\infty}^{\infty} f(s)\, \log_2 f(s)\, \mathrm{d}s - \frac{1}{2} \sum_{\forall k} p_k \log_2 \Delta_k^2
\end{aligned}
$$

$$
\boxed{R = h(S) - \frac{1}{2} \sum_{\forall k} p_k \log_2 \Delta_k^2}
$$

High-Rate Approximation: Optimal Entropy-Constrained Scalar Quantizers

- Will use: **Jensen's inequality** for convex functions $\psi(x)$

$$\sum_k \alpha_k = 1 \qquad \Longrightarrow \qquad \sum_k \alpha_k \, \psi(x_k) \;\geq\; \psi\left(\sum_k \alpha_k \, x_k \right) \qquad \Big[\text{ equality iff } x_k = \text{const} \Big]$$

High-Rate Approximation: Optimal Entropy-Constrained Scalar Quantizers

- Will use: **Jensen's inequality** for convex functions $\psi(x)$

$$\sum_k \alpha_k = 1 \qquad \Longrightarrow \qquad \sum_k \alpha_k \, \psi(x_k) \;\geq\; \psi\left(\sum_k \alpha_k \, x_k \right) \qquad \left[\text{equality iff } x_k = \text{const} \right]$$

High-Rate Approximation for Average Bit Rate

- Apply Jensen's inequality for convex function $\psi(x) = -\log_2(x)$

$$R \;=\; h(S) - \frac{1}{2} \sum_{\forall k} p_k \log_2 \Delta_k^2$$

High-Rate Approximation: Optimal Entropy-Constrained Scalar Quantizers

- Will use: **Jensen's inequality** for convex functions $\psi(x)$

$$\sum_k \alpha_k = 1 \quad \Longrightarrow \quad \sum_k \alpha_k \, \psi(x_k) \;\geq\; \psi\left(\sum_k \alpha_k \, x_k \right) \qquad \Big[\, \text{equality iff } x_k = \text{const} \,\Big]$$

High-Rate Approximation for Average Bit Rate

- Apply Jensen's inequality for convex function $\psi(x) = -\log_2(x)$

$$\begin{aligned} R &= h(S) - \frac{1}{2} \sum_{\forall k} p_k \log_2 \Delta_k^2 \\ &\geq h(S) - \frac{1}{2} \log_2 \left(\sum_{\forall k} p_k \Delta_k^2 \right) \qquad \Big[\, \text{equality iff } \Delta_k = \text{const} \,\Big] \end{aligned}$$

High-Rate Approximation: Optimal Entropy-Constrained Scalar Quantizers

- Will use: **Jensen's inequality** for convex functions $\psi(x)$

$$\sum_k \alpha_k = 1 \quad \Longrightarrow \quad \sum_k \alpha_k \, \psi(x_k) \;\geq\; \psi\left(\sum_k \alpha_k \, x_k \right) \qquad \Big[\text{ equality iff } x_k = \text{const} \Big]$$

High-Rate Approximation for Average Bit Rate

- Apply Jensen's inequality for convex function $\psi(x) = -\log_2(x)$

$$\begin{aligned} R &= h(S) - \frac{1}{2} \sum_{\forall k} p_k \log_2 \Delta_k^2 \\ &\geq h(S) - \frac{1}{2} \log_2 \left(\sum_{\forall k} p_k \Delta_k^2 \right) \qquad \Big[\text{ equality iff } \Delta_k = \text{const} \Big] \\ &= h(S) - \frac{1}{2} \log_2 \big(12\, D \big) \end{aligned}$$

High-Rate Approximation for MSE Distortion: Gish & Pierce Asymptote

➡ **MSE distortion & high rates: Optimal scalar quantizers have uniform step sizes**

High-Rate Approximation for MSE Distortion: Gish & Pierce Asymptote

➡ **MSE distortion & high rates: Optimal scalar quantizers have uniform step sizes**

➡ MSE Distortion at high rates

$$D = \frac{1}{12} \sum_{\forall k} p_k \, \Delta_k^2 = \frac{\Delta^2}{12}$$

High-Rate Approximation for MSE Distortion: Gish & Pierce Asymptote

➡ **MSE distortion & high rates: Optimal scalar quantizers have uniform step sizes**

➡ MSE Distortion at high rates

$$D = \frac{1}{12} \sum_{\forall k} p_k \, \Delta_k^2 = \frac{\Delta^2}{12}$$

➡ **High-rate operational rate-distortion function** (Gish & Pierce)

$$\boxed{R_V(D) = h(S) - \frac{1}{2} \log_2 \left(12\, D \right)}$$

High-Rate Approximation for MSE Distortion: Gish & Pierce Asymptote

➡ **MSE distortion & high rates: Optimal scalar quantizers have uniform step sizes**

➡ MSE Distortion at high rates

$$D = \frac{1}{12} \sum_{\forall k} p_k \, \Delta_k^2 = \frac{\Delta^2}{12}$$

➡ **High-rate operational rate-distortion function** (Gish & Pierce)

$$\boxed{R_V(D) = h(S) - \frac{1}{2} \log_2 \left(12\, D \right)}$$

➡ **High-rate operational distortion-rate function**

$$\boxed{D_V(R) = \frac{1}{12} \, 2^{2h(S)} \, 2^{-2R}}$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd :} \qquad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \qquad \text{and} \qquad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\left(2\pi e\,D\right) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\big(12\,D\big) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\big(2\pi e\,D\big) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

➞ **Distortion increase** (at same rate) relative to Shannon lower bound

$$\frac{D_V(R)}{D_L(R)}$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\left(2\pi e\,D\right) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

➡ **Distortion increase** (at same rate) relative to Shannon lower bound

$$\frac{D_V(R)}{D_L(R)} = \frac{\pi\,e}{6} \approx 1.42$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\left(2\pi e\,D\right) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

➡ **Distortion increase** (at same rate) relative to Shannon lower bound

$$\frac{D_V(R)}{D_L(R)} = \frac{\pi\,e}{6} \approx 1.42$$

➡ **1.53 dB loss in SNR**

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\left(2\pi e\,D\right) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

➡ **Distortion increase** (at same rate) relative to Shannon lower bound

$$\frac{D_V(R)}{D_L(R)} = \frac{\pi\,e}{6} \approx 1.42$$

➡ **1.53 dB loss in SNR**

➡ **Rate increase** (at same distortion) relative to Shannon lower bound

$$R_V(D) - R_L(D)$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\left(2\pi e\,D\right) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

➡ **Distortion increase** (at same rate) relative to Shannon lower bound

$$\frac{D_V(R)}{D_L(R)} = \frac{\pi\,e}{6} \approx 1.42$$

➡ **1.53 dB loss in SNR**

➡ **Rate increase** (at same distortion) relative to Shannon lower bound

$$R_V(D) - R_L(D) = \frac{1}{2}\log_2\left(\frac{\pi\,e}{6}\right) \approx 0.2546$$

Comparison to Shannon Lower Bound

- High-rate approximations for MSE distortion

$$\text{EC Lloyd}: \quad R_V(D) = h(S) - \frac{1}{2}\log_2\left(12\,D\right) \quad \text{and} \quad D_V(R) = \frac{1}{12}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

$$\text{SLB}: \quad R_L(D) = h(S) - \frac{1}{2}\log_2\left(2\pi e\,D\right) \quad \text{and} \quad D_L(R) = \frac{1}{2\pi e}\cdot 2^{2h(S)}\cdot 2^{-2R}$$

➡ **Distortion increase** (at same rate) relative to Shannon lower bound

$$\frac{D_V(R)}{D_L(R)} = \frac{\pi\,e}{6} \approx 1.42$$

➡ **1.53 dB loss in SNR**

➡ **Rate increase** (at same distortion) relative to Shannon lower bound

$$R_V(D) - R_L(D) = \frac{1}{2}\log_2\left(\frac{\pi\,e}{6}\right) \approx 0.2546$$

➡ **roughly 1/4 bit per sample**

Summary: High-Rate Approximations for MSE Distortion

- General form of high-rate approximations for MSE distortion

$$\boxed{D_X(R) \;=\; \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}} \qquad \text{and} \qquad \boxed{R_X(D) \;=\; \frac{1}{2}\log_2\left(\frac{\varepsilon_X^2 \cdot \sigma^2}{D}\right)}$$

Summary: High-Rate Approximations for MSE Distortion

- General form of high-rate approximations for MSE distortion

$$\boxed{D_X(R) \;=\; \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}} \qquad \text{and} \qquad \boxed{R_X(D) \;=\; \frac{1}{2}\log_2\left(\frac{\varepsilon_X^2 \cdot \sigma^2}{D}\right)}$$

	Shannon lower bound	EC Lloyd + VLC	Lloyd + FLC
general :	$\varepsilon_L^2 = \frac{1}{2\pi e}\, 2^{2h(S/\sigma)}$	$\varepsilon_V^2 = \frac{1}{12}\, 2^{2h(S/\sigma)}$	$\varepsilon_F^2 = \frac{1}{12}\int_{-\infty}^{\infty} \sqrt[3]{f(s/\sigma)}\,\mathrm{d}s$

Summary: High-Rate Approximations for MSE Distortion

- General form of high-rate approximations for MSE distortion

$$\boxed{D_X(R) \;=\; \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}} \qquad \text{and} \qquad \boxed{R_X(D) \;=\; \frac{1}{2}\log_2\left(\frac{\varepsilon_X^2 \cdot \sigma^2}{D}\right)}$$

	Shannon lower bound	EC Lloyd + VLC	Lloyd + FLC
general :	$\varepsilon_L^2 = \frac{1}{2\pi e}\, 2^{2h(S/\sigma)}$	$\varepsilon_V^2 = \frac{1}{12}\, 2^{2h(S/\sigma)}$	$\varepsilon_F^2 = \frac{1}{12}\int_{-\infty}^{\infty} \sqrt[3]{f(s/\sigma)}\,\mathrm{d}s$
uniform :	$\varepsilon_L^2 = \frac{6}{\pi e} \approx 0.70$	$\varepsilon_V^2 = 1$	$\varepsilon_F^2 = 1$

Summary: High-Rate Approximations for MSE Distortion

- General form of high-rate approximations for MSE distortion

$$\boxed{D_X(R) \;=\; \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}} \quad \text{and} \quad \boxed{R_X(D) \;=\; \frac{1}{2}\log_2\left(\frac{\varepsilon_X^2 \cdot \sigma^2}{D}\right)}$$

	Shannon lower bound	EC Lloyd + VLC	Lloyd + FLC
general :	$\varepsilon_L^2 = \frac{1}{2\pi e}\,2^{2h(S/\sigma)}$	$\varepsilon_V^2 = \frac{1}{12}\,2^{2h(S/\sigma)}$	$\varepsilon_F^2 = \frac{1}{12}\int_{-\infty}^{\infty}\sqrt[3]{f(s/\sigma)}\,\mathrm{d}s$
uniform :	$\varepsilon_L^2 = \frac{6}{\pi e} \approx 0.70$	$\varepsilon_V^2 = 1$	$\varepsilon_F^2 = 1$
Laplace :	$\varepsilon_L^2 = \frac{e}{\pi} \approx 0.86$	$\varepsilon_V^2 = \frac{e^2}{6} \approx 1.23$	$\varepsilon_F^2 = \frac{9}{2} = 4.5$

Summary: High-Rate Approximations for MSE Distortion

- General form of high-rate approximations for MSE distortion

$$\boxed{D_X(R) \;=\; \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}} \qquad \text{and} \qquad \boxed{R_X(D) \;=\; \frac{1}{2}\log_2\left(\frac{\varepsilon_X^2 \cdot \sigma^2}{D}\right)}$$

	Shannon lower bound	EC Lloyd + VLC	Lloyd + FLC
general :	$\varepsilon_L^2 = \frac{1}{2\pi e}\, 2^{2h(S/\sigma)}$	$\varepsilon_V^2 = \frac{1}{12}\, 2^{2h(S/\sigma)}$	$\varepsilon_F^2 = \frac{1}{12}\int_{-\infty}^{\infty}\sqrt[3]{f(s/\sigma)}\,\mathrm{d}s$
uniform :	$\varepsilon_L^2 = \frac{6}{\pi e} \approx 0.70$	$\varepsilon_V^2 = 1$	$\varepsilon_F^2 = 1$
Laplace :	$\varepsilon_L^2 = \frac{e}{\pi} \approx 0.86$	$\varepsilon_V^2 = \frac{e^2}{6} \approx 1.23$	$\varepsilon_F^2 = \frac{9}{2} = 4.5$
Gauss :	$\varepsilon_L^2 = 1$	$\varepsilon_V^2 = \frac{\pi e}{6} \approx 1.42$	$\varepsilon_F^2 = \frac{\sqrt{3}\,\pi}{2} \approx 2.72$

Comparison of Quantizers and High-Rate Approximations: Gaussian Source

High-rate approximations

$$D_X(R) = \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}$$

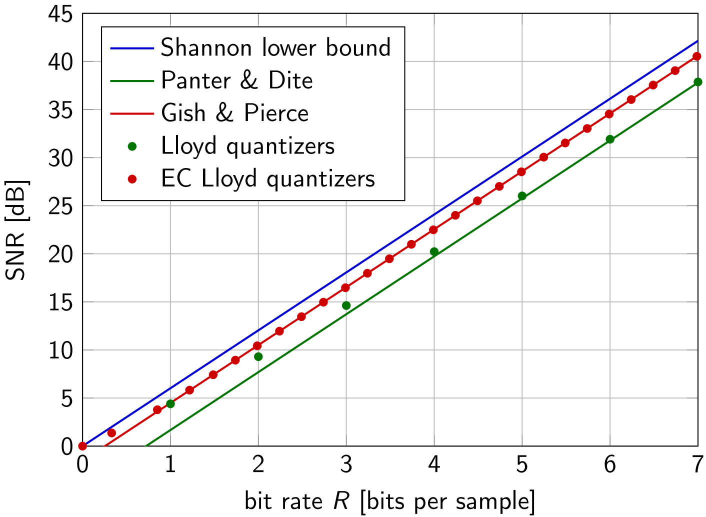
SLB: $\varepsilon_L^2 = 1$

Lloyd: $\varepsilon_F^2 = \frac{\sqrt{3}\,\pi}{2}$

EC-Lloyd: $\varepsilon_V^2 = \frac{\pi\, e}{6}$

$$\frac{D_F}{D_L} = \frac{\sqrt{3}\,\pi}{2} \approx 2.72 \quad (4.34\,\text{dB})$$

$$\frac{D_V}{D_L} = \frac{\pi\, e}{6} \approx 1.42 \quad (1.53\,\text{dB})$$



Comparison of Quantizers and High-Rate Approximations: Laplacian Source

High-rate approximations

$$D_X(R) = \varepsilon_X^2 \cdot \sigma^2 \cdot 2^{-2R}$$

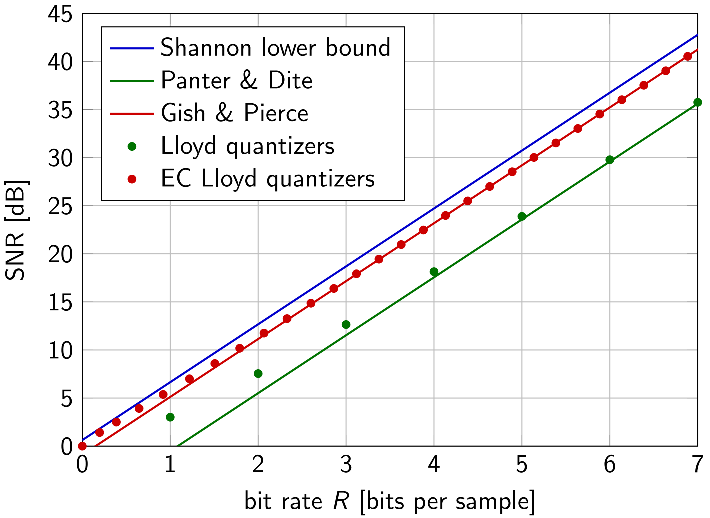
SLB: $\varepsilon_L^2 = \dfrac{e}{\pi}$

Lloyd: $\varepsilon_F^2 = \dfrac{9}{2}$

EC-Lloyd: $\varepsilon_V^2 = \dfrac{e^2}{6}$

$$\frac{D_F}{D_L} = \frac{9\,\pi}{2\,e} \approx 5.20 \quad (7.16\,\text{dB})$$

$$\frac{D_V}{D_L} = \frac{\pi\,e}{6} \approx 1.42 \quad (1.53\,\text{dB})$$



Scalar Quantization in Practice

Quantization in Practice

- Most quantizers used in practice are scalar quantizers
- Examples for usage of scalar quantization (in combination with other techniques):
  - Audio coding: MP3, AAC
  - Image coding: JPEG, JPEG-2000, JPEG-XR
  - Video coding: MPEG-2 Video, H.264/AVC, H.265/HEVC

Scalar Quantization in Practice

Quantization in Practice

- Most quantizers used in practice are scalar quantizers
- Examples for usage of scalar quantization (in combination with other techniques):
  - Audio coding: MP3, AAC
  - Image coding: JPEG, JPEG-2000, JPEG-XR
  - Video coding: MPEG-2 Video, H.264/AVC, H.265/HEVC

Entropy-Constrained Scalar Quantizers ?

- Rarely used in practice

Scalar Quantization in Practice

Quantization in Practice

- Most quantizers used in practice are scalar quantizers
- Examples for usage of scalar quantization (in combination with other techniques):
  - Audio coding: MP3, AAC
  - Image coding: JPEG, JPEG-2000, JPEG-XR
  - Video coding: MPEG-2 Video, H.264/AVC, H.265/HEVC

Entropy-Constrained Scalar Quantizers ?

- Rarely used in practice

➡ Problem: Reconstruction levels depends on source properties
  - In practice, source to be coded has unknown statistical properties
  - Need to transmit reconstruction levels (can change over time)

Scalar Quantization in Practice

Quantization in Practice

- Most quantizers used in practice are scalar quantizers
- Examples for usage of scalar quantization (in combination with other techniques):
  - Audio coding: MP3, AAC
  - Image coding: JPEG, JPEG-2000, JPEG-XR
  - Video coding: MPEG-2 Video, H.264/AVC, H.265/HEVC

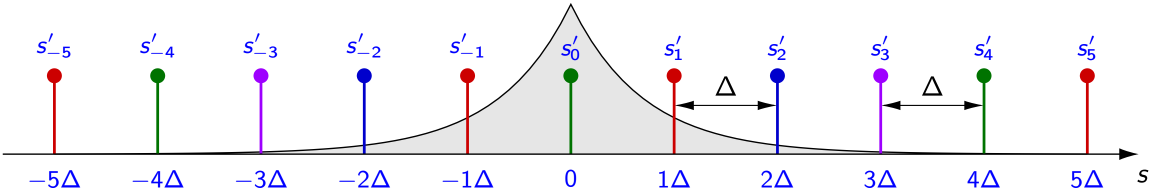
Entropy-Constrained Scalar Quantizers ?

- Rarely used in practice

➡ Problem: Reconstruction levels depends on source properties
  - In practice, source to be coded has unknown statistical properties
  - Need to transmit reconstruction levels (can change over time)

➡ **Need simpler, but still efficient design**

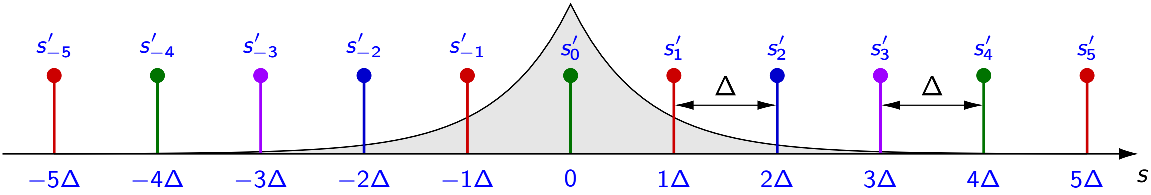
Uniform Reconstruction Quantizers (URQs)



Uniform reconstruction quantizers

- Equally spaced reconstruction levels (indicated by quantization step size Δ)

Uniform Reconstruction Quantizers (URQs)

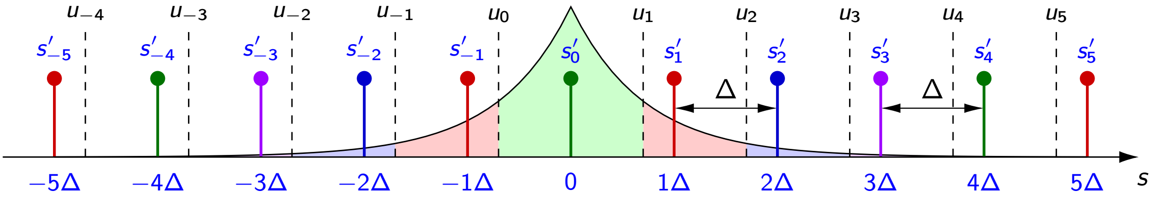


Uniform reconstruction quantizers

- Equally spaced reconstruction levels (indicated by quantization step size Δ)

➡ **Decoder:**
- Reconstruction levels are completely specified by **quantization step size Δ**
- Simple decoding process: $\quad s'_n = \Delta \cdot q_n$

Uniform Reconstruction Quantizers (URQs)



Uniform reconstruction quantizers

- Equally spaced reconstruction levels (indicated by quantization step size Δ)

➡ **Decoder:**
- Reconstruction levels are completely specified by **quantization step size Δ**
- Simple decoding process: $s'_n = \Delta \cdot q_n$

➡ **Encoder:**
- Freedom to adapt decision thresholds to source statistics
- Simple encoding (rounding) or advanced encoding (Lagrange optimization)

Optimum Uniform Reconstruction Quantizer (URQ)

Optimum URQ Design for MSE Distortion

- Minimization of Lagrange cost for given Lagrange multiplier λ

$$J = D + \lambda \cdot R$$

Optimum Uniform Reconstruction Quantizer (URQ)

Optimum URQ Design for MSE Distortion

- Minimization of Lagrange cost for given Lagrange multiplier λ

$$\begin{aligned} J &= D + \lambda \cdot R \\ &= \mathrm{E}\Big\{ \big(S - Q(S)\big)^2 \Big\} + \lambda \cdot \mathrm{E}\big\{ \ell\big(Q(S)\big) \big\} \end{aligned}$$

Optimum Uniform Reconstruction Quantizer (URQ)

Optimum URQ Design for MSE Distortion

- Minimization of Lagrange cost for given Lagrange multiplier λ

$$
\begin{aligned}
J &= D + \lambda \cdot R \\
&= \mathrm{E}\Big\{ \big(S - Q(S)\big)^2 \Big\} + \lambda \cdot \mathrm{E}\big\{ \ell\big(Q(S)\big) \big\} \\
&= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s) \, \mathrm{d}s + \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s \qquad \big[\, s'_k = k \cdot \Delta \,\big]
\end{aligned}
$$

Optimum Uniform Reconstruction Quantizer (URQ)

Optimum URQ Design for MSE Distortion

- Minimization of Lagrange cost for given Lagrange multiplier λ

$$
\begin{aligned}
J &= D + \lambda \cdot R \\
&= \mathrm{E}\Big\{ \big(S - Q(S)\big)^2 \Big\} + \lambda \cdot \mathrm{E}\big\{ \ell\big(Q(S)\big) \big\} \\
&= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s) \, \mathrm{d}s + \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s \qquad \big[\, s'_k = k \cdot \Delta \,\big]
\end{aligned}
$$

➡ Select Lagrange multiplier λ (which determines operation point)

Optimum Uniform Reconstruction Quantizer (URQ)

Optimum URQ Design for MSE Distortion

- Minimization of Lagrange cost for given Lagrange multiplier λ

$$\begin{aligned} J &= D + \lambda \cdot R \\ &= \mathrm{E}\Big\{ \big(S - Q(S)\big)^2 \Big\} + \lambda \cdot \mathrm{E}\big\{ \ell\big(Q(S)\big) \big\} \\ &= \sum_{\forall k} \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s)\, \mathrm{d}s + \lambda \cdot \sum_{\forall k} \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s \qquad \big[\, s'_k = k \cdot \Delta \,\big] \end{aligned}$$

➡ Select Lagrange multiplier λ (which determines operation point)

➡ Minimize J with respect to

- Decision thresholds u_k
- Codeword lengths ℓ_k
- Quantization step size Δ

Optimization Criterions for URQs with MSE Distortion

$$J = D + \lambda \cdot R = \sum_k \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s) \, \mathrm{d}s + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

Optimization Criterions for URQs with MSE Distortion

$$J = D + \lambda \cdot R = \sum_k \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s) \, \mathrm{d}s + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

1. **Optimal decision thresholds** u_k for given Δ and ℓ_k (same as for EC Lloyd)

$$u_k = \Delta \left(k - \frac{1}{2} \right) + \frac{\lambda}{2\Delta} \big(\ell_k - \ell_{k-1} \big) \qquad \Big[\text{note: } s'_k = k\Delta \Big]$$

Optimization Criterions for URQs with MSE Distortion

$$J = D + \lambda \cdot R = \sum_k \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s)\, \mathrm{d}s + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

1. **Optimal decision thresholds** u_k for given Δ and ℓ_k (same as for EC Lloyd)

$$u_k = \Delta \left(k - \frac{1}{2} \right) + \frac{\lambda}{2\Delta} \left(\ell_k - \ell_{k-1} \right) \qquad \left[\text{note: } s'_k = k\Delta \right]$$

2. **Optimal codeword lengths** ℓ_k for given u_k (same as for EC Lloyd)

$$\ell_k = -\log_2 p_k = -\log_2 \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s$$

Optimization Criterions for URQs with MSE Distortion

$$J = D + \lambda \cdot R = \sum_k \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s) \, \mathrm{d}s + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

1 Optimal decision thresholds u_k for given Δ and ℓ_k (same as for EC Lloyd)

$$u_k = \Delta \left(k - \frac{1}{2} \right) + \frac{\lambda}{2\Delta} \left(\ell_k - \ell_{k-1} \right) \qquad \left[\text{note: } s'_k = k\Delta \right]$$

2 Optimal codeword lengths ℓ_k for given u_k (same as for EC Lloyd)

$$\ell_k = -\log_2 p_k = -\log_2 \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

3 Optimum quantization step size Δ for given u_k

$$\frac{\partial}{\partial \Delta} J = \frac{\partial}{\partial \Delta} D = 0$$

Optimization Criterions for URQs with MSE Distortion

$$J = D + \lambda \cdot R = \sum_k \int_{u_k}^{u_{k+1}} (s - k\Delta)^2 \, f(s) \, \mathrm{d}s + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

1. **Optimal decision thresholds** u_k for given Δ and ℓ_k (same as for EC Lloyd)

$$u_k = \Delta \left(k - \frac{1}{2} \right) + \frac{\lambda}{2\Delta} \left(\ell_k - \ell_{k-1} \right) \qquad \left[\text{note: } s'_k = k\Delta \right]$$

2. **Optimal codeword lengths** ℓ_k for given u_k (same as for EC Lloyd)

$$\ell_k = -\log_2 p_k = -\log_2 \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s$$

3. **Optimum quantization step size** Δ for given u_k

$$\frac{\partial}{\partial \Delta} J = \frac{\partial}{\partial \Delta} D = 0 \qquad \Longrightarrow \qquad \Delta = \frac{\sum_k k \int_{u_k}^{u_{k+1}} s \, f(s) \, \mathrm{d}s}{\sum_k k^2 \int_{u_k}^{u_{k+1}} f(s) \, \mathrm{d}s}$$

Iterative URQ Design Algorithm

Given is:

- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative URQ Design Algorithm

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

Iterative URQ Design Algorithm

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial quantization step size Δ and initial codeword lengths $\{\ell_k\}$

Iterative URQ Design Algorithm

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial quantization step size Δ and initial codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \Delta \left(k - \frac{1}{2} \right) + \frac{\lambda}{2\Delta} (\ell_k - \ell_{k-1})$$

Iterative URQ Design Algorithm

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial quantization step size Δ and initial codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \Delta\left(k - \frac{1}{2}\right) + \frac{\lambda}{2\Delta}\left(\ell_k - \ell_{k-1}\right)$$

3. Update the codeword lengths $\{\ell_k\}$ and quantization step size Δ according to

$$\ell_k = -\log_2 p_k, \qquad \Delta = \frac{\sum_k k \int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\sum_k k^2 \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

Iterative URQ Design Algorithm

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial quantization step size Δ and initial codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \Delta\left(k - \frac{1}{2}\right) + \frac{\lambda}{2\Delta}\left(\ell_k - \ell_{k-1}\right)$$

3. Update the codeword lengths $\{\ell_k\}$ and quantization step size Δ according to

$$\ell_k = -\log_2 p_k, \qquad\qquad \Delta = \frac{\sum_k k \int_{u_k}^{u_{k+1}} s\, f(s)\,\mathrm{d}s}{\sum_k k^2 \int_{u_k}^{u_{k+1}} f(s)\,\mathrm{d}s}$$

4. Repeat the previous three steps until convergence

Iterative URQ Design Algorithm

Given is:
- the marginal probability density function $f(s)$ of the source
- a Lagrange multiplier $\lambda > 0$

Iterative quantizer design

1. Choose an initial quantization step size Δ and initial codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

$$u_k = \Delta \left(k - \frac{1}{2} \right) + \frac{\lambda}{2\Delta} \left(\ell_k - \ell_{k-1} \right)$$

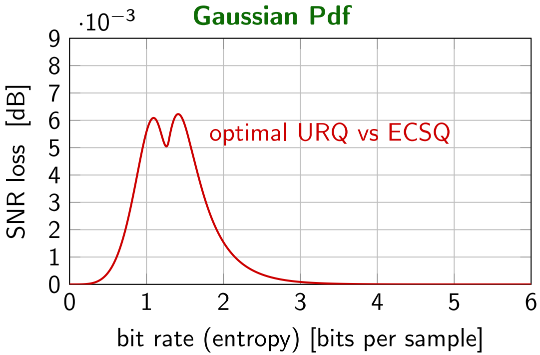
3. Update the codeword lengths $\{\ell_k\}$ and quantization step size Δ according to

$$\ell_k = -\log_2 p_k, \qquad \Delta = \frac{\sum_k k \int_{u_k}^{u_{k+1}} s\, f(s)\, \mathrm{d}s}{\sum_k k^2 \int_{u_k}^{u_{k+1}} f(s)\, \mathrm{d}s}$$

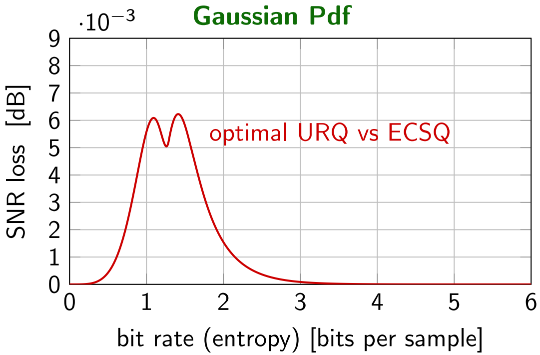
4. Repeat the previous three steps until convergence

Note: Similar iterative algorithm for training set (instead of pdf)

Coding Efficiency Comparison: Optimal URQs vs ECSQs

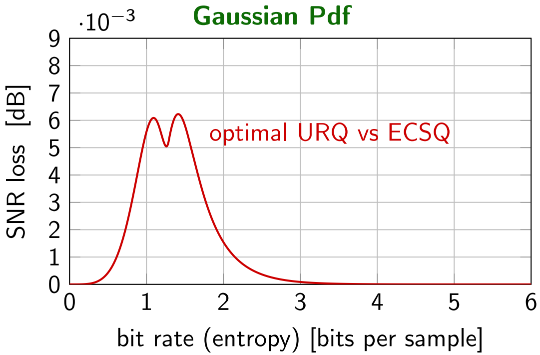


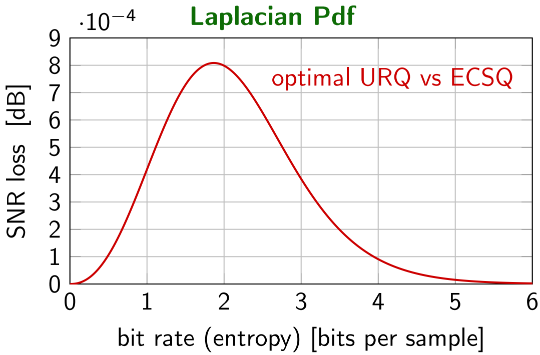
Coding Efficiency Comparison: Optimal URQs vs ECSQs



$$\Delta\text{SNR} < 0.0063 \quad \Rightarrow \quad \frac{D_{\text{URQ}}}{D_{\text{opt}}} < 1.0015$$

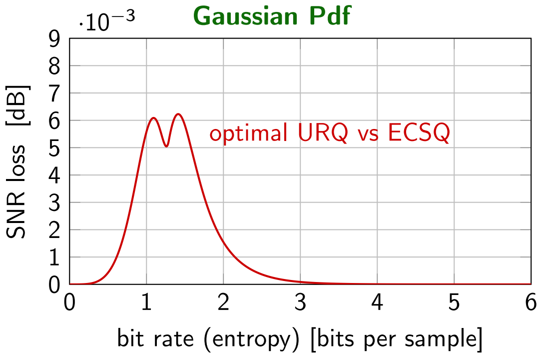
Coding Efficiency Comparison: Optimal URQs vs ECSQs

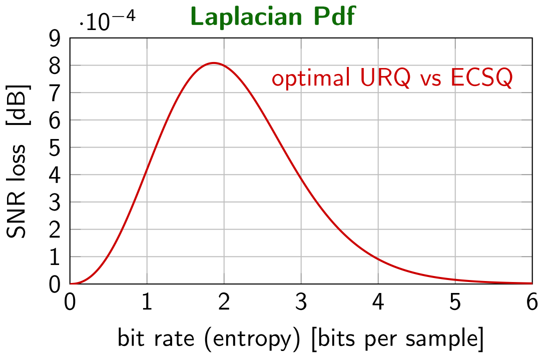




$$\Delta\text{SNR} < 0.0063 \quad \Rightarrow \quad \frac{D_{\text{URQ}}}{D_{\text{opt}}} < 1.0015$$

Coding Efficiency Comparison: Optimal URQs vs ECSQs

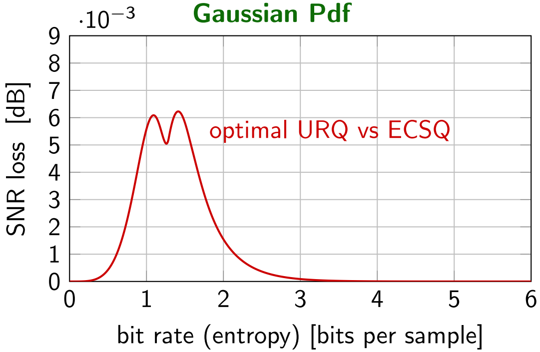


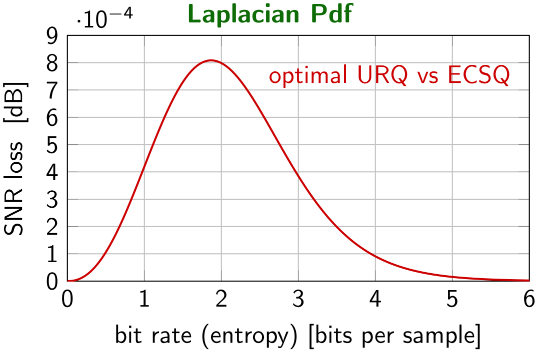


$$\Delta\mathrm{SNR} < 0.0063 \quad \rightarrow \quad \frac{D_{\mathrm{URQ}}}{D_{\mathrm{opt}}} < 1.0015$$

$$\Delta\mathrm{SNR} < 0.00081 \quad \rightarrow \quad \frac{D_{\mathrm{URQ}}}{D_{\mathrm{opt}}} < 1.0002$$

Coding Efficiency Comparison: Optimal URQs vs ECSQs



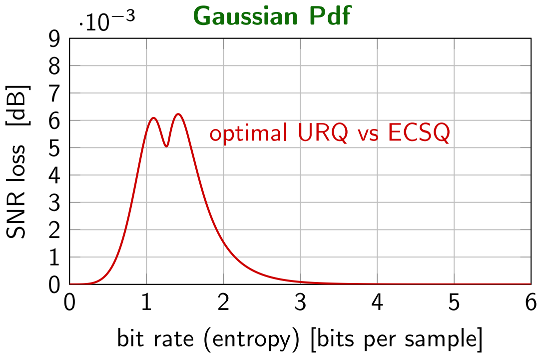


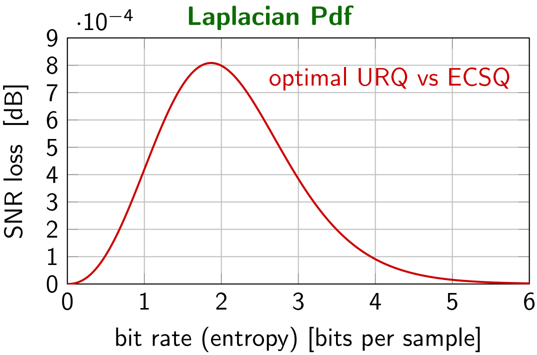
$\Delta\text{SNR} < 0.0063$ ➡ $\dfrac{D_{\text{URQ}}}{D_{\text{opt}}} < 1.0015$

$\Delta\text{SNR} < 0.00081$ ➡ $\dfrac{D_{\text{URQ}}}{D_{\text{opt}}} < 1.0002$

➡ **For typical pdfs: Negligible loss versus optimal ECSQ**

Coding Efficiency Comparison: Optimal URQs vs ECSQs





$\Delta\text{SNR} < 0.0063 \;\Rightarrow\; \frac{D_{\text{URQ}}}{D_{\text{opt}}} < 1.0015$ $\qquad$ $\Delta\text{SNR} < 0.00081 \;\Rightarrow\; \frac{D_{\text{URQ}}}{D_{\text{opt}}} < 1.0002$

➡ **For typical pdfs: Negligible loss versus optimal ECSQ**

➡ **Same high-rate performance as optimal ECSQ**

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad\qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0 \qquad \Rightarrow \qquad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} \, D(R)$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad\qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0 \qquad \Rightarrow \qquad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} D(R)$$

➡ Lagrange multiplier at high rates

$$\lambda \;=\; -\frac{\mathrm{d}}{\mathrm{d}R} D(R)$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0 \qquad \Rightarrow \qquad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} \, D(R)$$

➡ Lagrange multiplier at high rates

$$\lambda \;=\; -\frac{\mathrm{d}}{\mathrm{d}R} \, D(R) = 2 \cdot \ln 2 \cdot \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad\qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0 \qquad \Rightarrow \qquad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} \, D(R)$$

➡ Lagrange multiplier at high rates

$$\lambda \; = \; -\frac{\mathrm{d}}{\mathrm{d}R} \, D(R) = 2 \cdot \ln 2 \cdot \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R} = 2 \cdot \ln 2 \cdot D$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0 \qquad \Rightarrow \qquad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} D(R)$$

➡ Lagrange multiplier at high rates

$$\lambda \; = \; -\frac{\mathrm{d}}{\mathrm{d}R} D(R) = 2 \cdot \ln 2 \cdot \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R} = 2 \cdot \ln 2 \cdot D = \frac{\ln 2}{6} \cdot \Delta^2$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda \, R \big) = 0 \qquad \Rightarrow \qquad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} D(R)$$

➡ Lagrange multiplier at high rates

$$\lambda \; = \; -\frac{\mathrm{d}}{\mathrm{d}R} D(R) = 2 \cdot \ln 2 \cdot \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R} = 2 \cdot \ln 2 \cdot D = \frac{\ln 2}{6} \cdot \Delta^2$$

➡ **Often used relationship between λ and Δ**

$$\lambda = \frac{\ln 2}{6} \cdot \Delta^2$$

Quantization Step Size vs Lagrange Multiplier

- High-rate distortion approximations

$$D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \qquad D = \frac{1}{12} \sum_k p_k \, \Delta_k^2 = \frac{\Delta^2}{12} \qquad \text{(also valid for URQ)}$$

- Lagrangian optimization

$$\frac{\mathrm{d}}{\mathrm{d}R} \big(D(R) + \lambda\, R \big) = 0 \quad \Rightarrow \quad \lambda = -\frac{\mathrm{d}}{\mathrm{d}R} D(R)$$

➡ Lagrange multiplier at high rates

$$\lambda = -\frac{\mathrm{d}}{\mathrm{d}R} D(R) = 2 \cdot \ln 2 \cdot \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R} = 2 \cdot \ln 2 \cdot D = \frac{\ln 2}{6} \cdot \Delta^2$$

➡ **Often used relationship between λ and Δ**

$$\lambda = \frac{\ln 2}{6} \cdot \Delta^2 \qquad \text{or, more generally,} \qquad \lambda = \mathrm{const} \cdot \Delta^2$$

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k
- Reconstruction at decoder: $s' = k \cdot \Delta$

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k
- Reconstruction at decoder: $s' = k \cdot \Delta$

Encoding Process: Determine optimal quantization indexes

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k
- Reconstruction at decoder: $s' = k \cdot \Delta$

Encoding Process: Determine optimal quantization indexes

- Set Lagrange multiplier according to $\lambda = \mathrm{const} \cdot \Delta^2$

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k
- Reconstruction at decoder: $s' = k \cdot \Delta$

Encoding Process: Determine optimal quantization indexes

- Set Lagrange multiplier according to $\lambda = \mathrm{const} \cdot \Delta^2$
- Codeword length $\{\ell_k\}$ are given by
  - Codeword table (specified in standard) or
  - Probabilities used in arithmetic coding ($\ell_k = -\log_2 p_k$)

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k
- Reconstruction at decoder: $s' = k \cdot \Delta$

Encoding Process: Determine optimal quantization indexes

- Set Lagrange multiplier according to $\lambda = \mathrm{const} \cdot \Delta^2$
- Codeword length $\{\ell_k\}$ are given by
  - Codeword table (specified in standard) or
  - Probabilities used in arithmetic coding ($\ell_k = -\log_2 p_k$)
- For each sample s: Choose quantization index k that minimizes

$$J(k) = (s - k\Delta)^2 + \lambda \cdot \ell_k$$

URQs used in Practice

Bitstream Syntax and Decoding Process

- Select quantization step size Δ at encoder: Trade-off quality and bit rate
- Transmit quantization step size Δ and quantization indexes k
- Reconstruction at decoder: $s' = k \cdot \Delta$

Encoding Process: Determine optimal quantization indexes

- Set Lagrange multiplier according to $\lambda = \mathrm{const} \cdot \Delta^2$
- Codeword length $\{\ell_k\}$ are given by
  - Codeword table (specified in standard) or
  - Probabilities used in arithmetic coding ($\ell_k = -\log_2 p_k$)
- For each sample s: Choose quantization index k that minimizes

$$J(k) = (s - k\Delta)^2 + \lambda \cdot \ell_k$$

➡ Note: We only need to check the two neighboring reconstruction levels

$$k_1 = \lfloor s/\Delta \rfloor \qquad \text{and} \qquad k_2 = \lceil s/\Delta \rceil$$

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

➡ Leave all optimizations to encoder (may or may not be exploited)

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

- Codeword lengths ℓ_k given by probabilities $\ell_k = -\log_2 p_k$

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

- Codeword lengths ℓ_k given by probabilities $\ell_k = -\log_2 p_k$
- Optimal encoder decision: Choose quantization index k that minimizes Lagrangian cost $J(k)$

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers
- ➡ Transmit single parameter Δ for specifying operating point
- ➡ Very simple decoding process: $s' = k\Delta$
- ➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

- Codeword lengths ℓ_k given by probabilities $\ell_k = -\log_2 p_k$
- Optimal encoder decision: Choose quantization index k that minimizes Lagrangian cost $J(k)$
- Quantizer (thresholds) and entropy coding adapt to source statistics

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers
- ➡ Transmit single parameter Δ for specifying operating point
- ➡ Very simple decoding process: $s' = k\Delta$
- ➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

- Codeword lengths ℓ_k given by probabilities $\ell_k = -\log_2 p_k$
- Optimal encoder decision: Choose quantization index k that minimizes Lagrangian cost $J(k)$
- Quantizer (thresholds) and entropy coding adapt to source statistics
- ➡ Suitable for unknown and/or instationary sources

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

- Codeword lengths ℓ_k given by probabilities $\ell_k = -\log_2 p_k$
- Optimal encoder decision: Choose quantization index k that minimizes Lagrangian cost $J(k)$
- Quantizer (thresholds) and entropy coding adapt to source statistics

➡ Suitable for unknown and/or instationary sources

➡ Straightforward to exploit conditional probabilities

Advantages of Uniform Reconstruction Quantizers

URQ vs Optimal Scalar Quantizers (ECSQs)

- Performance of optimal URQs is very close to that of optimal scalar quantizers

➡ Transmit single parameter Δ for specifying operating point

➡ Very simple decoding process: $s' = k\Delta$

➡ Leave all optimizations to encoder (may or may not be exploited)

Useful Design: URQ + Adaptive Arithmetic Coding

- Codeword lengths ℓ_k given by probabilities $\ell_k = -\log_2 p_k$
- Optimal encoder decision: Choose quantization index k that minimizes Lagrangian cost $J(k)$
- Quantizer (thresholds) and entropy coding adapt to source statistics

➡ Suitable for unknown and/or instationary sources

➡ Straightforward to exploit conditional probabilities

➡ Most quantizers used in practice are URQs

Summary of Lecture

Optimal Scalar Quantizers

- Minimizes Lagrangian cost $J = D + \lambda\, R$ (where $\lambda > 0$ determines operation point)
- Three optimization criterions: ➡ centroid condition
  ➡ entropy condition
  ➡ modified nearest neighbour condition
- May need large number ($K \to \infty$) of intervals for obtaining optimal quantizer
- High-rate approximation: ➡ distortion is factor 1.42 higher than SLB (1.53 dB)
  ➡ bit rate is roughly 0.25 bits per sample larger than SLB

Uniform Reconstruction Quantizers

- Uniformly spaced reconstruction levels (specified by quantization step size Δ)
- Very simple decoder mapping: $s' = \Delta \cdot k$
- Coding efficiency very close to optimal scalar quantizers (with suitable encoder decisions)

➡ **Most often used quantizer in practice**

Exercise 1: Implement the Entropy-Constrained Lloyd Algorithm (optional)

Implement the entropy-constrained Lloyd algorithm using a programming language of your choice.

- Test the algorithm for
  - a unit-variance Gaussian pdf:
$$f(s) = \frac{1}{\sqrt{2\pi}}\, e^{-\frac{1}{2}\, s^2}$$
  - a unit-variance Laplacian pdf:
$$f(s) = \frac{1}{\sqrt{2}}\, e^{-\sqrt{2}\,|s|}$$
- Use the following Lagrange multipliers: $\lambda = 0.5, 0.2, 0.1, 0.05, 0.02, 0.01$.
- Determine the rate (entropy) R and the distortion D for your quantizers.
- Compare the R-D performance of your quantizers to the high-rate approximation.

You can implement the EC Lloyd algorithm that directly uses the pdf or the EC Lloyd algorithm that uses a training set (files with 1 000 000 samples in float32 format are provided on the course web site)

Exercise 2: Quantization of Sources with Memory

Consider a discrete Markov process $\mathbf{X} = \{X_n\}$ with the symbol alphabet $\mathcal{A}_X = \{0, 2, 4, 6\}$ and the conditional pmf

$$p_{X_n|X_{n-1}}(x_n|x_{n-1}) = \begin{cases} a & : \quad x_n = x_{n-1} \\ \frac{1}{3}(1-a) & : \quad x_n \neq x_{n-1} \end{cases}$$

The parameter a, with $0 < a < 1$, is a variable that specifies the probability that the current symbol is equal to the previous symbol. For $a = 1/4$, our source $\mathbf{X}$ would be i.i.d.

Given is a two-interval quantizer with the reconstruction levels $s'_0 = 1$ and $s'_1 = 5$ and the decision threshold $u_1 = 3$.

(a) Assume optimal entropy coding using the marginal probabilities of the quantization indices and determine the rate-distortion point of the quantizer.

(b) Can the overall quantizer performance be improved by applying conditional entropy coding (e.g., using arithmetic coding with conditional probabilities)?

How does it depend on the parameter a?

Exercise 3: High-Rate Quantization

Consider scalar quantization of a Laplacian source at high rates:

$$f(x) = \frac{\lambda}{2} \cdot e^{-\lambda\,|x|} \qquad \text{with} \qquad \sigma_S^2 = \frac{2}{\lambda^2}$$

In a given system, the used quantizer is a Lloyd quantizer with fixed-length entropy coding (the number of quantization intervals represents a power of 2).

How many bits per sample (for the same MSE distortion) can be saved if the quantizer is replaced by an entropy-constrained quantizer with optimal entropy coding?

Note:

Assume that the operation points of the quantizers can be accurately described by the corresponding high rate approximations.

Exercise 4: Implementation of First Lossy Image Codec

- Use the PPM format as raw data format (see earlier exercise on lossless image coding)
- Use any of the lossless image codecs available in the KVV (or your own implementation) as basis

Implement an Image Encoder

- Quantize the original image samples $s[x, y]$ using a fixed quantization step size Δ
  - ➡ Simple rounding is sufficient for our purpose: $k[x, y] = \mathrm{round}(\, s[x, y]/\Delta \,)$
  - ➡ Transmit the quantization step size Δ at the beginning of the bitstream
- Use the lossless codec for coding the quantization indexes $k[x, y]$

Implement the corresponding Image Decoder

- Decode the quantization indexes $k[x, y]$ using the lossless codec
- Reconstruct the image samples according to: $s'[x, y] = k[x, y] \cdot \Delta$

Test your Codec

- Code selected test images with different quantization step sizes (e.g., $\Delta = 2, 4, 8, 16, 32, 64$)
- Measure the compression factors (based on the file sizes) and judge the image quality by visual inspection

Exercise 5: Quantization of Exponential Source (optional / more difficult)

Consider uniform threshold quantization of an exponential pdf given by $f(x) = a\,e^{-ax}$.

With Δ denoting the quantization step size, the thresholds are given by $u_k = k\Delta$, with $k = 0, 1, 2, \cdots$.

(a) Determine the pmf for the quantization indexes.

Calculate the rate (entropy) as function of the probability $p = P(X > \Delta) = e^{-a\Delta}$.

Describe an entropy coding scheme for the quantization indices that virtually achieves the entropy.

(b) Derive a formula for the optimal reconstruction levels s'_k, for MSE distortion, as function of the quantization step size Δ, the lower interval boundaries u_k, and the probability $p = e^{-a\Delta}$.

(c) Is the obtained quantizer an optimal entropy-constrained scalar quantizer?

(d) Determine the distortion in dependence of the quantization step size for the developed quantizer.

Hint: For $|a| < 1$, $\displaystyle\sum_{k=0}^{\infty} a^k = \frac{1}{1-a}$, $\displaystyle\sum_{k=0}^{\infty} k\,a^k = \frac{a}{(1-a)^2}$, $\displaystyle\sum_{k=0}^{\infty} k^2\,a^k = \frac{a\,(1+a)}{(1-a)^3}$