# Quantization and Erasures on Frame Representations

Petros Boufounos
petrosb@rice.edu

Rice ECE presentation
Oct. 3, 2006

# Motivation

- Frame representations are overcomplete generalizations to basis expansions.
  - Frame representations are often used in signal processing applications to provide robustness to errors.
  - Oversampling is one example, and the main motivation for the thesis.

- Goal: Exploit the redundancy of frame representations to develop efficient algorithms that provide robustness to quantization and erasures.

- Main Tool: Error compensation using projections.

# Overview

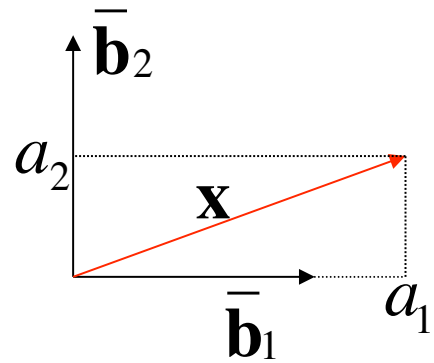
- **Background**
- Quantization and Sigma-Delta noise shaping.
- Erasure Compensation using projections
- Puncturing of Dense Representations
- Summary

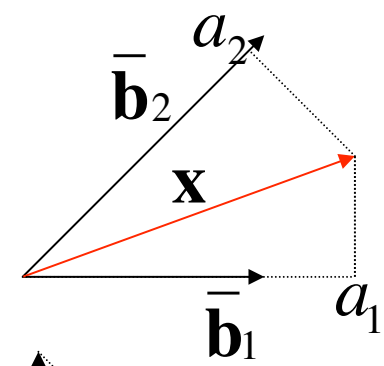
# Basis Expansions

Orthonormal

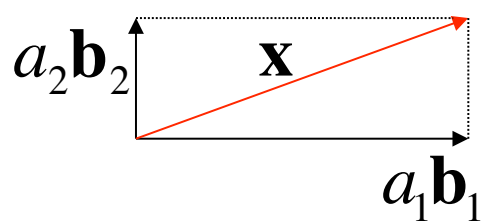
Oblique

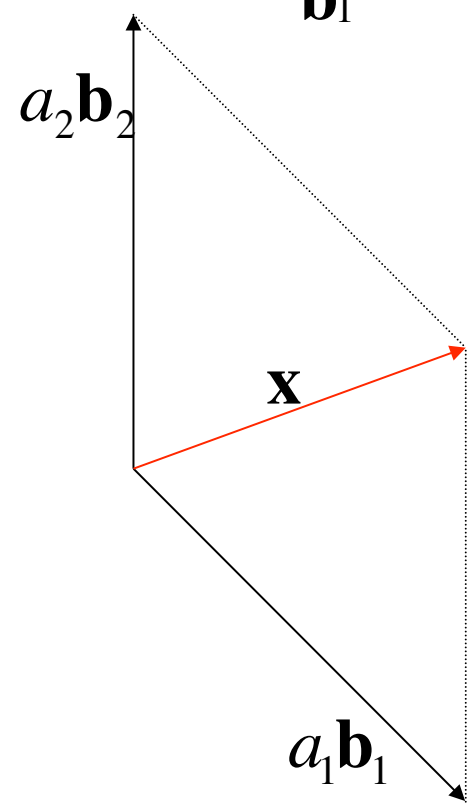
Analysis: $a_k = \left\langle \mathbf{x}, \bar{\mathbf{b}}_k \right\rangle$



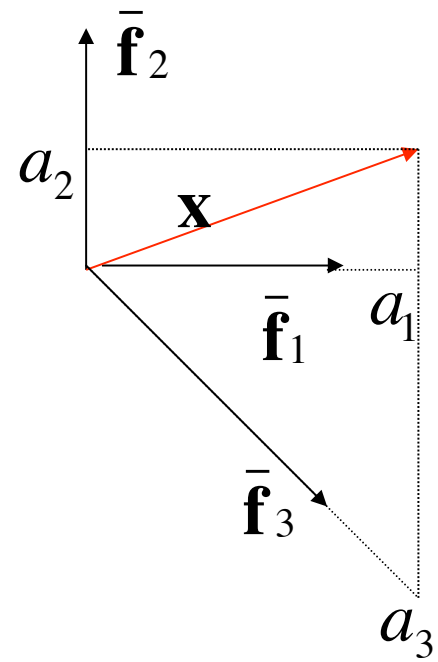

Synthesis: $\mathbf{x} = \sum_k a_k \mathbf{b}_k$



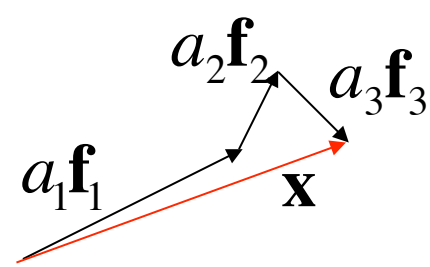



# Frame Representations

Analysis: $a_k = \left\langle \mathbf{x}, \bar{\mathbf{f}}_k \right\rangle$




Synthesis: $\mathbf{x} = \sum_k a_k \mathbf{f}_k$




$N$ dimensional signal space, $M \geq N$ frame vectors/coefficient
Redundancy $r = M/N$

# Examples of Frames and Frame Expansions

## Matrix Operations in $\mathbb{R}^N$ :

Analysis: $$\begin{bmatrix} -\bar{\mathbf{f}}_1- \\ \vdots \\ -\bar{\mathbf{f}}_M- \end{bmatrix}\begin{bmatrix} | \\ \mathbf{x} \\ | \end{bmatrix} = \begin{bmatrix} a_1 \\ \vdots \\ a_M \end{bmatrix} \Leftrightarrow a_k = \left\langle \mathbf{x}, \bar{\mathbf{f}}_k \right\rangle$$

Synthesis: $$\begin{bmatrix} | & & | \\ \mathbf{f}_1 & \cdots & \mathbf{f}_M \\ | & & | \end{bmatrix}\begin{bmatrix} a_1 \\ \vdots \\ a_M \end{bmatrix} = \begin{bmatrix} | \\ \mathbf{x} \\ | \end{bmatrix} \Leftrightarrow \mathbf{x} = \sum_k a_k \mathbf{f}_k$$

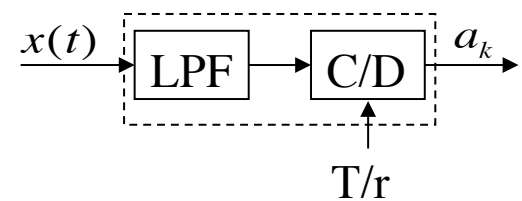
Redundancy:

$r = M/N$

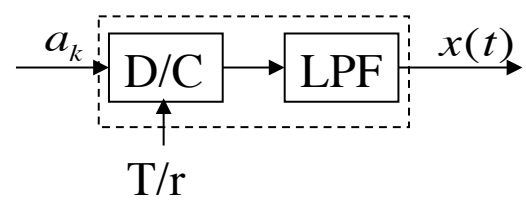
---

## $r$-times Oversampling:

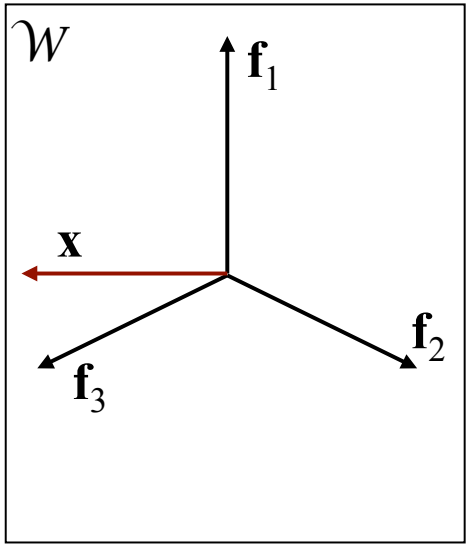
Analysis: $$a_k = \int_{-\infty}^{\infty} x(t) \frac{1}{rT} \operatorname{sinc}\left(\frac{r}{T}t - k\right) dt \Leftrightarrow a_k = \left\langle \mathbf{x}, \bar{\mathbf{f}}_k \right\rangle$$




Synthesis: $$x(t) = \sum_k a_k \operatorname{sinc}\left(\frac{r}{T}t - k\right) \Leftrightarrow \mathbf{x} = \sum_k a_k \mathbf{f}_k$$



# The Frame Operators




Signal Space $\mathcal{W}$

Frame: $\{f_k \in \mathcal{W} | k = 1,\ldots,M\}$

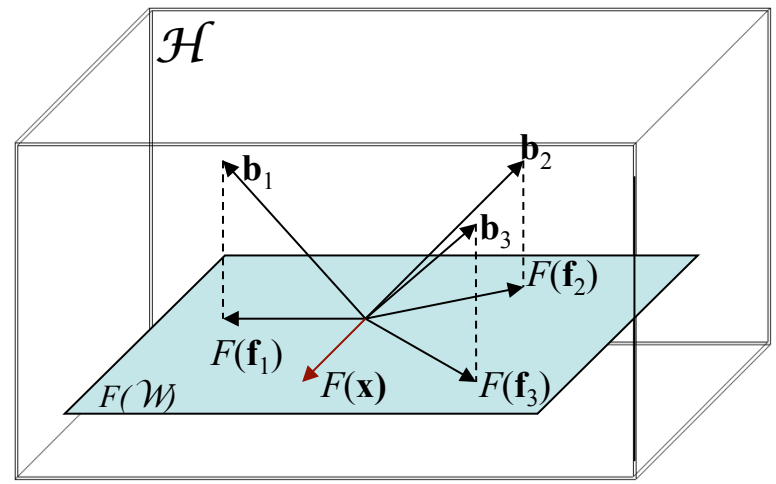
$\dim(\mathcal{W}) = N$

Analysis Operator $F$

$\mathrm{rank}(F) = N$

Synthesis Operator $S = F^*$

$\mathrm{rank}(S) = N$




Coefficient Space $\mathcal{H}$

Orthonormal Basis: $\{b_k \in \mathcal{H} | k = 1,\ldots,M\}$

$\dim(\mathcal{H}) = M \geq N$

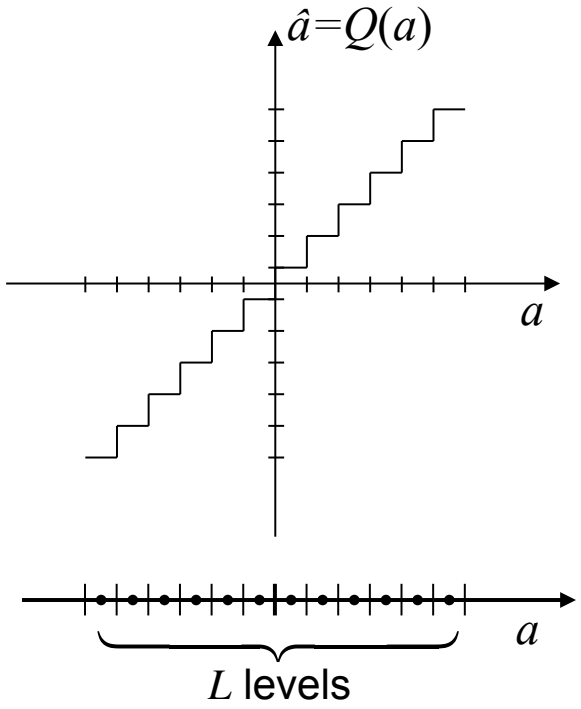
Analysis Operator: $F : \mathcal{W} \to \mathcal{H}$ s.t. $F(\mathbf{x}) = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \mathbf{b}_k = \sum_k a_k \mathbf{b}_k$

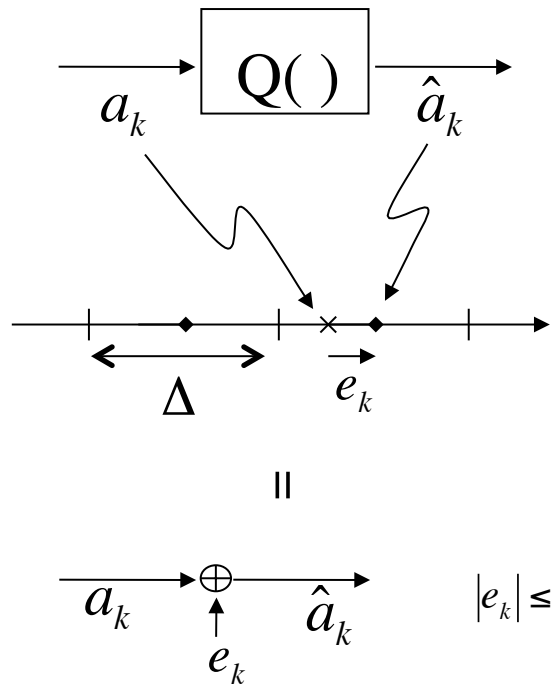
Synthesis Operator: $S : \mathcal{H} \to \mathcal{W}$ s.t. $S(\mathbf{x}) = \sum_k \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{f}_k = \sum_k a_k \mathbf{f}_k$

# Overview

- Background
- **Quantization and Sigma-Delta noise shaping.**
- Erasure Compensation using projections
- Puncturing of Dense Representations
- Summary

# Scalar Quantization



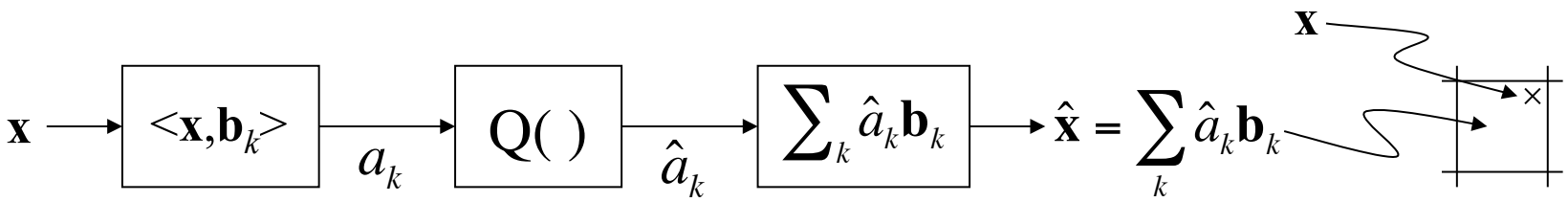

$$|e_k| \le \frac{\Delta}{2}$$

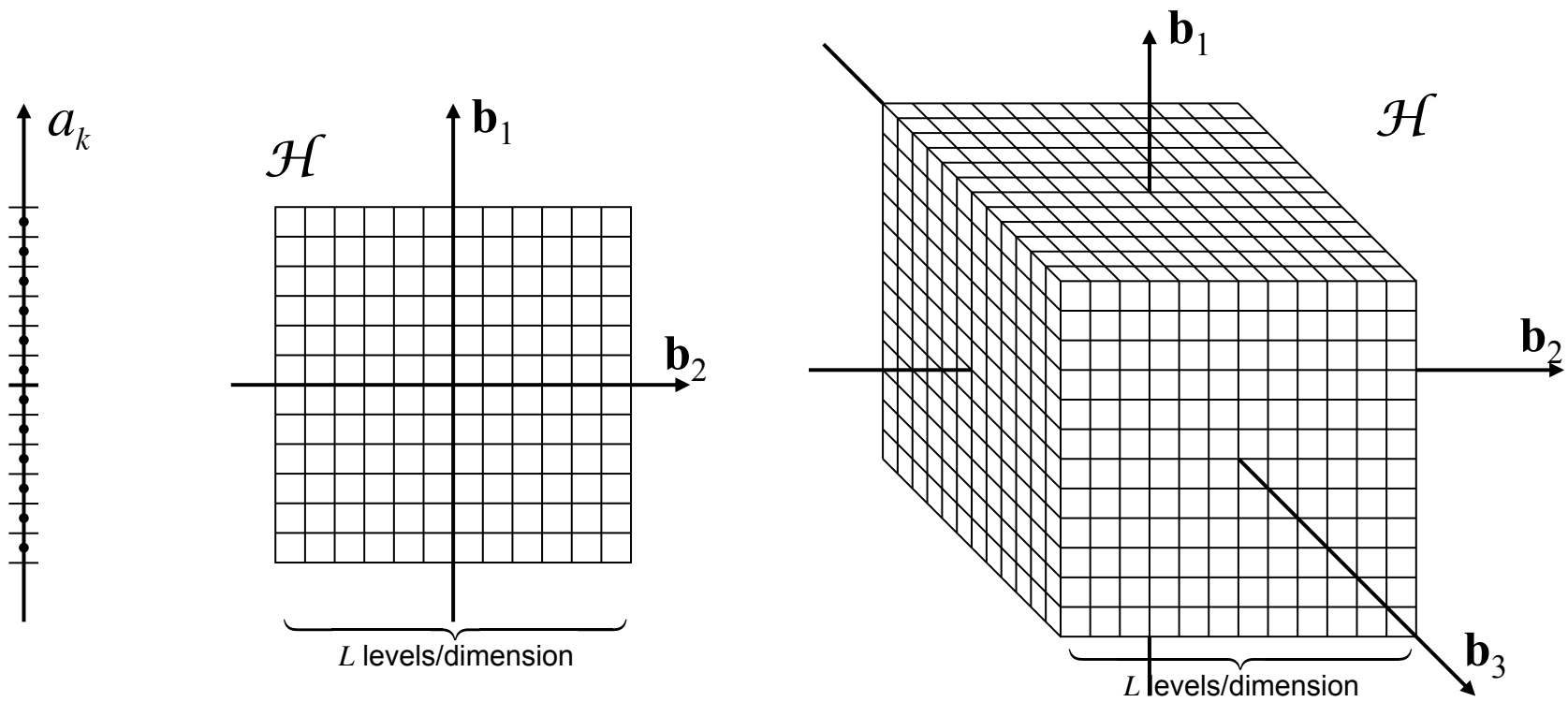
$L$ level quantizer: $\sim\log_2(L)$ bits per coefficient

Additive noise model: $e_k$ uncorrelated, uniform in $\pm\frac{\Delta}{2}$

$$\sigma_e^2 = \frac{\Delta^2}{12} \propto \frac{2^{-2b}}{12}$$

# Quantization of Orthonormal Basis Expansions

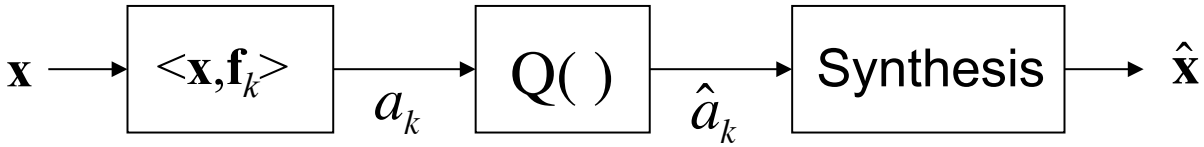


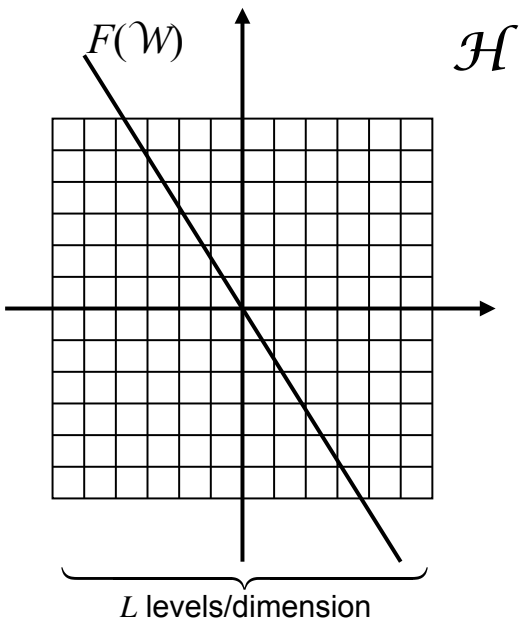



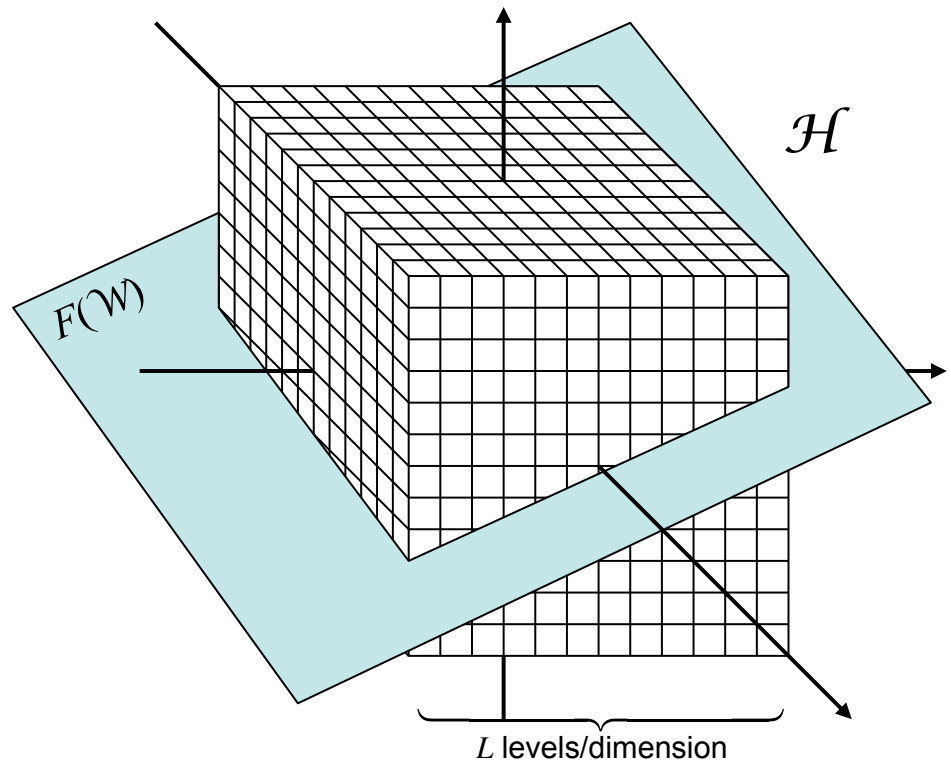

$\log_2 L$ bits per coefficient
$M$ expansion coefficients

$\Rightarrow M \log_2 L$ bits used

# Quantization of Frame Representations

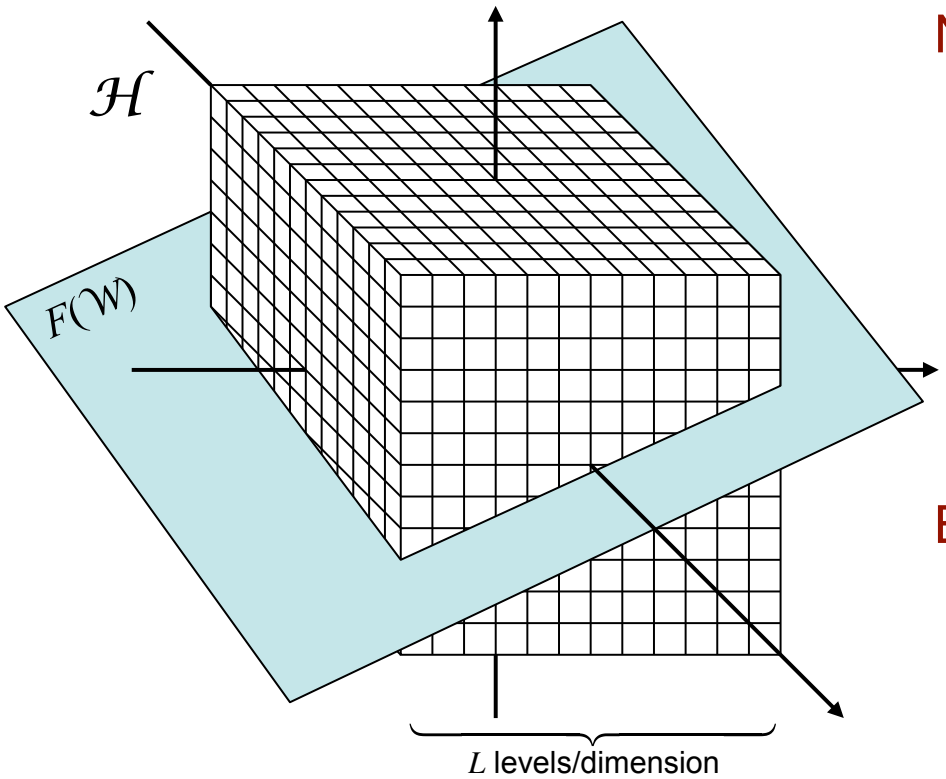



Very few quantization cells are intersected.

# Bounds on Scalar Quantization




Number of cells intersected $I(M,N,L)$:

$$I(M,N,L) \le (2L)^N \binom{M}{N} \le \left(\frac{2LMe}{N}\right)^N = (2Lre)^N$$

$$\dim(\mathcal{W}) = \operatorname{rank}(F) = N$$

$$\dim(\mathcal{H}) = M, \quad r = \frac{M}{N}$$

Bit-use efficiency:

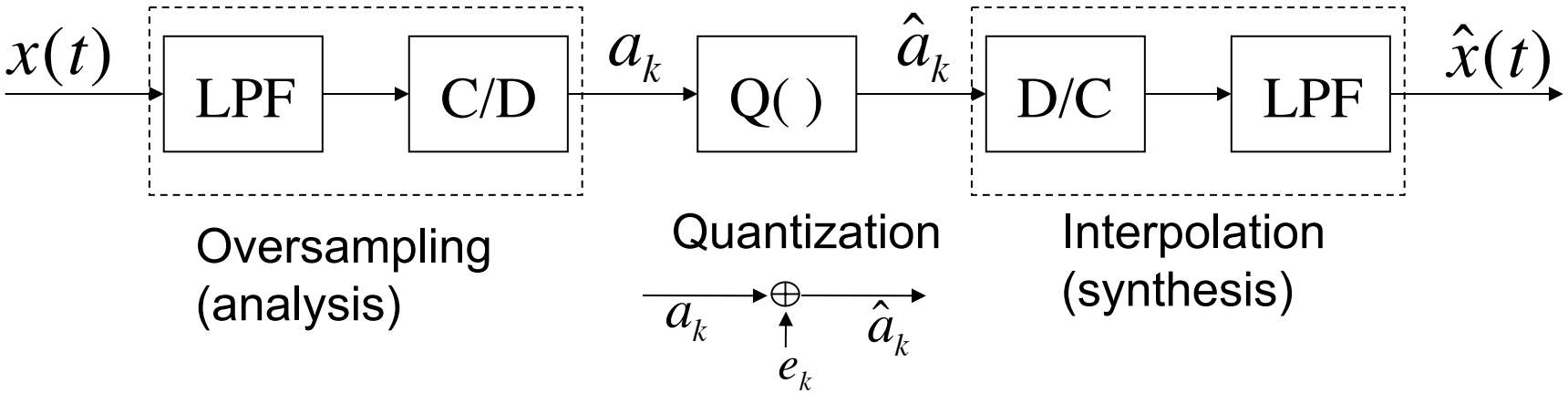
$$\frac{\text{bits necessary}}{\text{bits used}} \le \frac{\log_2(I(M,N,L))}{M \log_2 L} \le \frac{\log_2(2Lre)}{r \log_2 L},$$

Quantization Error Reduction Rate:

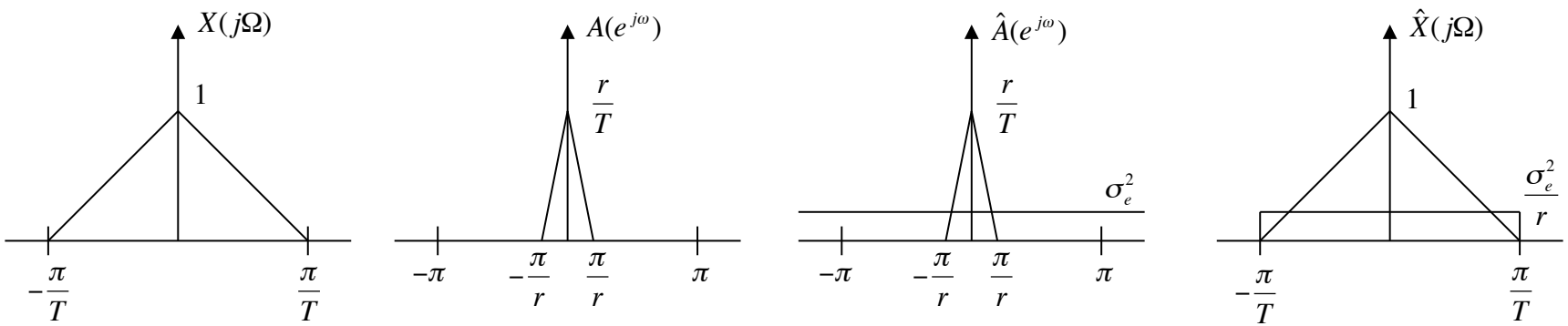
$$\varepsilon^2 \to \Omega(1/r^2)$$

Simple scalar quantization is inefficient.

# Oversampling and Quantization



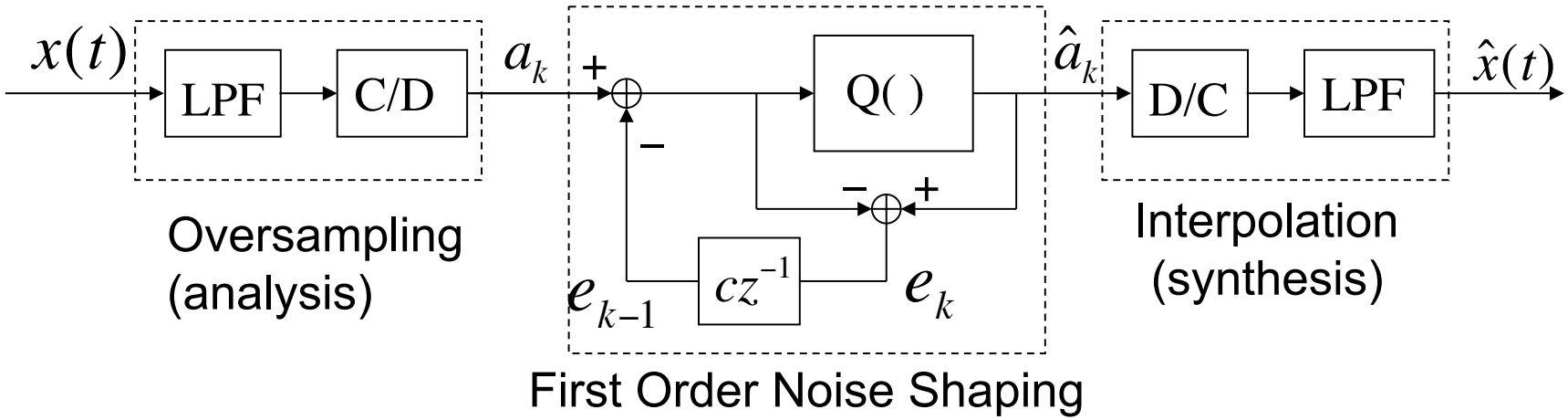

Using the additive noise model, $e_k$ uncorrelated, uniform in $\pm\frac{\Delta}{2}$:

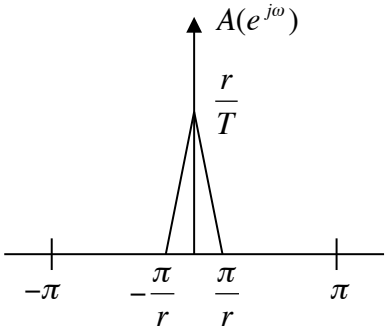


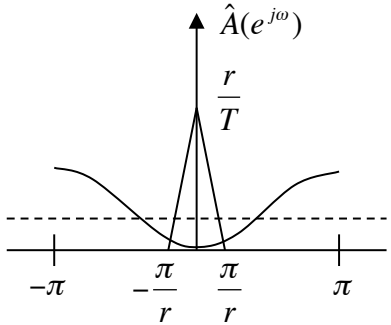

Tradeoff: Gain 1 bit for each 4 times oversampling

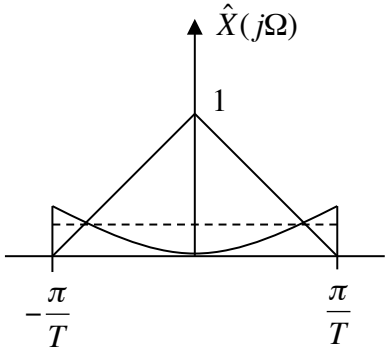
Similar tradeoff for arbitrary frame expansions

# First Order Noise Shaping



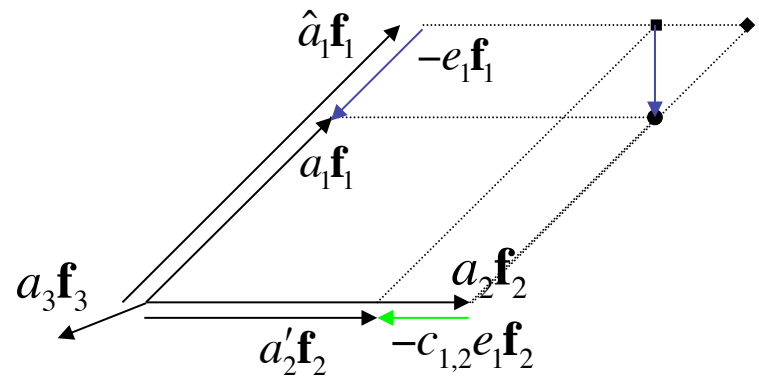




Optimal choice $c = \mathrm{sinc}\left(\frac{\pi}{r}\right)$ ($\approx 1$ for $r \geq 4$)

Can we extend noise shaping to arbitrary frames?

# Error compensation using projections

$$\mathbf{x} = a_1\mathbf{f}_1 + a_2\mathbf{f}_2 + a_3\mathbf{f}_3$$




**1. Quantization**

$$\mathbf{x} = \hat{a}_1\mathbf{f}_1 + a_2\mathbf{f}_2 + a_3\mathbf{f}_3 - e_1\mathbf{f}_1$$

**2. Compensation using projection**

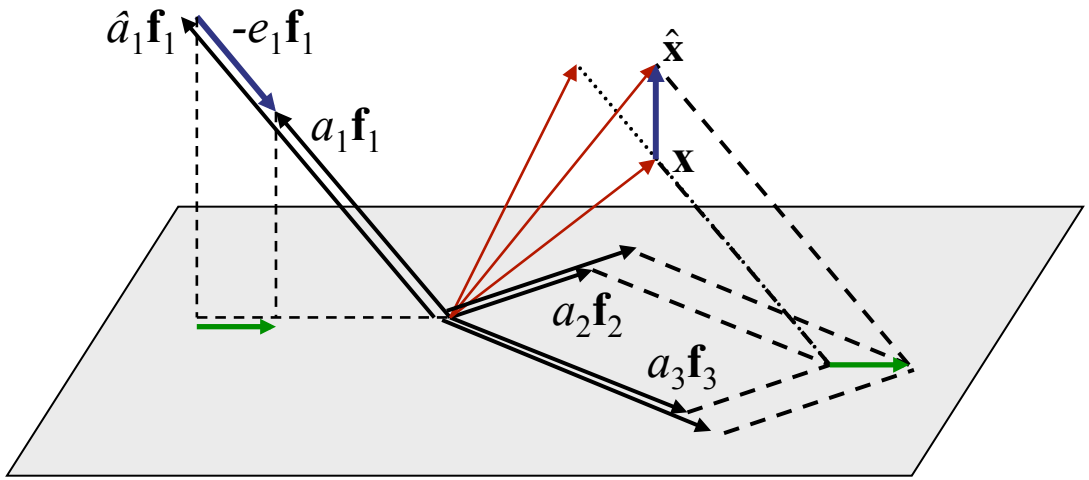
$$a_2' = a_2' - e_1 c_{1,2}$$

$$\mathbf{x} = \hat{a}_1\mathbf{f}_1 + a_2'\mathbf{f}_2 + a_3\mathbf{f}_3 - e_1(\mathbf{f}_1 - c_{1,2}\mathbf{f}_2)$$

Incremental error: $-e_1(\mathbf{f}_1 - c_{1,2}\mathbf{f}_2)$ $\quad \Rightarrow c_{1,2} = \dfrac{\langle \mathbf{f}_1, \mathbf{f}_2 \rangle}{\|\mathbf{f}_2\|^2}$

Compensation linear in the error.
Projection coefficients $c_{i,j}$ can be pre-computed.

# Higher Order Projections




$$\mathbf{x} = a_1\mathbf{f}_1 + a_2\mathbf{f}_2 + a_3\mathbf{f}_3$$

**1. Quantization:**

$$\hat{a}_k = Q(a'_k) = a'_k + e_k$$

**2. Projection:**

$$a'_{k+1} = a_{k+1} - c_{k,k+1}e_k$$

$$\vdots$$

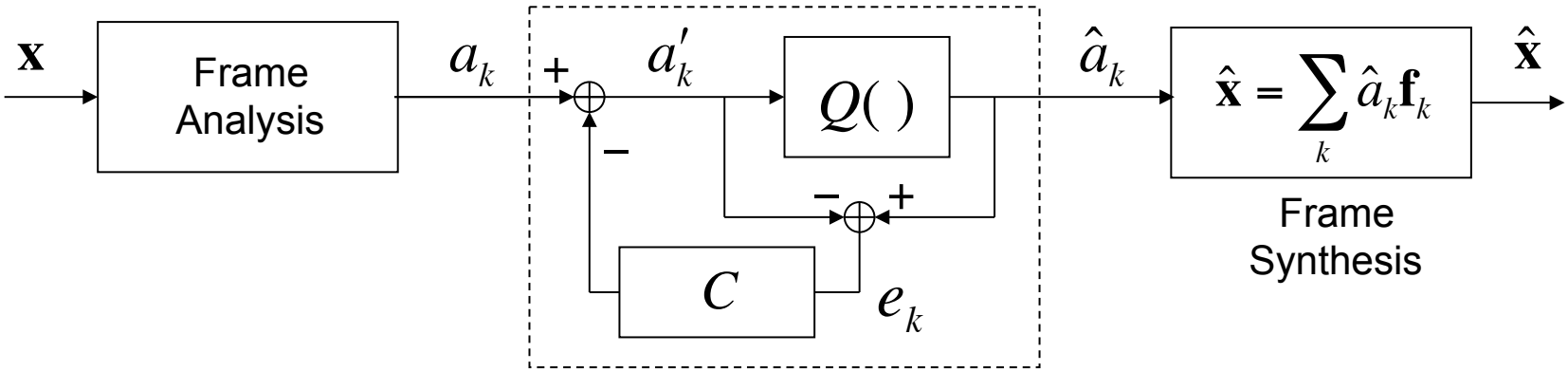
$$a'_{k+p} = a_{k+p} - c_{k,k+p}e_k$$

Projection coefficients $c_{k,k+i}$ satisfy:

$$\underbrace{\begin{bmatrix} \langle \mathbf{f}_{k+1},\mathbf{f}_{k+1}\rangle & \langle \mathbf{f}_{k+2},\mathbf{f}_{k+1}\rangle & \cdots & \langle \mathbf{f}_{k+p},\mathbf{f}_{k+1}\rangle \\ \langle \mathbf{f}_{k+1},\mathbf{f}_{k+2}\rangle & \langle \mathbf{f}_{k+2},\mathbf{f}_{k+2}\rangle & & \langle \mathbf{f}_{k+p},\mathbf{f}_{k+2}\rangle \\ \vdots & & \ddots & \vdots \\ \langle \mathbf{f}_{k+1},\mathbf{f}_{k+p}\rangle & \langle \mathbf{f}_{k+2},\mathbf{f}_{k+p}\rangle & \cdots & \langle \mathbf{f}_{k+p},\mathbf{f}_{k+p}\rangle \end{bmatrix}}_{\mathbf{R}} \underbrace{\begin{bmatrix} c_{k,k+1} \\ c_{k,k+2} \\ \vdots \\ c_{k,k+} \end{bmatrix}}_{\mathbf{c}} = \underbrace{\begin{bmatrix} \langle \mathbf{f}_k,\mathbf{f}_{k+1}\rangle \\ \langle \mathbf{f}_k,\mathbf{f}_{k+2}\rangle \\ \vdots \\ \langle \mathbf{f}_k,\mathbf{f}_{k+p}\rangle \end{bmatrix}}_{\rho}$$

Solution might not be unique; all have the same error.

# System Description




**1. Quantization:**

$$\hat{a}_k = Q(a'_k) = a'_k + e_k$$

**2. Projection (Coefficient Update):**

$$a'_{k+1} = a_{k+1} - c_{k,k+1} e_k$$

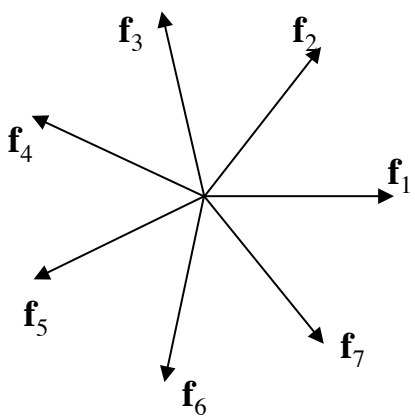
$$\vdots$$

$$a'_{k+p} = a_{k+p} - c_{k,k+p} e_k$$

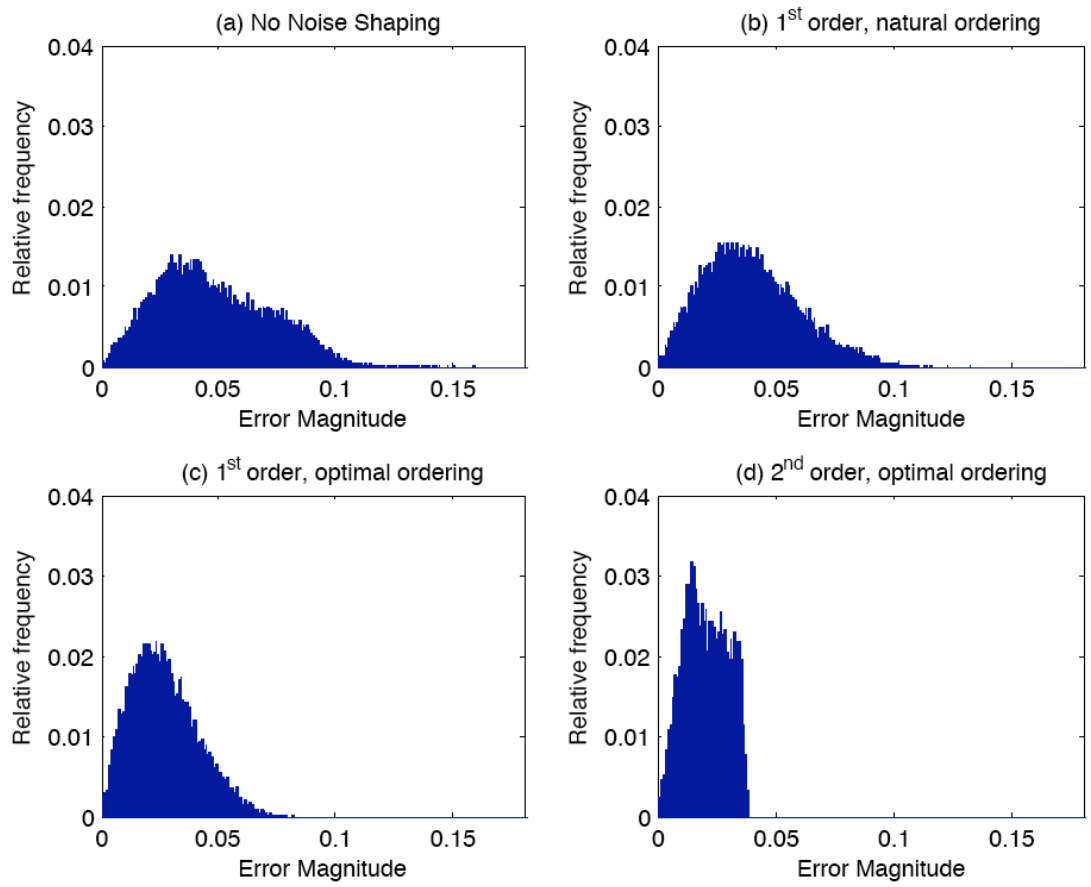
Average and worst case error is reduced.

# Noise Shaping Example

Frame: 7th roots of unity

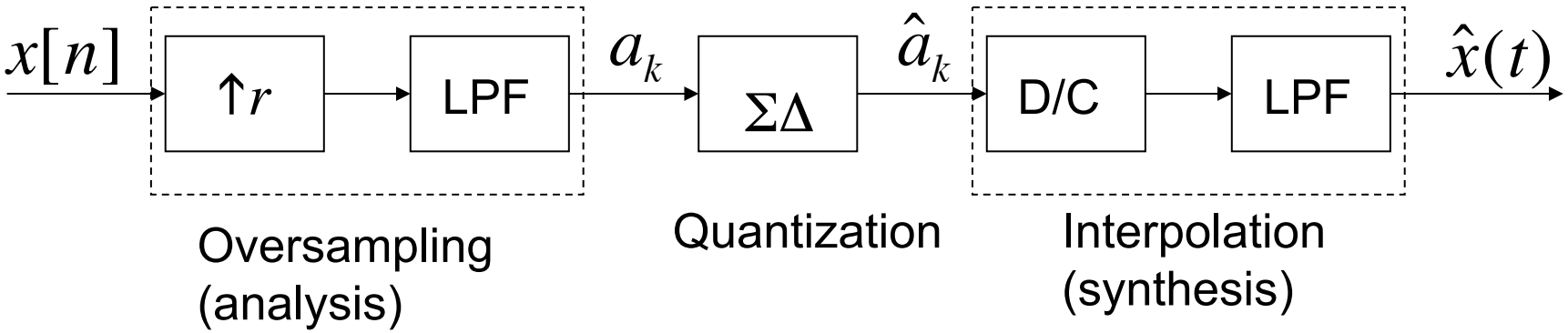


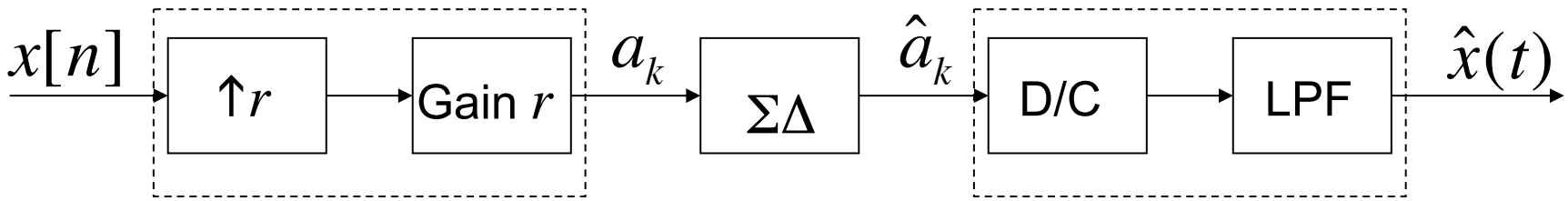

**Histogram of the Error Magnitude**



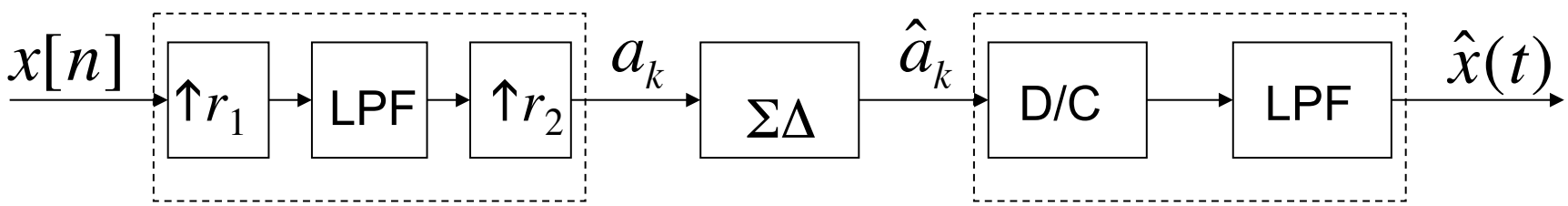

- Random points on the plane, uniform inside the unit cirlce.
- Quantization points: (-7/8, -5/8, -3/8, -1/8, 1/8, 3/8, 5/8, 7/8)
- Optimal ordering (one of many) is: $(\mathbf{f}_1, \mathbf{f}_4, \mathbf{f}_7, \mathbf{f}_3, \mathbf{f}_6, \mathbf{f}_2, \mathbf{f}_5)$

# Simplified D/A Converters



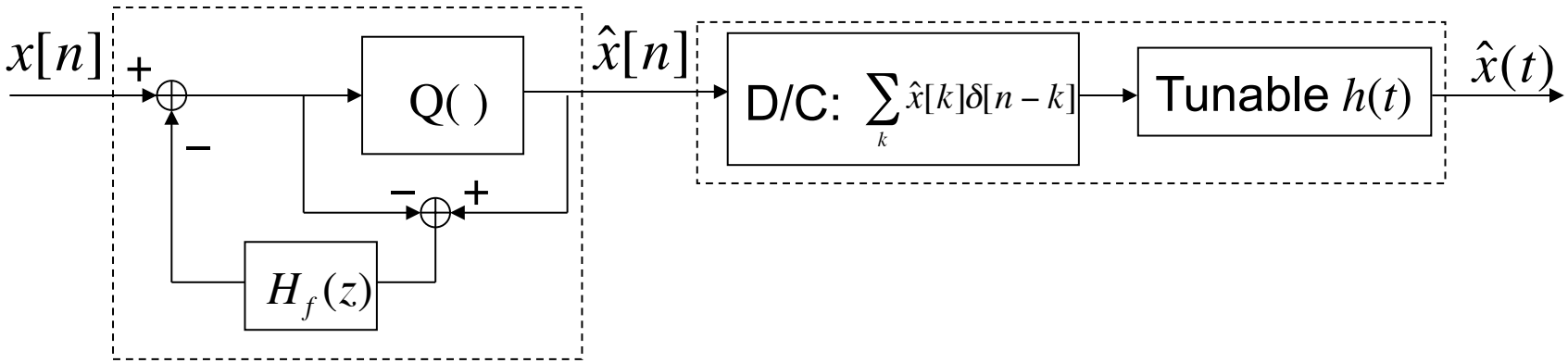




**Simplified D/A structure:**

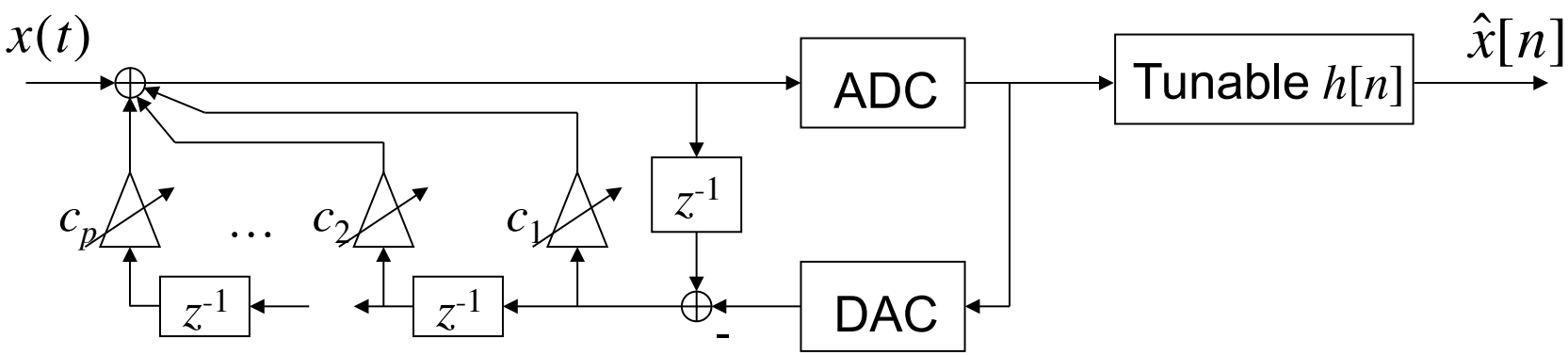


# Tunable DACs and ADCs

**Tunable DAC:**



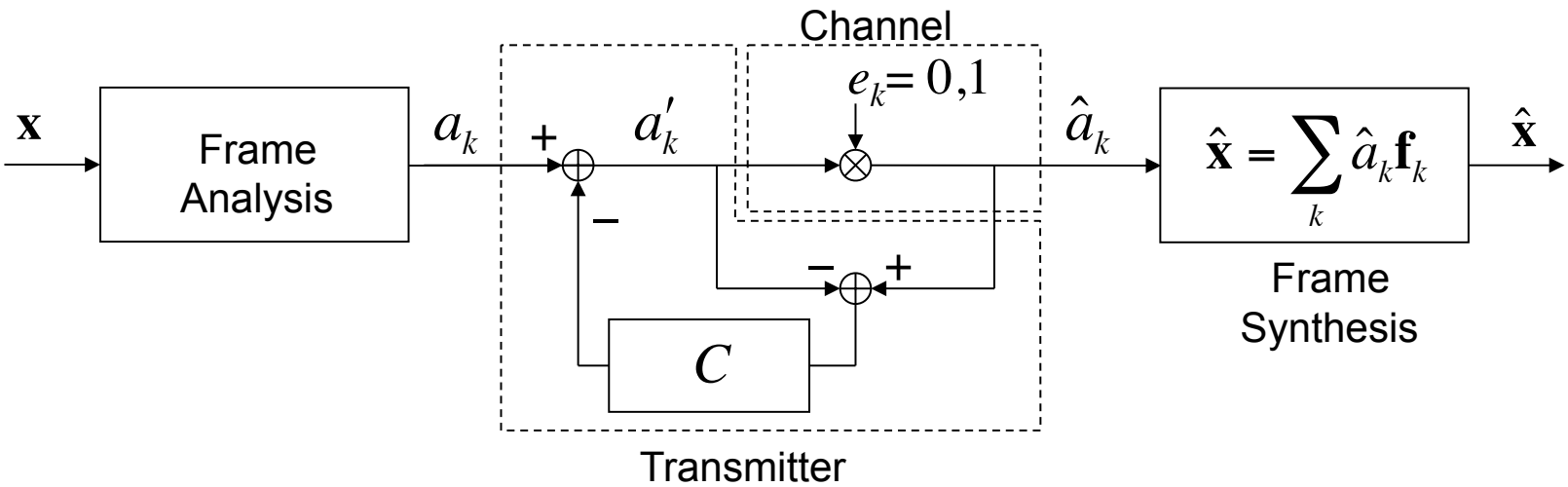

**Tunable ADC:**

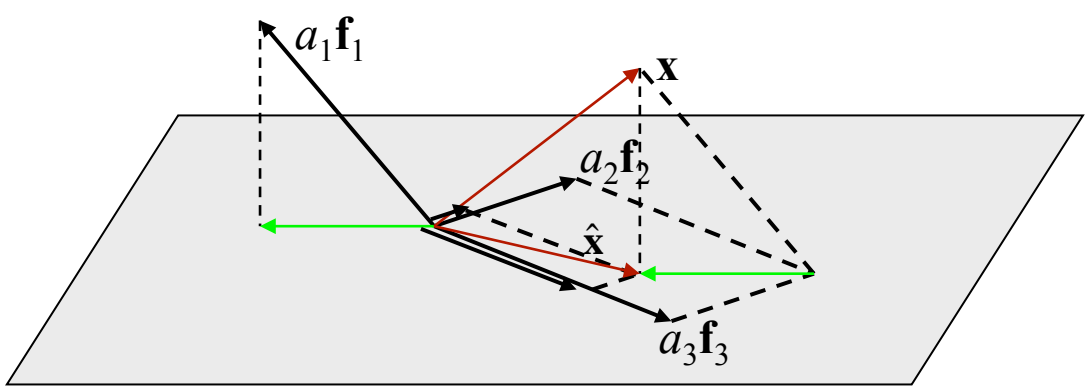


# Overview

- Background
- Quantization and Sigma-Delta noise shaping.
- **Erasure Compensation using projections**
- Puncturing of Dense Representations
- Summary

# Erasures




**1. Erasure:**

$$e_k = 0 \Rightarrow \hat{a}_k = 0$$

**2. Projection:**

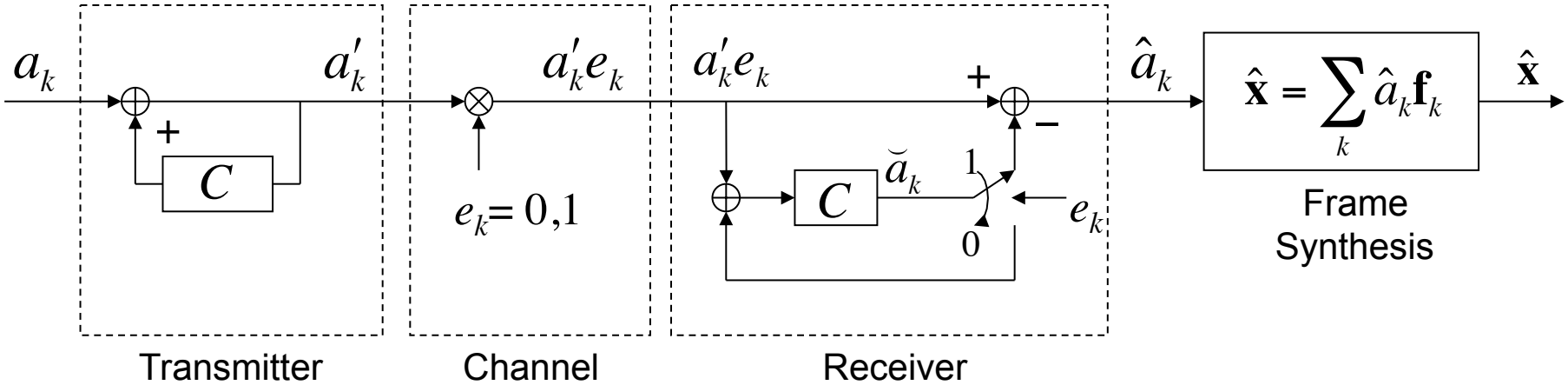
$$a'_{k+1} = a_{k+1} + c_{k,k+1} a_k$$

$$\vdots$$

$$a'_{k+p} = a_{k+p} + c_{k,k+p} a_k$$

The transmitter should know the erasure occurred.

# Pre-compensation with Post-correction



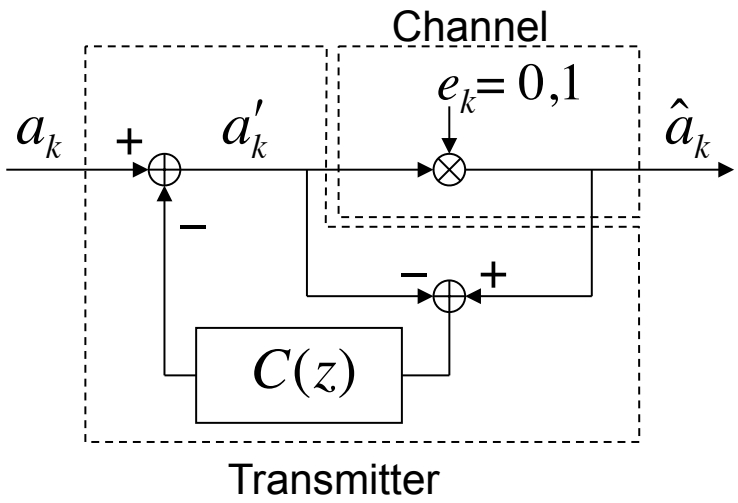

Algorithm:

1. **Project** as if an erasure will occur.
2. **Transmit**, subject to erasures.
3. **Undo** the projection if erasure does not occur . Otherwise **do nothing**.

Input-output behavior is the same.

Only the receiver needs to be aware of the erasure.

# System Stability (LTI case)



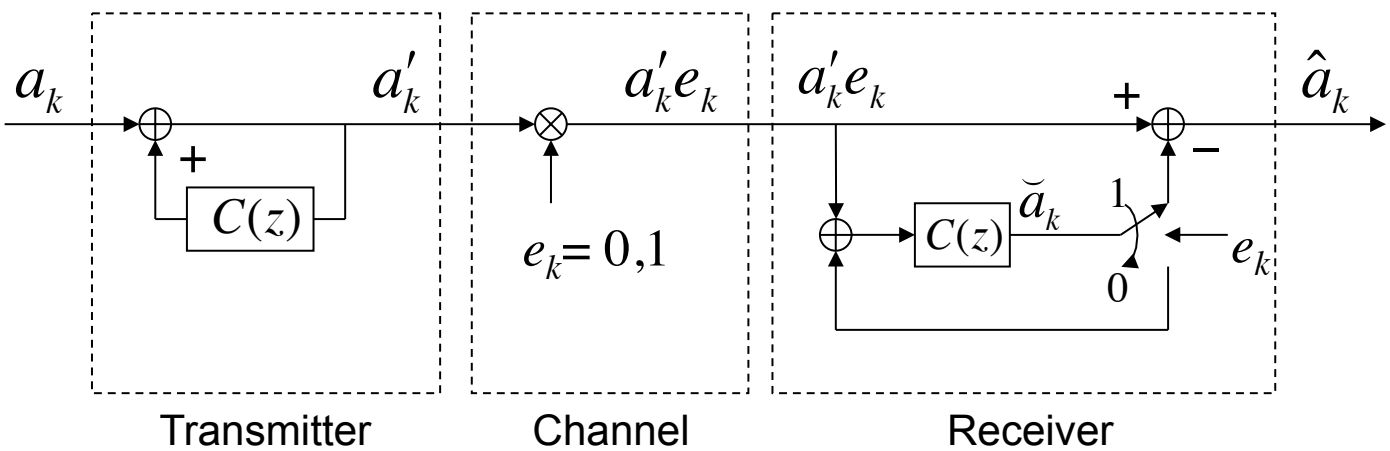

i.i.d erasures, $P$(erasure)=$q$
Shift invariant frame

Necessary condition:

System $\dfrac{1}{1-\sum_{k=1}^{p} q c_k z^{-k}}$ is stable

Sufficient condition:

System $\dfrac{1}{1-\sum_{k=1}^{p} \sqrt{q}|c_k| z^{-k}}$ is stable

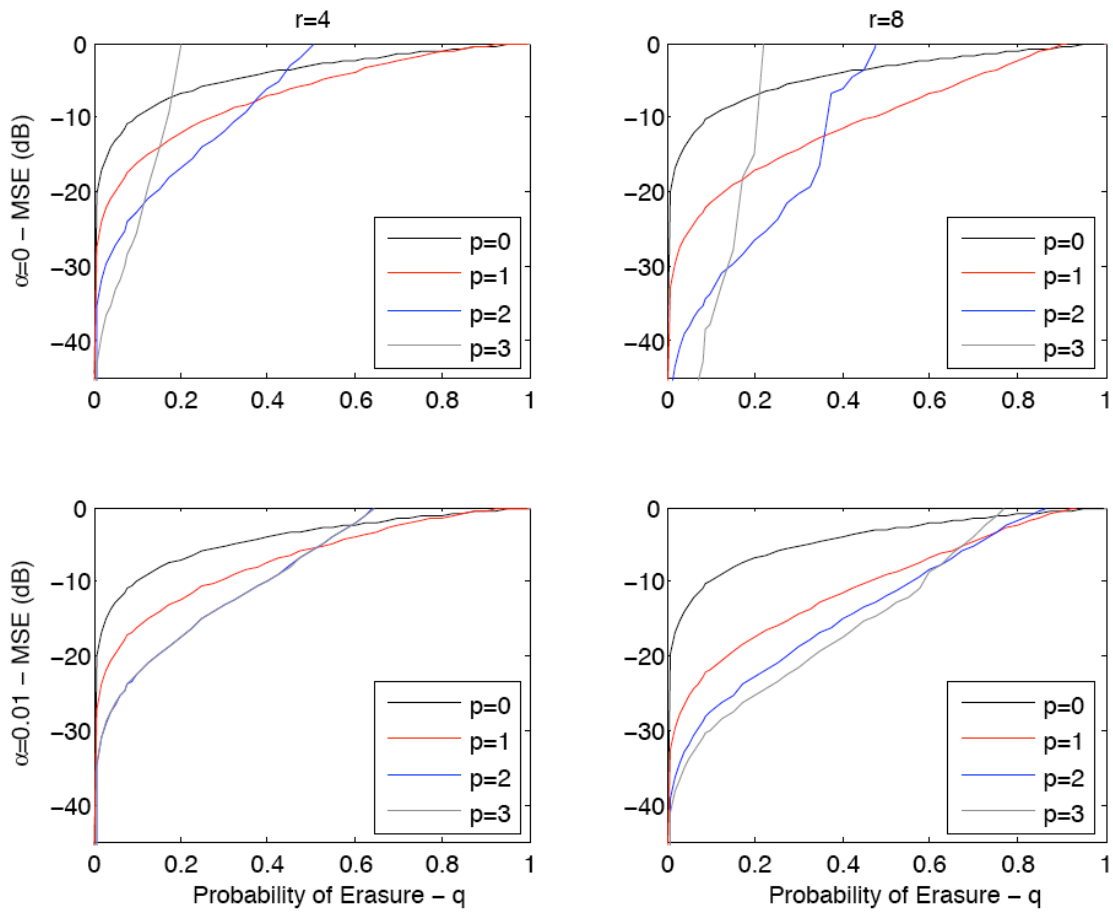



Transmitter:

$$\frac{1}{1-\sum_{k=1}^{p} c_k z^{-k}}$$

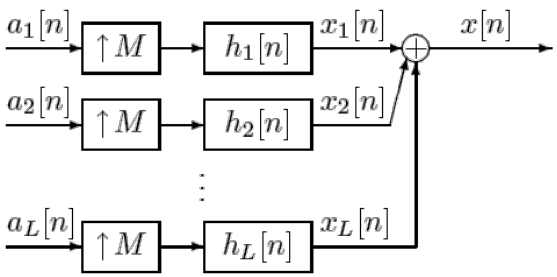
Sub-optimal compensation is sometimes necessary for stability.

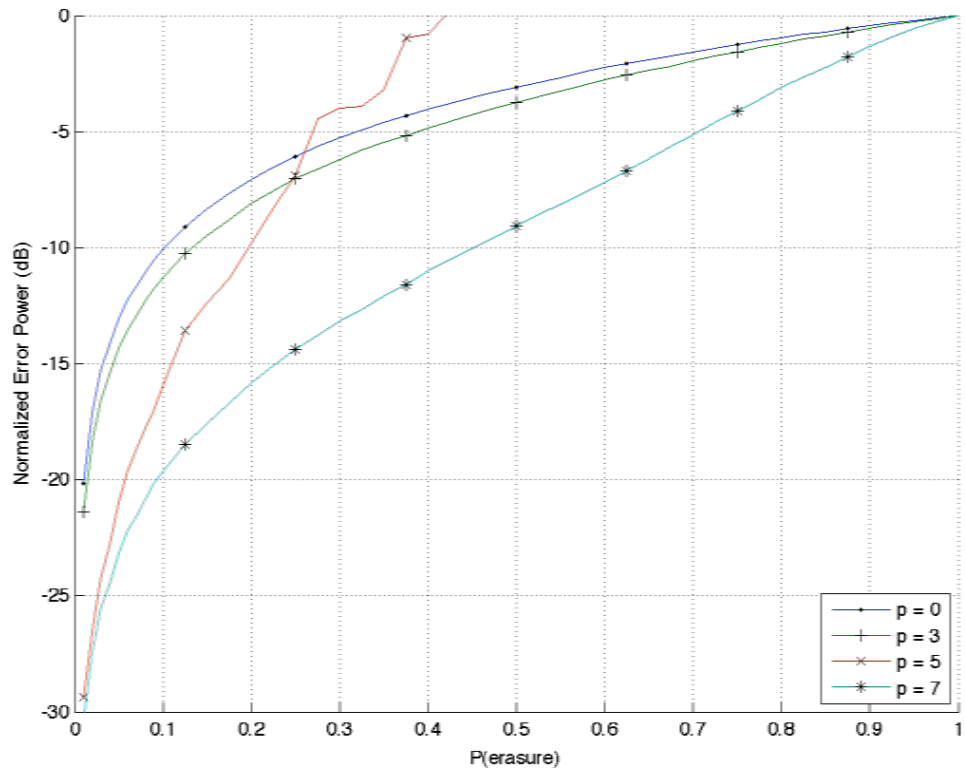
# Simulation with the Oversampling Frame




- Input $a_k$: 0-mean, unit variance, white Gaussian process.
- Erasures $e_k$: i.i.d, $P$(erasure)=$q$
- Oversampling frame with ratio r=4 and 8, approximated with 4096 tap Hamming window FIR filter

# Simulation with Oversampled Filterbank




- $h_l[n]$=Length 4 Harmonic Frame, $M$=1, $L$=8, $r$=8
- Input $a_k$: 0-mean, unit variance, white Gaussian process.
- Erasures $e_k$: i.i.d, $P$(erasure)=$q$

# Overview

- Background
- Quantization and Sigma-Delta noise shaping.
- Erasure Compensation using projections
- **Puncturing of Dense Representations**
- Summary

# Puncturing of Dense Representations

Starting with a dense overcomplete frame representation:

$\{a_1, a_2, \ldots, a_M\}$, such that $\mathbf{x} = \sum_{k=1}^{M} a_k \mathbf{f}_k, \quad \mathbf{x} \in \mathcal{W}, \quad \dim(\mathcal{W}) = N$

Erase coefficients to produce a sparse approximation of $\mathbf{x}$, projecting the erasure error to the remaining ones:

$\{\hat{a}_k | k \in S\}$, such that $\hat{\mathbf{x}} = \sum_{k \in S} \hat{a}_k \mathbf{f}_k \approx \mathbf{x}, \quad \varepsilon = \mathbf{x} - \hat{\mathbf{x}},$

with $S$ being a sparse subset of indices (i.e. $|S|<N$).

Puncturing algorithm:

1. **Select** coefficients to be erased at iteration $i$
2. **Project** the erasure error to **all** the remaining ones
3. **Iterate** from 1, or stop.

# Puncturing Algorithm Properties

- The schedule is dynamically determined.
  - The choice of the schedule affects the performance.
  - Can be determined by the error introduced at each step.
- First $M$-$N$ coefficients can be erased without error.
  - The sequence of erasures can affect subsequent performance
- Each iteration performs an orthogonal projection.
- Error properties
  - The total error $\boldsymbol{\varepsilon}$ is orthogonal to the remaining data
  - The incremental error $\boldsymbol{\varepsilon}_i$ forms an orthogonal set:

$$\boldsymbol{\varepsilon} = \sum_i \boldsymbol{\varepsilon}_i, \quad \left\langle \boldsymbol{\varepsilon}_i, \boldsymbol{\varepsilon}_j \right\rangle \propto \delta_{i,j}$$

$$\Rightarrow \left\| \boldsymbol{\varepsilon} \right\|^2 = \sum_i \left\| \boldsymbol{\varepsilon}_i \right\|^2$$

General algorithm. Details depend on the application.

# Summary

- Error compensation using projections
  - Principle can be used for several types of errors.
- Generalization of Sigma-Delta noise shaping.
  - Improves quantization performance over scalar quantization.
- Erasure Compensation using projections
  - Provides an efficient, causal, system to compensate at the transmitter, even if the transmitter is not aware of the erasure occurrence.
- Puncturing of Dense Representations
  - General algorithm that can be applied in different applications.

# Acknowledgements


- Al Oppenheim
- George Verghese
- Vivek Goyal
- DSP Group at MIT

Questions/Comments?