Quantization III
Uniform reconstruction quantizers

- Equally spaced reconstruction levels (indicated by step size $\Delta$)
- From decoder perspective: Parameterized by single parameter $\Delta$
- Simple decoder mapping $\beta: k \mapsto s'$

$$s' = \beta(k) = \Delta \cdot k$$

- Encoder has freedom to adapt decision thresholds to source
- Can use iterative design algorithm similar to that for ECSQs
Optimum Uniform Reconstruction Quantizer (URQ)

Optimum URQ Design for MSE

- Consider minimization of Lagrange function for given Lagrange multiplier $\lambda$

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

$$= E\left\{ (S - Q(S))^2 \right\} + \lambda \cdot E\{ \ell(Q(S)) \}$$

$$= \int_{\mathbb{R}} (s - Q(S))^2 f(s) \, ds + \lambda \cdot \int_{\mathbb{R}} \ell(Q(s)) f(s) \, ds$$

$$= \sum_k \int_{u_k}^{u_k+1} (s - k\Delta)^2 f(s) \, ds + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_k+1} f(s) \, ds \quad \left[ s'_k = k\Delta \right]$$

- Select Lagrange multiplier $\lambda$ (which determines operation point)

- Minimize $J$ with respect to
  - Decision thresholds $u_k$ (for given $\Delta$ and $\ell_k$)
  - Codeword length $\ell_k$ (for given $\Delta$ and $u_k$)
  - Quantization step size $\Delta$ (for given $u_k$ and $\ell_k$)
Optimization Criterions for URQs with MSE Distortion

\[ J = D + \lambda \cdot R = \sum_k \int_{u_k}^{u_k+1} (s - k\Delta)^2 f(s) \, ds + \lambda \cdot \sum_k \ell_k \int_{u_k}^{u_k+1} f(s) \, ds \]

1. **Optimal decision thresholds** \( u_k \) for given \( \Delta \) and \( \ell_k \) (same as for EC-Lloyd)

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

2. **Optimal codeword length** \( \ell_k \) for given \( u_k \) (same as for EC-Lloyd)

   \[ \ell_k = -\log_2 \rho_k = -\log_2 \left( \int_{u_k}^{u_k+1} f(s) \, ds \right) \]

3. **Optimum quantization step size** \( \Delta \) for given \( u_k \)

   \[ \frac{\partial}{\partial \Delta} J = \frac{\partial}{\partial \Delta} D = 0 \quad \Rightarrow \quad \Delta = \frac{\sum_k k \int_{u_k}^{u_k+1} s f(s) \, ds}{\sum_k k^2 \int_{u_k}^{u_k+1} f(s) \, ds} \]
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 $\Delta$ and initial codeword lengths $\{\ell_k\}$
2. Update the decision thresholds $\{u_k\}$ according to

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

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

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

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

- Laplacian Pdf
  - maximum $\Delta_{SNR} \approx 0.0008$
  - maximum $\frac{D_{URQ}}{D_{ECSQ}} \approx 1.0002$

- Gaussian Pdf
  - maximum $\Delta_{SNR} \approx 0.0062$
  - maximum $\frac{D_{URQ}}{D_{ECSQ}} \approx 1.0014$

➤ 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

\[ D(R) = \varepsilon^2 \cdot \sigma^2 \cdot 2^{-2R}, \quad D = \frac{1}{12} \sum_k p_k \Delta_k^2 = \frac{\Delta^2}{12} \]

- Lagrangian optimization

\[ \frac{d}{dR} (D(R) + \lambda R) = 0 \quad \implies \quad \lambda = -\frac{d}{dR} D(R) \]

- Lagrange multiplier at high rates

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

- Often used relationship between \( \lambda \) and \( \Delta \)

\[ \lambda = \text{const} \cdot \Delta^2 \]

(with experimentally determined proportionality factor)
**URQs used in Practice**

**Bitstream Syntax and Decoding Process**
- Select quantization step size $\Delta$ at encoder: Trade-off quality and bit rate
- Transmit quantization step size $\Delta$ and quantization indexes $k$
- Reconstruction at decoder:
  \[ s' = k \cdot \Delta \]

**Encoding Process: Determine optimal quantization indexes**
- Set Lagrange multiplier according to $\lambda = \text{const} \cdot \Delta^2$
- 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
  \[ J(k) = (s - k\Delta)^2 + \lambda \cdot \ell_k \]
  
  \text{Note: We only need to check the two neighboring reconstruction levels}

\[ k_1 = \lfloor s/\Delta \rfloor \quad \text{and} \quad k_1 = \lceil s/\Delta \rceil \]
Advantages of Uniform Reconstruction Quantizers

**URQ vs Optimal Scalar Quantizers (ECSQs)**

- Performance of URQs is very close to that of optimal scalar quantizers
- Transmit single parameter $\Delta$ 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 $\ell_k$ given by probabilities $\ell_k = -\log_2 p_k$
- Optimal encoder decision: Choose quantization index $k$ that minimizes
  \[ J(k) = D(k) + \lambda \ell_k = (s - k\Delta)^2 - \lambda \cdot \log_2 p_k \]
- 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 (or similar designs)
Can We Further Improve Quantization?

- Scalar quantization: Special case of vector quantization (with $N = 1$)

  ![Graph depicting scalar quantization]

- Vector quantization with $N > 1$ allows a number of new options

  ![Graph depicting vector quantization]

  - pdf
  - Representative vector
  - Amplitude 1
  - Amplitude 2
  - Cell
Vector Quantization

- Generalization of scalar quantization: Consider vectors of $N > 1$ samples
- Mapping of $N$-dimensional vectors of input samples $s$ to countable set of $N$-dimensional reconstruction vectors $s'$

$$Q : \mathbb{R}^N \mapsto \{s'_0, s'_1, s'_2, \ldots\}$$ (1)

- Partition $N$-dimensional space $\mathbb{R}^N$ into quantization cells $\{C_k\}$

$$C_k = \{s \in \mathbb{R}^N : Q(s) = s'_k\}$$ (2)

$$\bigcup_{k} C_k = \mathbb{R}^N \quad \text{with} \quad \forall k \neq j : C_k \cap C_j = \emptyset$$ (3)

- All samples vectors $s$ that fall inside a quantization cell $C_k$ are represented by a reconstruction vector $s'_k$

$$\alpha(s) = k, \quad \forall s \in C_k \quad \text{and} \quad \beta(k) = s'_k$$ (4)

$$Q(s) = s'_k, \quad \forall s \in C_k$$ (5)
Vector Quantizer Performance: Distortion

**Average distortion** (per sample)

\[
D = \mathbb{E}\{d_N(S, Q(S))\} = \int_{\mathbb{R}^N} d_N(s, Q(s)) f_S(s) \, ds \quad (6)
\]

\[
= \sum_{\forall k} \int_{C_k} d_N(s, s'_k) f_S(s) \, ds \quad (7)
\]

- **MSE distortion for \( N \)-dimensional vectors**

\[
d_N(s, s') = \frac{1}{N} \sum_{i=0}^{N-1} (s_i - s'_i)^2 = \frac{1}{N} (s - s')^T (s - s') \quad (8)
\]

**Average MSE distortion** (per sample)

\[
D = \frac{1}{N} \sum_{\forall k} \int_{C_k} (s - s'_k)^T (s - s'_k) f_S(s) \, ds \quad (9)
\]
Vector Quantizer Performance: Rate

- **Average rate** (per sample)

\[
R = \frac{1}{N} \mathbb{E}\{ \ell(S') \} = \frac{1}{N} \mathbb{E}\{ \ell(Q(S)) \} 
\]

\[
= \frac{1}{N} \int_{\mathbb{R}^N} \ell(Q(s)) f(s) \, ds \quad (11)
\]

\[
= \frac{1}{N} \sum_k \ell(s'_k) \int_{C_k} f(s) \, ds \quad (12)
\]

\[
= \frac{1}{N} \sum_k \ell_k p_k \quad (13)
\]

- **Approximate average rate by entropy**

\[
R = -\frac{1}{N} \sum_{\forall k} p_k \log_2 p_k \quad \text{with} \quad p_k = \int_{C_k} f(s) \, ds 
\]
Vector Quantizer with Fixed-Length Codes

Neglect entropy coding (similar to Lloyd quantizer)
- Minimize distortion $D$ for given number $K$ of quantization cells
- Rate can be represented by

$$R = \frac{1}{N} \log_2 K \tag{15}$$

Necessary conditions for optimality (MSE distortion)
- Centroid condition (for representative vectors $s'_k$)

$$s'_k = \mathbb{E}\{ S \mid S \in C_k \} = \frac{1}{p_k} \int_{C_k} s f(s) \, ds \tag{16}$$

- Nearest neighbor condition (for quantization cells $C_k$)

$$\alpha(s) = \arg \min_{\forall k} d_N(s, s'_k) \tag{17}$$

→ Quantizer design: Similar to Lloyd algorithm
Linde-Buzo-Gray (LBG) Algorithm for a Training Set

Given is
- a sufficiently large realization \( \{s_n\} \) of considered source
- the number \( K \) of reconstruction vectors \( \{s'_k\} \)

Iterative quantizer design (extension of Lloyd algorithm)

1. Choose an initial set of \( K \) reconstruction vectors \( \{s'_k\} \)
2. Associate all vectors of the training set \( \{s_n\} \) with one of the quantization cells \( C_k \) according to
   \[
   \alpha(s_n) = \arg \min_{\forall k} d(s_n, s'_k) \quad \text{(nearest neighbor condition)}
   \]
3. Update the reconstruction vectors \( \{s'_k\} \) according to
   \[
   s'_k = \arg \min_{s' \in \mathbb{R}^N} \mathbb{E}\left\{ d_N(S, s') \mid \alpha(S) = k \right\} \quad \text{(centroid condition)}
   \]
4. Repeat the previous two steps until convergence
Example: LBG Algorithm Result for Gaussian IID Source

Result for dimension $N = 2$ and size $K = 16$ (rate of $R = 2$ bit/sample)

- After iteration 8:
  Same performance as in scalar case (SNR of 9.3 dB)

- After iteration 49:
  Improvement to 9.67 dB
Example: LBG Algorithm Result for Gaussian IID Source

Result for dimension $N = 2$ and size $K = 256$ (rate of $R = 4$ bit/sample)

- Gain around 0.9 dB for two-dimensional VQ compared to SQ with fixed-length codes resulting in 20.64 dB (of conjectured 21.05 dB)
Example: LBG Algorithm Result for Laplacian IID Source

Result for dimension $N = 2$ and size $K = 16$ (rate of $R = 2$ bit/sample)

- Large gain (1.32 dB) for two-dimensional VQ compared to SQ with fixed-length codes resulting in 8.87 dB
Example: LBG Algorithm Result for Laplacian IID Source

Result for dimension $N = 2$ and size $K = 256$ (rate of $R = 4$ bit/sample)

- Large gain (1.84 dB) for two-dimensional VQ compared to SQ with fixed-length codes resulting in 19.4 dB (of conjectured 19.99 dB)
The Vector Quantizer Advantage

Gain over scalar quantization can be assigned to 3 effects:

- **Space filling advantage:**
  - $\mathbb{Z}^N$ lattice is not most efficient sphere packing in $N$ dimensions ($N > 1$)
  - Independent from source distribution or statistical dependencies
  - Maximum gain for $N \to \infty$: 1.53 dB

- **Shape advantage:**
  - Exploit shape of source pdf
  - Can also be exploited using entropy-constrained scalar quantization

- **Memory advantage:**
  - Exploit statistical dependencies of the source
  - Can also be exploited using predictive coding, transform coding, block entropy coding or conditional entropy coding
Space Filling Advantage: Example

Consider uniform iid source

- SQ with $R = \log_2 10 = 3.32$ bits/sample: $D = 19.98$ dB
- LBG algorithm converged towards 20.08 dB showing an approximate hexagonal lattice in 2D
Space Filling Advantage: Sphere Packing Density

Center density
- Consider $N$-dimensional spheres with radius $r = 1$
- Measure for packing density: **Center density**

\[ \delta = \frac{\text{average number of sphere centers}}{\text{unit volume}} \]

- Example: $N = 1$ (SQ with intervals of size $2r = 2$)

\[ \delta = \frac{1}{2} \]

Roger’s bound
- Theoretical upper bound for center density (last term being approximate)

\[ \log_2 \delta \leq \frac{N}{2} \log_2 \left( \frac{N}{4e\pi} \right) + \frac{1}{2} \log_2 \left( \frac{\pi N^3}{e^2} \right) + \frac{21}{4N + 10} \]

(19)
Space Filling Advantage: Densest Known Sphere Packings

- Densest known packings for dimensions $\mathcal{N} \leq 48$  
  [Conway, Sloane, 1998]
- Vertical axis: $\log_2 \delta + \mathcal{N}(24 - \mathcal{N})/96$
## Space Filling Advantage: Approximate SNR Gain

<table>
<thead>
<tr>
<th>Dim.</th>
<th>Densest Packing</th>
<th>Name</th>
<th>Highest Kissing Number</th>
<th>Approximate Gain [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \mathbb{Z} )</td>
<td>Integer lattice</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>( A_2 )</td>
<td>Hexagonal lattice</td>
<td>6</td>
<td>0.17</td>
</tr>
<tr>
<td>3</td>
<td>( A_3 \simeq D_3 )</td>
<td>Cuboidal lattice</td>
<td>12</td>
<td>0.29</td>
</tr>
<tr>
<td>4</td>
<td>( D_4 )</td>
<td></td>
<td>24</td>
<td>0.39</td>
</tr>
<tr>
<td>5</td>
<td>( D_5 )</td>
<td></td>
<td>40</td>
<td>0.47</td>
</tr>
<tr>
<td>6</td>
<td>( E_6 )</td>
<td></td>
<td>72</td>
<td>0.54</td>
</tr>
<tr>
<td>7</td>
<td>( E_7 )</td>
<td></td>
<td>126</td>
<td>0.60</td>
</tr>
<tr>
<td>8</td>
<td>( E_8 )</td>
<td>Gosset lattice</td>
<td>240</td>
<td>0.66</td>
</tr>
<tr>
<td>9</td>
<td>( \Lambda_9 )</td>
<td>Laminated lattice</td>
<td>240</td>
<td>0.70</td>
</tr>
<tr>
<td>10</td>
<td>( P_{10c} )</td>
<td>Non-lattice arrangement</td>
<td>336</td>
<td>0.74</td>
</tr>
<tr>
<td>12</td>
<td>( K_{12} )</td>
<td>Coxeter-Todd lattice</td>
<td>756</td>
<td>0.81</td>
</tr>
<tr>
<td>16</td>
<td>( BW_{16} \simeq \Lambda_{16} )</td>
<td>Barnes-Wall lattice</td>
<td>4320</td>
<td>0.91</td>
</tr>
<tr>
<td>24</td>
<td>( \Lambda_{24} )</td>
<td>Leech lattice</td>
<td>196560</td>
<td>1.04</td>
</tr>
<tr>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td>1.35</td>
</tr>
<tr>
<td>( \infty )</td>
<td></td>
<td></td>
<td></td>
<td>1.53</td>
</tr>
</tbody>
</table>
Vector Quantizer with Variable-Length Codes

- Minimize distortion Lagrangian function $J = D + \lambda \cdot R$

$$J = \sum_{\forall k} \int_{C_k} d_N(s, s'_{k}) f(s) \, ds + \lambda \cdot \sum_{\forall k} \ell_k \int_{C_k} f(s) \, ds$$  \hspace{1cm} (20)

**Necessary conditions for optimality**

- Centroid condition for MSE distortion (for representative vectors $s'_{k}$)

$$s'_{k} = \mathbb{E}\{ S | S \in C_k \} = \frac{1}{p_k} \int_{C_k} s f(s) \, ds$$  \hspace{1cm} (21)

- Entropy condition (for codeword length $\ell_k$)

$$\ell_k = - \log_2 p_k$$  \hspace{1cm} (22)

- Modified nearest neighbor condition (for quantization cells $C_k$)

$$\alpha(s) = \arg \min_{\forall k} d_N(s, s'_{k}) + \lambda \cdot \ell_k$$  \hspace{1cm} (23)

→ Quantizer design: Similar to entropy-constrained Lloyd algorithm
Chou-Lookabaugh-Gray (CLG) Algorithm: ECVQ

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

Iterative quantizer design (extension of EC Lloyd algorithm)

1. Choose initial set of reconstruction vectors \( \{s'_k\} \) and codeword lengths \( \{\ell_k\} \)

2. Associate all vectors of the training set \( \{s_n\} \) with one of the quantization cells \( C_k \) according to

\[
\alpha(s_n) = \arg \min \forall_k d(s_n, s'_k) + \lambda \cdot \ell_k
\]

3. Update the reconstruction vectors \( \{s'_k\} \) according to

\[
s'_k = \arg \min_{s' \in \mathbb{R}^N} \mathbb{E}\{ d_N(S, s') \mid \alpha(S) = k \}
\]

4. Update the codeword lengths \( \{\ell_k\} \) according to

\[
\ell_k = -\log_2 p_k
\]

5. Repeat the previous three steps until convergence
Shape Advantage: Results for Gaussian IID ($N = 2, R = 2$)

Result of CLG algorithm for Gaussian iid

- Gain of ECVQ compared to ECSQ is 0.26 dB
- Gain of VQ compared to SQ with fixed-length codes is 0.37 dB
Shape Advantage: Results for Laplace IID ($N = 2$, $R = 2$)

Result of CLG algorithm for Laplace iid

- Gain of ECVQ compared to ECSQ is 0.20 dB
- Gain of VQ compared to SQ with fixed-length codes is 1.32 dB
Result of CLG algorithm for Gaussian iid

- Gain of ECVQ compared to ECSQ is 0.17 dB (only space filling gain)
- Gain of VQ compared to SQ with fixed-length codes is 0.9 dB
Result of CLG algorithm for 2D Laplace iid

- Gain of ECVQ compared to ECSQ is 0.17 dB (only space filling gain)
- Gain of VQ compared to SQ with fixed-length codes is 1.84 dB
Summary on Shape Advantage

- When comparing ECSQ with ECVQ for iid sources:
  The gain due to $N > 1$ reduces to the space filling gain
- VQ with fixed-length codes can also exploit the gain that ECSQ shows compared to SQ with fixed-length codes
Memory Advantage: Results for Gauss-Markov with $\rho = 0.9$

VQ results from LBG algorithm for Gauss-Markov source with $\rho = 0.9$

- $R = 1$ bit/scalar
- $R = 2$ bit/scalar
- $R = 3$ bit/scalar
- $R = 4$ bit/scalar

LBG algorithm has been extended by re-inserting discarded symbols $s'_k$ using random choices.
Memory Advantage: Results for Gauss-Markov with $\rho = 0.9$

- Gains are additive from space-filling, shape and memory effects
Summary on Memory Advantage

- Largest gain to be made if source contains statistical dependencies
- Exploiting the memory advantage is one of the most relevant aspects of source coding (shape advantage can be obtained using entropy coding)
Vector Quantizer Advantage for a Gauss-Markov Source

Gauss-Markov source with correlation factor $\rho = 0.9$

- Fixed-Length Coded SQ ($K=1$) (Panter-Dite Approximation)
- ECSQ using EC Lloyd Algorithm
- VQ, $K=2$ (e)
- VQ, $K=2$ using LBG algorithm
- VQ, $K=5$ (e)
- VQ, $K=10$ (e)
- VQ, $K=100$ (e)

$SNR \ [dB]$

$R(D)$

$R \ [bit/scalar]$
Vector Quantization with Structural Constraints

Vector Quantization Performance
- Performance gains compared to scalar quantization
- Vector quantizers can asymptotically achieve rate-distortion curve for $N \to \infty$

Vector Quantization Complexity
- Increased storage and computation requirements; additional delay
- Complexity increases with dimension $N$
- Unconstrained vector quantizers rarely used in practice

Reduce complexity by imposing structural constraints
- Tree-Structured Vector Quantizer
- Gain-Shape Vector Quantizer
- Predictive Vector Quantizer
- Lattice Vector Quantizer (special case: Transform Coding)
- Trellis-Coded Quantization
Part Summary

Scalar Quantization: Lloyd Quantizer
- Minimizes distortion $D$ for given number $K$ of quantization intervals
- Interpretation: Optimal quantizer for fixed-length entropy coding
  - Centroid condition
  - Nearest neighbor condition

Entropy-Constraint Scalar Quantization (ECSQ)
- Optimal scalar quantizer in connection with optimal entropy coding
  - Centroid condition (same as for Lloyd)
  - Entropy condition
  - Modified nearest neighbor condition (shifted threshold)

Uniform Reconstruction Quantizer (URQ)
- Uniformly spaced reconstruction levels & simple decoder mapping
- Performance close to optimal ECSQ (with suitable encoder mapping)
  - Most often used quantizer in practice
Vector Quantization (VQ)
- Straightforward extension of scalar quantization to higher dimensions $N$
- Opt. VQ with fixed-length codes: Similar to Lloyd quantizer
- Opt. VQ with variable-length codes: Similar to EC-Lloyd quantizer

Vector Quantizer Advantages
- Space-filling advantage: Unique to vector quantizers (1.53 dB for $N \to \infty$)
- Shape advantage: Can also be exploited by ECSQ
- Memory advantage: Can also be exploited by other coding techniques
- Most important aspect for sources with memory is the “memory advantage”

Vector quantization can achieve rate-distortion bound! – Are we done?
- No! – Complexity of vector quantization is a serious issue!
- **Require lossy coding techniques with high rate-distortion efficiency and a complexity suitable for wide range of implementations**
- Particularly important: **Exploitation of dependencies between samples**