SESSION 2: BINNING SCHEMES & QIM

- Binning Schemes
- Basic Quantization Index Modulation (QIM)
- Distortion-Compensated QIM
- Sparse QIM
- Lattice QIM
- Minimum Distance Decoders
- Practical QIM Codes

Binning Schemes

- Fundamental information-theoretic technique (Marton'79)
- Application: encoding data with side info at transmitter only

$\Rightarrow$ blind data hiding

Example 1: binary length-3 sequence S

- Embed information in S, obtain X
- Distortion constraint: S and X differ at most by 1 bit
  $\Rightarrow \quad S \oplus X \in \{000, 001, 010, 110\}$
  $\Rightarrow \quad$ can embed at most 2 bits
- Spread Spectrum doesn't work!
- Consider instead the following binning scheme:

	$m = 00$	$m = 01$	$m = 10$	$m = 11$
$x =$	**000**	001	**010**	**011**
	111	**110**	101	100

- Find X closest to S in column m [try $S = 010$]
- Error-free decoding

Example 2: LSB Coding

- Consider $\mathcal{S} = \{0, 1, \cdots, 2^b - 1\}$, partition into two bins:

$$\mathcal{S}_e = \{0, 2, \cdots, 2^b - 2\}, \quad \mathcal{S}_o = \{1, 3, \cdots, 2^b - 1\}$$

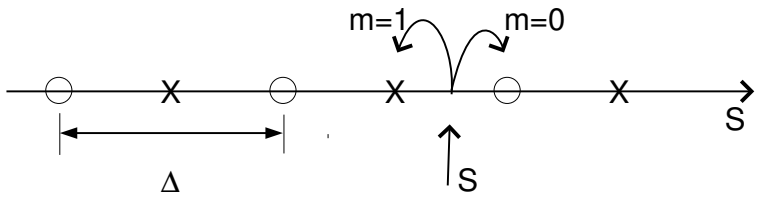
- Binary sequence $s = \{s_1, s_2, \cdots, s_n\}$
- Distortion constraint: $|x_i - s_i| \leq 1$ for all i
- Embed binary sequence $m = \{m_1, m_2, \cdots, m_n\}$
- LSB code can be written as $x_i = m_i + 2 \left\lfloor \frac{s_i}{2} \right\rfloor$

  $\Rightarrow$ choose $\begin{cases} x_i \in \mathcal{S}_e & : \text{if } m_i = 0 \\ x_i \in \mathcal{S}_o & : \text{if } m_i = 1 \end{cases}$

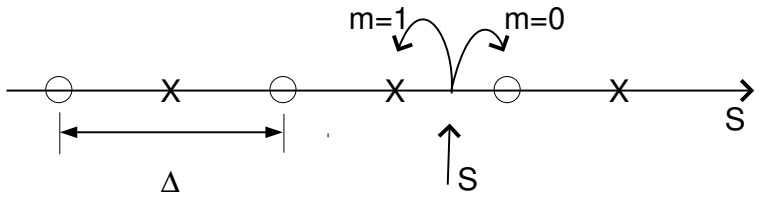
  $\Rightarrow$ view $\mathcal{S}_e$ and $\mathcal{S}_o$ as two bins

Quantization Index Modulation

- Introduced by Chen and Wornell (1999)
- Embed signal-dependent patterns using quantization techniques
- Example: Dithered scalar quantization (1 bit):
  - Let $m \in \{0, 1\}$ (1-bit message), $s \in \mathbb{R}$ (1 sample), no key k
  - Two quantizers $Q_0(s)$ and $Q_1(s)$.
  - Define $x(s, 0) = Q_0(s)$ and $x(s, 1) = Q_1(s)$



Minimum-Distance Decoder



- Two lattices: $\begin{cases} \Lambda_0 = -\frac{\Delta}{4} + \Delta\mathbb{Z} & : \text{circles} \\ \Lambda_1 = \quad \frac{\Delta}{4} + \Delta\mathbb{Z} & : \text{crosses} \end{cases}$
- Attack: $y = x + w$
- Decoder finds closest quantizer point and obtains

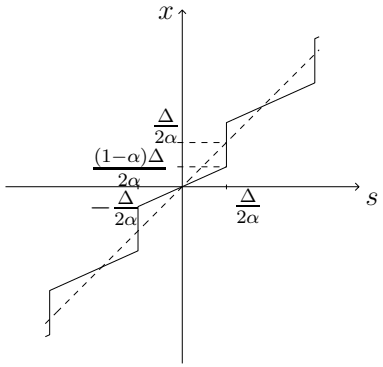
$$\hat{m} = \text{argmin}_{m\in\{0,1\}} \text{dist}(y, \Lambda_m)$$

- No decoding error if $|w| < \Delta/4$
- Binning scheme with noise protection

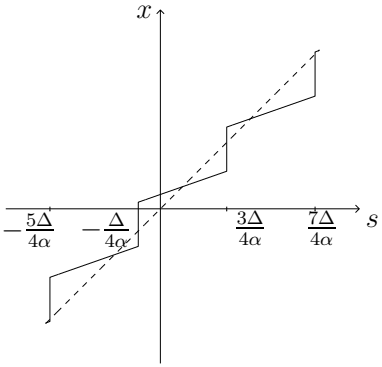
Distortion-compensated QIM

$$X = \begin{cases} Q_0(\alpha S) + (1-\alpha)S & : \ m = 0 \\ Q_1(\alpha S) + (1-\alpha)S & : \ m = 1 \end{cases}$$

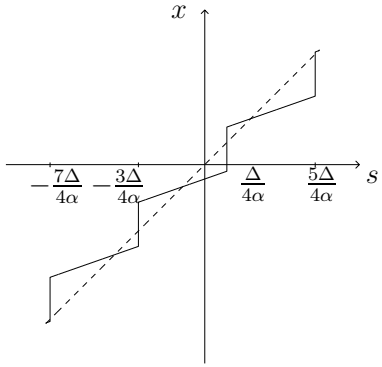
$$= \begin{cases} S + (Q_0(\alpha S) - \alpha S) & : \ m = 0 \\ S + (Q_1(\alpha S) - \alpha S) & : \ m = 1 \end{cases}$$



Prototype $X_{sym}(s)$



$m = 0$

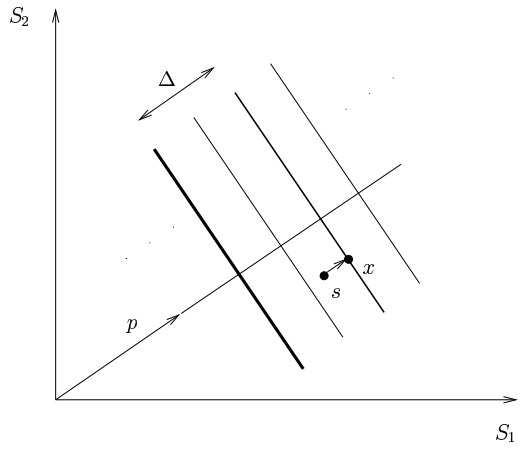


$m = 1$

Sparse QIM (Project & Quantize)

- Choose random unit vector $p \in \mathbb{R}^L$

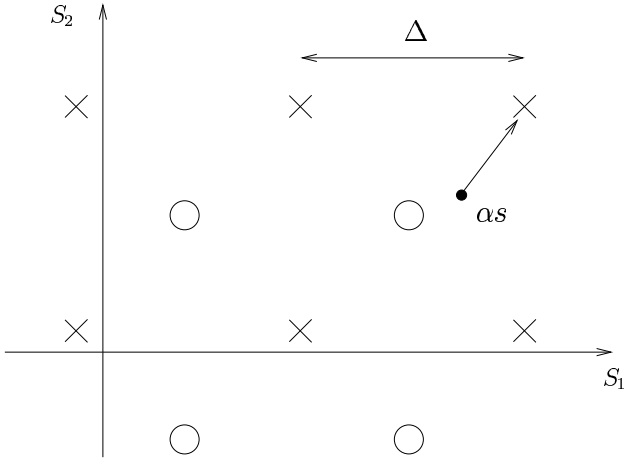
$$x = \begin{cases} s + (Q_0(s^T p) - s^T p)\, p: & m = 0 \\ s + (Q_1(s^T p) - s^T p)\, p: & m = 1 \end{cases}$$



- Decoder: $\hat{m} = \text{argmin}_{m \in \{0,1\}} \text{dist}(y^T p, \Lambda_m)$
- $d_{\min} = \frac{\Delta}{2} = \sqrt{3LD_e}$
- Can also use distortion compensation

Lattice QIM

Example: embed 1 bit in cubic lattice: $m \in \{0,1\}$, $s \in \mathbb{R}^L$



- Distance between Λ_0 and Λ_1 is $d_{\min} = \frac{1}{2}\Delta\sqrt{L} = \sqrt{3LD_e}$
- Decoder implements

$$\hat{m} = \text{argmin}_{m\in\{0,1\}} \text{dist}(y, \Lambda_m)$$

General Principles

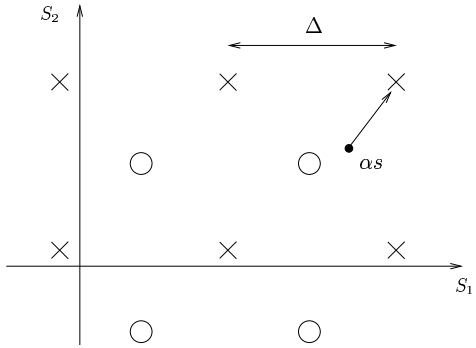
- Minimum distance of lattice code is $d_{\min} = O(\Delta\sqrt{L})$
- Rate of code is $R = 1/L$
- Robustness to noise increases with $\Delta\sqrt{L}/\sigma_w$
- Embedding distortion increases with Δ
- Δ determines tradeoff between robustness and fidelity
- α determines tradeoff between quantization noise and attack noise at receiver (see later why)

General Construction of Lattice QIM Codes

- Use *nested codes*
- Define

Λ/Λ'	$= L$-dimensional *lattice partition*
	$(\Lambda =$ *fine* lattice, $\Lambda' =$ *coarse* lattice$)$
$Q : \mathbb{R}^L \to \Lambda'$	$=$ quantization function
$\mathcal{V}$	$= \{x \in \mathbb{R}^N : Q(x) = 0\} =$ Voronoi cell of Λ'
$\mathcal{C}$	$=$ quotient Λ/Λ'
$\Lambda_m = c_m + \Lambda'$	$=$ coarse lattice translated by $c_m \in \mathcal{C}$

Example, Revisited



$$
\begin{aligned}
\Lambda' &= \Delta\mathbb{Z}^L \\
\Lambda &= D_L^+ = \Delta\mathbb{Z}^L \cup \left(\frac{\Delta}{2}, \cdots, \frac{\Delta}{2}\right) + \Delta\mathbb{Z}^L \\
\mathcal{C} &= \{(0, \cdots, 0), \left(\frac{\Delta}{2}, \cdots, \frac{\Delta}{2}\right)\} \\
\mathcal{V} &= \left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right]^L \quad \Rightarrow \quad D_e = \frac{\Delta^2}{12}
\end{aligned}
$$

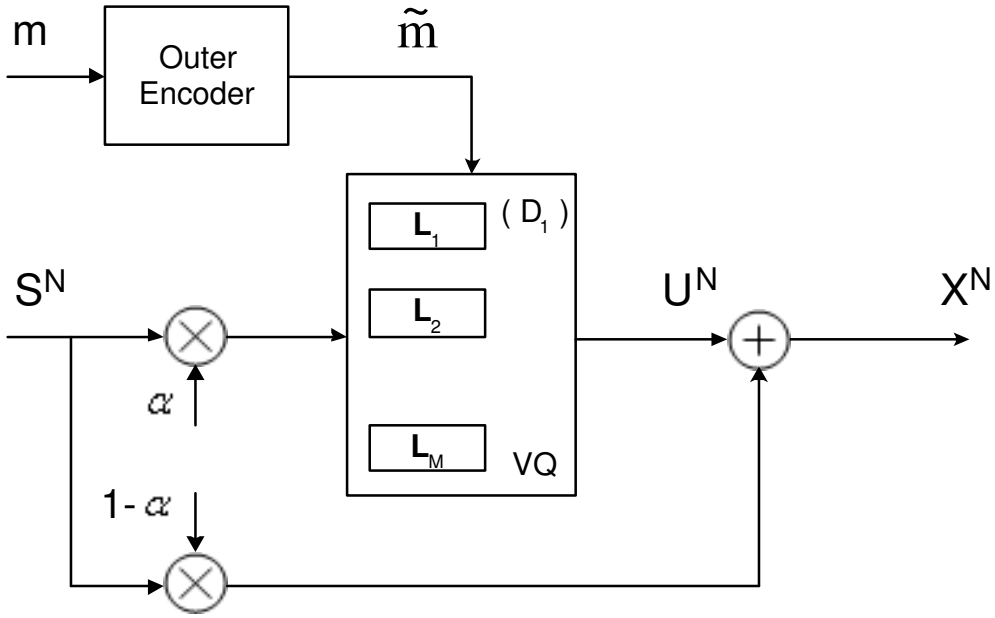
General Principles

- Q should be a *good vector quantizer* with m.s. distortion D_1
  $\Rightarrow$ $\mathcal{V} \sim$ "nearly spherical"
- $\mathcal{C}$ should be a *good channel code* w.r.t. Gaussian noise
  $\Rightarrow$ codewords in $\mathcal{C}$ are "far apart"

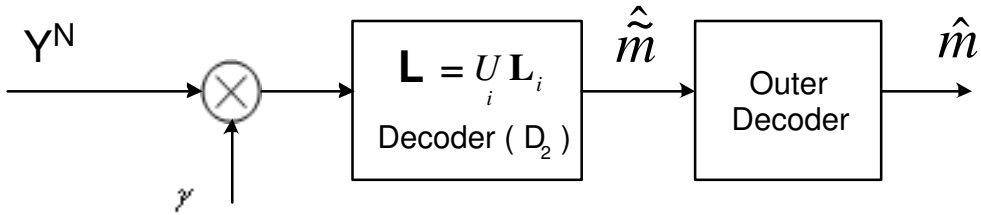
Encoder: outputs $x = (1-\alpha)s + Q_m(\alpha s - c_m) + c_m$
Decoder: outputs $\hat{m} = \text{argmin}_{m \in \mathcal{M}} \text{dist}(\alpha y, \Lambda_m)$

Practical Codes

- In practice, cannot afford arbitrary high-dim. lattices
- Use lattices with special structure:
  - product of low-dimensional lattices
  - trellis-coded scalar quantization
  - classical error correcting codes (Hamming, turbo, etc.)



Encoder



Decoder