Prediction and Lossless Audio Coding

Prof. Dr.-Ing. Gerald Schuller

Fraunhofer IDMT & Ilmenau Technical University
Ilmenau, Germany
Use of Redundancy (1)

- For higher correlation between samples → higher redundancy
- For "flat" PSD → low redundancy
- ACF (Auto Correlation Function):
  \[ r_{XX}(\tau) = \lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} x(t)x(t+\tau)dt = E\{x(t)x(t+\tau)\} \]
- PSD (Power Spectrum Density):
  \[ r_{XX}(\tau) \rightarrow S_{XX}(f) = \int_{-\infty}^{\infty} r_{XX}(\tau)e^{-j2\pi f\tau} d\tau \]
  \[ S_X(f) \cdot \overline{S_X(f)} = |S_X(f)|^2 \]
Use of Redundancy (2)

Signal

ACF

PSD

\[ u(t) \]

\[ \varphi_{xx}(\tau) \]

\[ \Phi_{xx}(f) \]

\[ \tau \]

\[ f \]
Predictive Coding (1)

- Use of the correlation of nearby samples
- Method:
  - Prediction of the current sample, using past samples
  - Transmission of the smaller prediction error (smaller code word)
Predictive Coding (2)

- **Encoder**
  \[ e(n) = x(n) - \hat{x}(n) \]
  \[ \hat{x}(n) = \sum_{j=1}^{N} h_j \cdot x(n - j) \]

- **Decoder**
  \[ x(n) = e(n) + \sum_{j=1}^{N} h_j \cdot x(n - j) \]

- **Goal:** Minimize the mean squared error by optimizing the filter coefficients \( h_j \)
  \[ \sigma_e^2 = E\{e^2(n)\} \]
Predictive Coding (3)

• Approach: \[ \frac{\partial \sigma_e^2}{\partial h_j} = 0 \]

\[ \sigma_e^2 = E \left\{ (x(n) - \hat{x}(n))^2 \right\} \]

\[ \frac{\partial \sigma_e^2}{\partial h_j} = 2E \left\{ (x(n) - \hat{x}(n))x(n - j) \right\} \]

Eq. 1: \[ \Rightarrow 0 = E \left\{ (x(n) - \hat{x}(n))x(n - j) \right\} \]

\[ \Rightarrow 0 = r_{XX}(k) - \sum_{j=1}^{N} h_j r_{XX}(k - j), \quad r_{XX}(k) = E \{x(n)x(n - k)\} \]

\[ \Rightarrow r_{XX}(k) = \sum_{j=1}^{N} h_j r_{XX}(k - j) \]
Predictive Coding (4)

- orthogonality principle: for the optimum coefficients the expectation (average) of the error is zero, hence the prediction error is said to be „orthogonal“ to the input signal (Eq. 1)

- auto correlation matrix:

$$R_{XX} = \begin{bmatrix}
    r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(N-1) \\
    r_{xx}(1) & r_{xx}(0) & r_{xx}(N-2) \\
    \vdots & \ddots & \ddots & \vdots \\
    r_{xx}(N-1) & r_{xx}(N-2) & \cdots & r_{xx}(0)
\end{bmatrix}$$
Wiener-Hopf-Equation

- Wiener-Hopf-Equation in matrix description

\[ r_{XX}(k) = \sum_{j=1}^{N} h_j r_{XX}(k-j) \]

\[
\begin{bmatrix}
  r_{XX}(1) \\
  \vdots \\
  r_{XX}(N)
\end{bmatrix} =
\begin{bmatrix}
  r_{XX}(0) & \cdots & r_{XX}(N-1)
\end{bmatrix}
\cdot
\begin{bmatrix}
  h_1 \\
  \vdots \\
  h_N
\end{bmatrix}
\]

\[ r_{XX} = R_{XX} h_{opt} \]

- Vector of ideal filter coefficients:

\[ h_{opt} = R_{XX}^{-1} r_{XX} \]
Deriving Wiener-Hopf with Pseudo Inverses (1)

- Input matrix $X$:

$$X = \begin{bmatrix}
x(0) & x(1) & \cdots & x(N-1) \\
x(1) & x(2) & \ddots & x(N) \\
\vdots & \vdots & \ddots & \vdots \\
x(B) & x(B+1) & \cdots & x(B+N-1)
\end{bmatrix}$$

- Solve equation as close as possible to „$d$“ as our desired signal, in a quadratic sense (minimize sum of quadratic error):

$$X \cdot h \approx d$$

Sequence of “next” values
Deriving Wiener-Hopf with Pseudo Inverses (2)

• Solving the matrix equation with pseudo inverse of the input matrix $X^T$

quadratic matrix $\rightarrow \begin{pmatrix} X^T & X \end{pmatrix} \cdot h = X^T \cdot d$

$h$ which approximates $d$ in quadratic error sense $\rightarrow h = \left( X^T X \right)^{-1} X^T \cdot d$

$\left( X^T X \right)^{-1}$ ACF estimation matrix

$X^T \cdot d$ Cross correlation vector

• This results in the Wiener-Hopf-Equation for block size $B \rightarrow \infty$
Coding Gain (1)

- Prediction error variance/power

\[ \sigma_e^2 = E \{ (x(n) - \hat{x}(n))^2 \} = E \{ x^2(n) + \hat{x}^2(n) - 2x(n)\hat{x}(n) \} \]

- with the orthogonality principle

\[ \sigma_e^2 = E \{ (x(n) - \hat{x}(n))\hat{x}(n) \} \]
\[ \Rightarrow E \{ (x(n)\hat{x}(n)) \} = E \{ \hat{x}^2(n) \} \]
\[ \Rightarrow \sigma_e^2 = E \{ x^2(n) - x(n)\hat{x}(n) \} \]

- Using the Wiener-Hopf-Equation

\[ \sigma_e^2 = \sigma_x^2 - \sum_{j=1}^{N} h_j R_{XX}(j) \]
Coding Gain (2)

- Minimal prediction error (plugging in definition of h)

\[
\sigma_e^2 = \sigma_x^2 - r_{XX}^T R_{XX}^{-1} r_{XX}
\]

\[
\lim_{N \to \infty} \sigma_e^2 = \exp \left[ \frac{1}{2\pi} \int_{-\pi}^{\pi} \log S_{XX}(e^{j\omega}) d\omega \right]
\]

\[
\frac{1}{2} \log(\sigma_e^2) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log S_{XX}(e^{j\omega}) d\omega
\]

- Comparable to bits for subband coding
- Coding gain depends on SFM

Reference: Digital Coding of Waveforms
Jayant, Noll, 1984
Predictive Coding – Subband Coding

- Reduce redundancy in input signal

- Redundancy in input signal is independent of method
  - Predictive coding and subband coding will achieve same results for $N \to \infty$
    different properties result for finite $N$

- Example:
  - few sinusoids $\to$ better prediction with finite $N$
  - narrowband noise $\to$ better subband coding with finite $N$
Lossless Coding

• definition:
  – the decoded and original signal are bit identical / integer identical

• original signal:
  – integer valued audio samples

• lossless coding only removes redundancy, no psychoacoustics or irrelevancy removal is done

• prediction is convenient for lossless compression
  – integer to integer prediction
  – prediction error can easily be made integer valued
  inverse prediction results in original integers!
Predictive Encoder
Predictive Decoder

integer

... Entropy
decod. +

round

integer

integer, as in original

\( x(n) \)

\( z^{-1} \)

\( h_N \)

\( x(n-N) \)

\( h_1 \)

\( x(n-1) \)

exactly the same rounding as in encoder. Same algebra needed, e.g. IEEE defined.
example: rounding of 0.5 needs to be the same
Approaches to Predictive Coding

- How to adapt $h_j$ for real world signals
  - Wiener-Hopf for a block of a certain length
    $\rightarrow$ transmit $h_j$ as side info (most freeware lossless audio coders)
    long blocksize: good for low side info
    short blocksize: good for signal adaptation
  - LMS-Method: Online update derived from Wiener-Hopf for $h_j$ based on past samples
    Normalized LMS:
    $$h_j(n+1) = h_j(n) + \frac{x(n) - \hat{x}(n)}{1 + \lambda \sigma_x^2} x(n-j)$$
    $\rightarrow$ no side info, no blocks necessary
References/Literature:

• Lossless Compression of Digital Audio
  H. Mat, R. Schafer
  IEEE Signal Processing Magazine
  July 2001
  http://ieeexplore.ieee.org

• Perceptual Coding Using Adaptive Pre- and Post-Filters and Lossless Compression
  G. Schuller et al.
  IEEE Trans. On Speech and Audio Signal Processing
  Sept 2002
Lossless Audio Coding with Filter Banks

- Perceptual audio codecs: usually based on filter banks

- Lossless audio codecs: usually based on prediction

- Lossless audio coding using filter banks?
Lossless Audio Coding with Filter Banks

- Problem: Input values integer, output values not integer
- Possible solution: add quantizer

\[
\begin{align*}
    x(n) & \quad h_{N-1}(n) \quad \downarrow N \quad Q \\
    \text{integer} & \quad h_0(n) \quad \downarrow N \quad Q \\
    & \quad g_{N-1}(n) \quad \uparrow N \quad g_0(n)
\end{align*}
\]

- Drawback of this quantization
  - destroys perfect reconstruction
  - has to be very fine or error in time domain has to be coded additionally

float \rightarrow \text{integer} \quad \text{rounding?}
Lifting Scheme (aka „Ladder Network“)

- Goal: Invertible integer-to-integer transform
- Principle: Insert quantizer without destroying perfect reconstruction
- Lifting Scheme or Ladder Network:

\[
\begin{align*}
x_1 &\rightarrow x_1 + a x_0 \\
x_0 &\rightarrow \text{round}(a x_0)
\end{align*}
\]

\[
\begin{align*}
y_1 &= x_1 + \text{round}(a x_0) \\
y_0 &= x_0 \\
\end{align*}
\]

\[
\begin{align*}
x_1' &= y_1 - \text{round}(a y_0) = x_1 \\
x_0' &= y_0 = x_0 \\
\end{align*}
\]

→ invertible integer-to-integer transform
Givens Rotations by Lifting Scheme

- Apply lifting scheme to Givens rotation

- Decomposition:

\[
\begin{bmatrix}
\cos \alpha & -\sin \alpha \\
\sin \alpha & \cos \alpha \\
\end{bmatrix}
= \begin{bmatrix}
1 & \cos \alpha - 1 \\
0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
\sin \alpha & 0 \\
\sin \alpha & 1 \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 \\
0 & 1 \\
\end{bmatrix}
\]
Application to MDCT

- MDCT can be decomposed into
  - Windowing / Time Domain Aliasing
  - DCT of type IV (DCT-IV)

- Both blocks can be decomposed into Givens rotations

- For DCT-IV: Fast algorithms usually provide such a decomposition
MDCT/inverse MDCT by Givens rotations and DCT_{IV}
Integer Modified Discrete Cosine Transform (IntMDCT)

- MDCT can be completely decomposed into Givens rotations
- Apply lifting scheme for each Givens rotation
- Result: Invertible integer approximation of MDCT, called “IntMDCT”
Properties of IntMDCT

• Inherit properties of MDCT
  – perfect reconstruction
  – critical sampling
  – overlapping of blocks
  – good spectral representation of audio signal

• Allows lossless coding in frequency domain by
  entropy coding of integer spectral values
IntMDCT and MDCT of sine wave (1kHz, -20dBFS)
IntMDCT, MDCT and difference values

Item: SQAM, track 64 (Orff: Carmina Burana)
Recent Improvement: Multi-Dimensional Lifting

- Decompose DCT-IV into two DCT-IV of half length
- Further decompose:
  \[
  \begin{bmatrix}
  1024 & 1024
  \end{bmatrix}
  \begin{pmatrix}
  \text{DCT}_{IV} & 0 \\
  0 & \text{DCT}_{IV}
  \end{pmatrix}
  \begin{bmatrix}
  2048 & 2048
  \end{bmatrix}
  \]

- Apply lifting scheme to 2x2 block matrices instead of 2x2 matrices
- Result: Approximation error reduced from \(O(N \log(N))\) to \(O(N)\)
Two blocks of DCT-IV by Multi-Dimensional Lifting

left

right

invertible rounding
Lossless enhancement of perceptual coder (1)

- IntMDCT closely approximates MDCT
- Scalable combination with MDCT-based perceptual codec (e.g. AAC) possible
- Scalable bitstream with two layers allows two stages of decoding
  - Perceptually coded (e.g. AAC @ 128 kBit/s)
  - Lossless (higher, variable bitrate)
Lossless enhancement of perceptual coder (2)

Encoder

Quantization & Coding

Encoding of bitstream

audio in

MDCT

Perceptual Model

IntMDCT

Inverse Quantization & Rounding

Entropy Coding

quantization below masking threshold

perceptually coded bitstream

lossless enhancement bitstream

Prof. Dr.-Ing. K. Brandenburg, bdg@idmt.fraunhofer.de
Prof. Dr.-Ing. G. Schuller, shl@idmt.fraunhofer.de
Lossless enhancement of perceptual coder (3)

Decoder

Decoding of bitstream → Inverse Quantization → Inverse MDCT

Entropy Decoding → Rounding

+ → Inverse IntMDCT

sounds exactly like original

perceptual audio

bit-exact reconstruction

lossless audio
## Compression Results

Results in bits per sample:

<table>
<thead>
<tr>
<th></th>
<th>48 kHz 16 bit</th>
<th>48 kHz 24 bit</th>
<th>96 kHz 24 bit</th>
<th>192 kHz 24 bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAC</td>
<td>1.3</td>
<td>1.3</td>
<td>0.8</td>
<td>0.5</td>
</tr>
<tr>
<td>Enhancement</td>
<td>6.5</td>
<td>14.4</td>
<td>11.0</td>
<td>9.2</td>
</tr>
<tr>
<td>AAC + Enhancement</td>
<td>7.8</td>
<td>15.7</td>
<td>11.8</td>
<td>9.7</td>
</tr>
<tr>
<td>Lossless-only</td>
<td>7.5</td>
<td>15.3</td>
<td>11.6</td>
<td>9.5</td>
</tr>
<tr>
<td>Monkey’s Audio 3.97</td>
<td>7.2</td>
<td>15.2</td>
<td>11.5</td>
<td>9.4</td>
</tr>
<tr>
<td>Simulcast (AAC + Monkey’s Audio)</td>
<td>8.5</td>
<td>16.5</td>
<td>12.3</td>
<td>9.9</td>
</tr>
</tbody>
</table>

Signals: Test set used in ongoing MPEG Lossless Audio activities
Conclusions

- Lossless Audio Coding with filter banks is possible

- Lifting Scheme or Ladder Network is appropriate tool

- IntMDCT allows
  - Efficient lossless audio coding
  - Scalable lossless enhancement of MDCT-based perceptual audio codec (e.g. AAC)