

マルチメディア配信システム

- No.4 画像圧縮技術 (2) -

渡辺 裕

Multimedia Distribution System

- No.4 Image Compression Technology (2) -

Hiroshi Watanabe

画像圧縮技術

- 色信号多重分離
- 直交変換
- サブバンド
- ウエーブレット
- 係数量子化
- エントロピー符号化

Image Compression Technology

- Chrominance Signal Mux/DeMux
- Orthogonal Transform
- Subband
- Wavelet
- Coefficient Quantization
- Entropy Coding

サブバンド

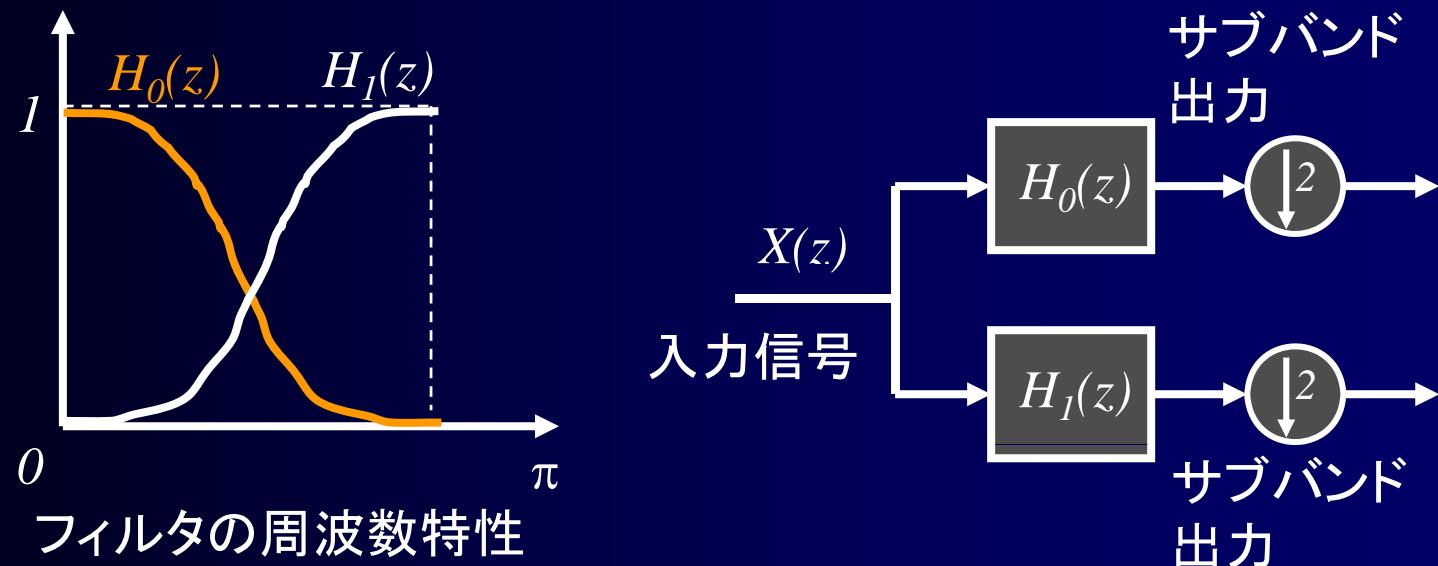
- 通過周波数帯域の限られたフィルタを適用することにより、特定の周波数帯域のサンプルだけを処理する手法
- 用いるフィルタをサブバンドフィルタと呼ぶ
- サブバンド出力を間引き、量子化することにより、変換符号化と等価になる
- チャネルイコライザー（伝送等化器）、オーディオイコライザー（オーディオ周波数調整器）、オーディオ符号化などに応用される
- 主にFIRフィルタが用いられる

Subband

- Processing samples having specified frequency range by applying limited bandwidth filter
- Filter is called “Subband Filter”
- It equals to transform coding by applying decimation and quantization to subband outputs
- Application: Channel equalizer (Transmission equalizer), Audio equalizer, Audio coding
- In most cases, FIR filter

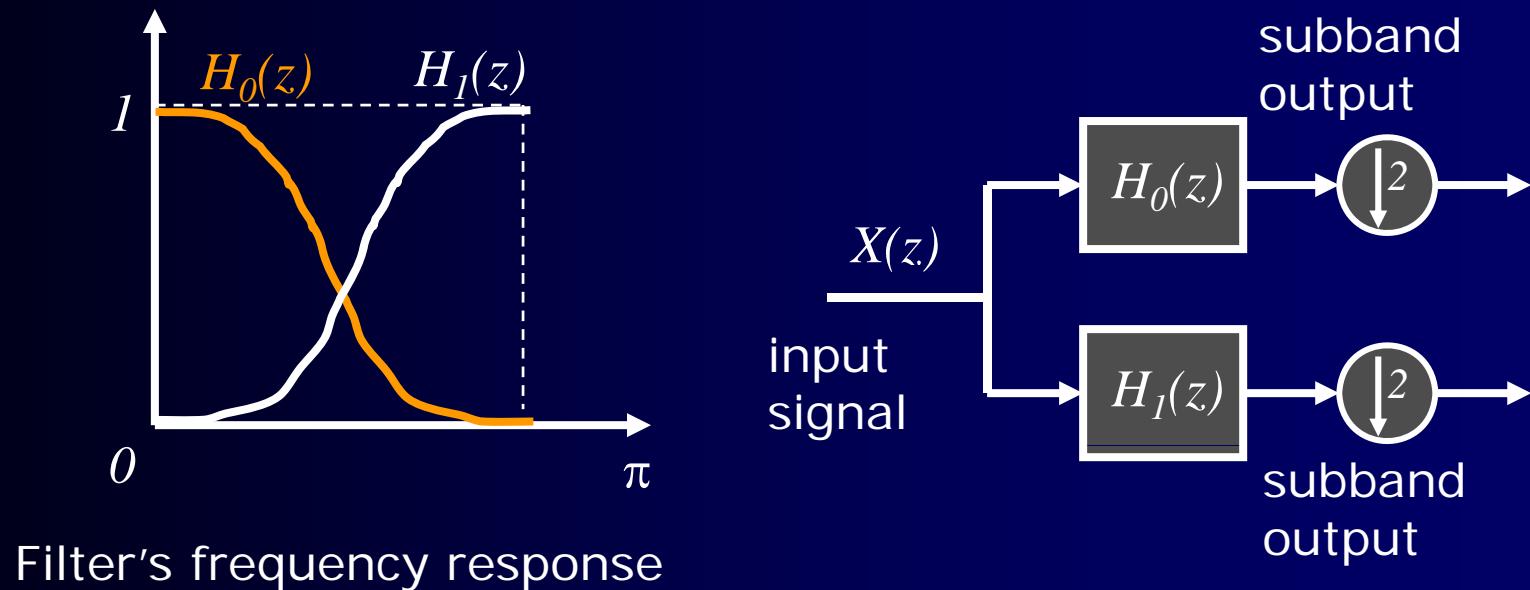
サブバンドフィルタ

- 最も簡単なサブバンド処理は2バンド分割
- 低域通過型FIRフィルタ $H_0(z)$ および高域通過型FIRフィルタ $H_I(z)$ による構成
- サブバンド出力は1:2にダウンサンプル(サブサンプル)される



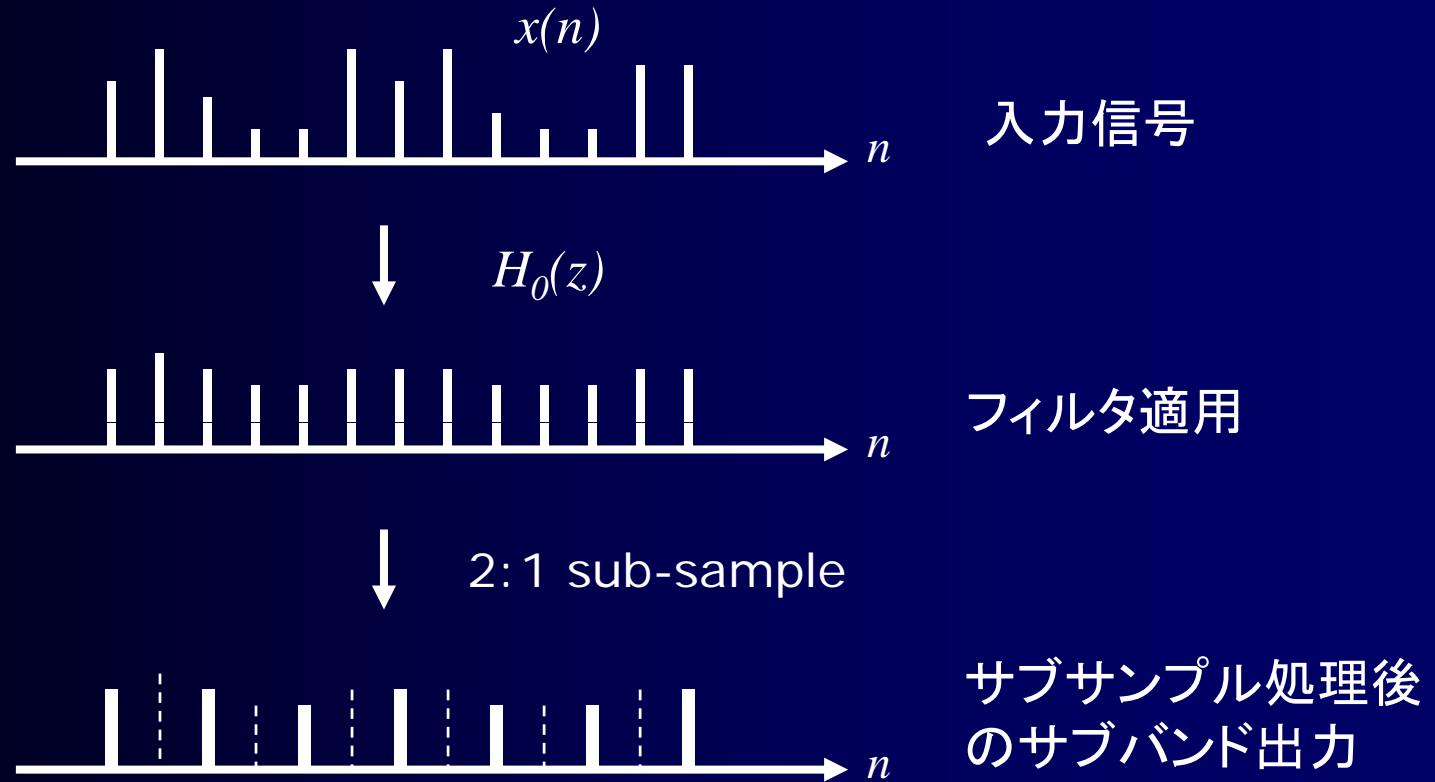
Subband Filter

- The simplest subband is two-band
- Consists of Low pass FIR filter $H_0(z)$ and High pass FIR filter $H_I(z)$
- Subband output is down-sampled to 1:2 (subsample)



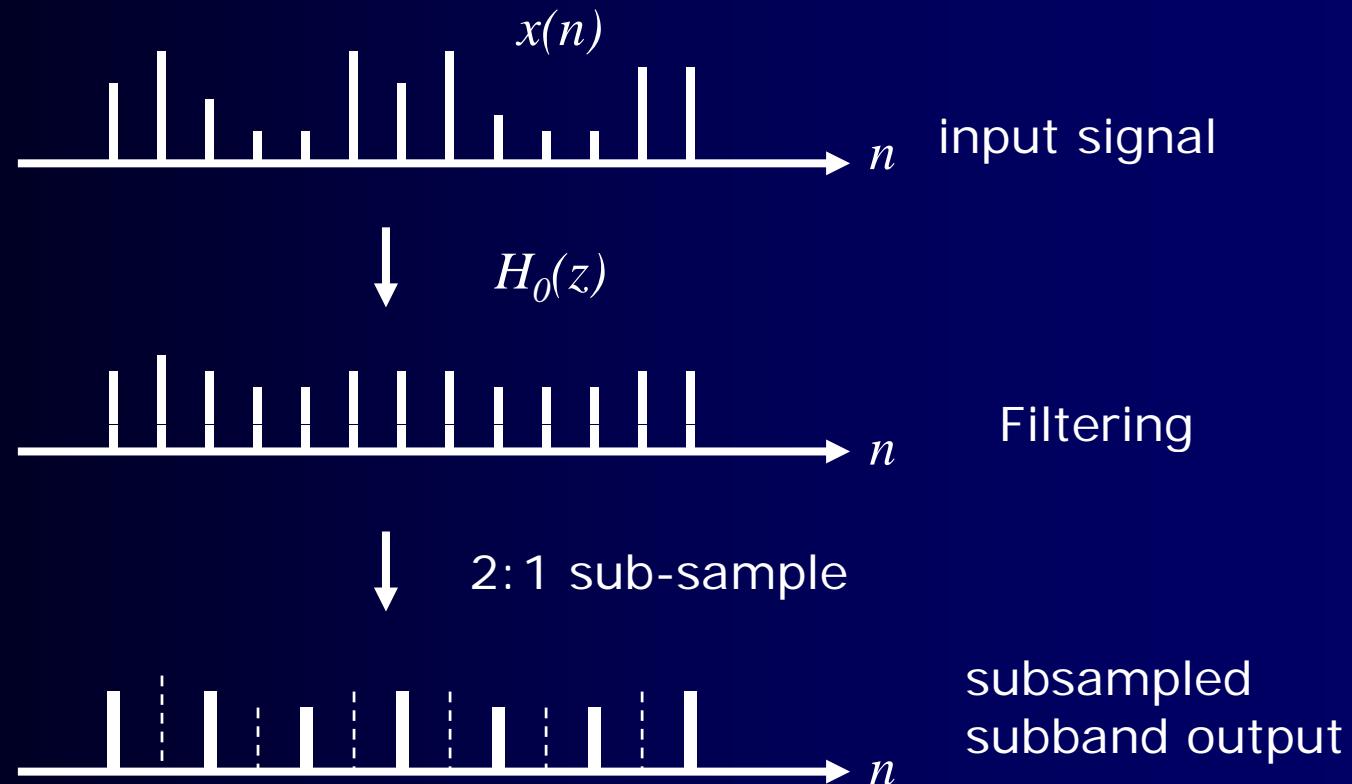
サブバンドフィルタ(2)

■ 2:1サブサンプル



Subband Filter(2)

■ 2:1 Subsample



ハーフバンドフィルタ

■ ハーフバンドフィルタ $H(z)$ の定義

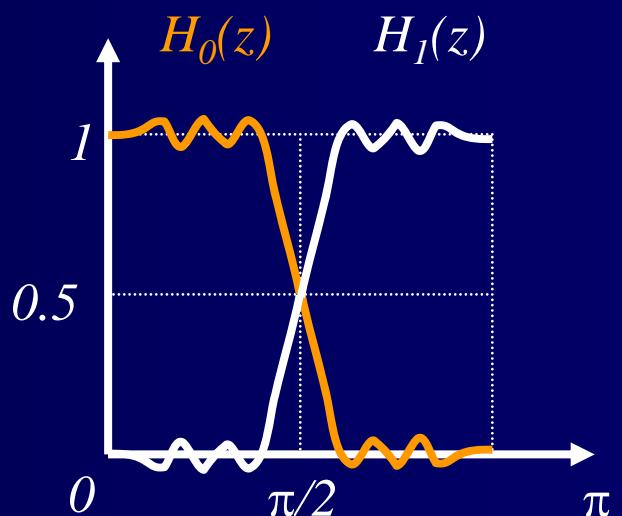
$$H(z) = H(z^{-1})$$

$$H(z) + H(-z^{-1}) = 1$$

■ 周波数特性

- 周波数特性は $\pi/2$ で対称
- 対称なフィルタ特性と加算すると1になる

$$H(e^{j\omega}) + H(-e^{-j\omega}) = H(e^{j\omega}) + H(e^{j(\pi-\omega)}) = 1$$



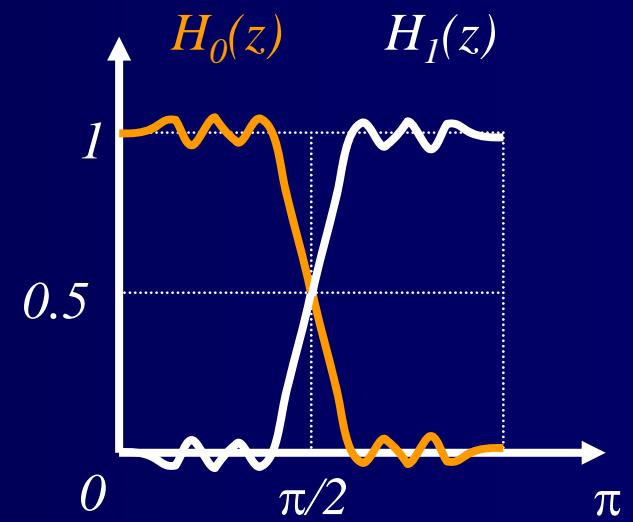
Halfband Filter

- Definition of halfband filter $H(z)$

$$H(z) = H(z^{-1})$$

$$H(z) + H(-z^{-1}) = 1$$

- Frequency response
 - Symmetric at $\pi/2$
 - Sum of two filters response equals to 1



$$H(e^{j\omega}) + H(-e^{-j\omega}) = H(e^{j\omega}) + H(e^{j(\pi-\omega)}) = 1$$

ハーフバンドフィルタ(2)

■ インパルス応答の性質

$$h(n) + (-1)^n h(n) = \delta(n)$$
$$h(n) = \begin{cases} 1/2 & (n = 0) \\ 0 & (n = even, n \neq 0) \end{cases}$$

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} = \cdots + h(-1)z^1 + h(0)z^0 + h(1)z^{-1} + \cdots$$

$$H(z^{-1}) = \sum_{n=-\infty}^{\infty} h(n)z^n = \cdots + h(-1)z^{-1} + h(0)z^0 + h(1)z^1 + \cdots$$

$$\begin{aligned} H(-z^{-1}) &= \sum_{n=-\infty}^{\infty} h(n)(-z)^n = \cdots - h(-2)z^{-2} - h(-1)z^{-1} + h(0)z^0 \\ &\quad - h(1)z^1 + h(2)z^2 \cdots \end{aligned}$$

Halfband Filter(2)

- Characteristic of impulse response

$$h(n) + (-1)^n h(n) = \delta(n)$$
$$h(n) = \begin{cases} 1/2 & (n = 0) \\ 0 & (n = even, n \neq 0) \end{cases}$$

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} = \cdots + h(-1)z^1 + h(0)z^0 + h(1)z^{-1} + \cdots$$

$$H(z^{-1}) = \sum_{n=-\infty}^{\infty} h(n)z^n = \cdots + h(-1)z^{-1} + h(0)z^0 + h(1)z^1 + \cdots$$

$$\begin{aligned} H(-z^{-1}) &= \sum_{n=-\infty}^{\infty} h(n)(-z)^n = \cdots - h(-2)z^{-2} - h(-1)z^{-1} + h(0)z^0 \\ &\quad - h(1)z^1 + h(2)z^2 \cdots \end{aligned}$$

ミラーフィルタ

■ ミラーフィルタの定義

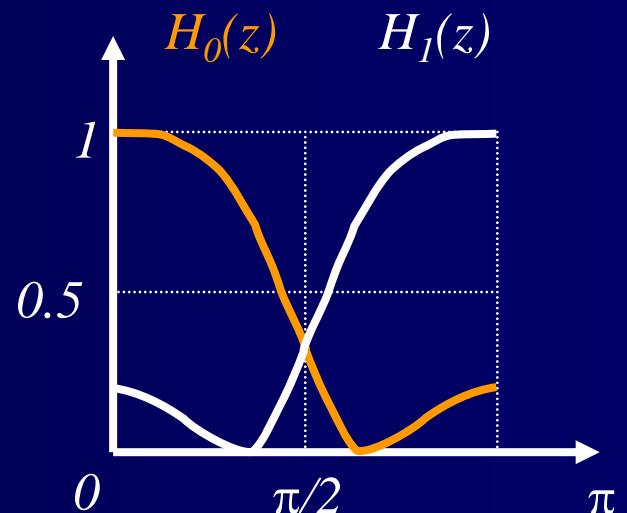
$$H_1(z) = H_0(-z)$$

$$H_1(e^{j\omega}) = H_0(e^{j(\omega-\pi)})$$

$$h_1(n) = (-1)^n h_0(n)$$

■ 周波数特性

- 周波数特性は $\pi/2$ で対称
- 対称なフィルタ特性と加算すると1にならない



Mirror Filter

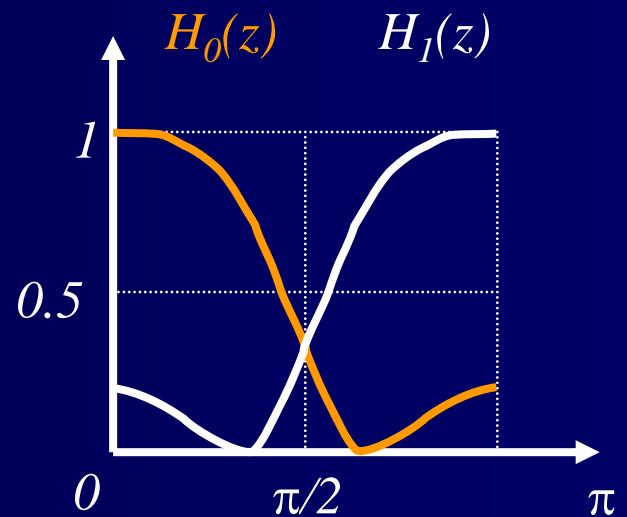
- Definition of mirror filter

$$H_1(z) = H_0(-z)$$

$$H_1(e^{j\omega}) = H_0(e^{j(\omega-\pi)})$$

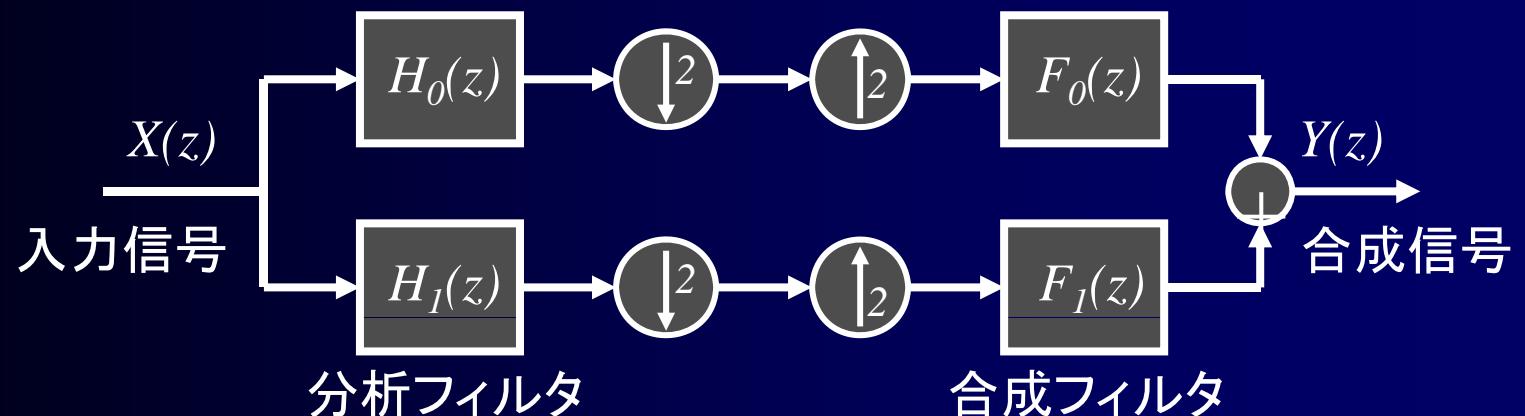
$$h_1(n) = (-1)^n h_0(n)$$

- Frequency response
 - Symmetric at $\pi/2$
 - Sum of two filters response does not equals to 1



分析・合成

■ 分析フィルタ部と合成フィルタ部

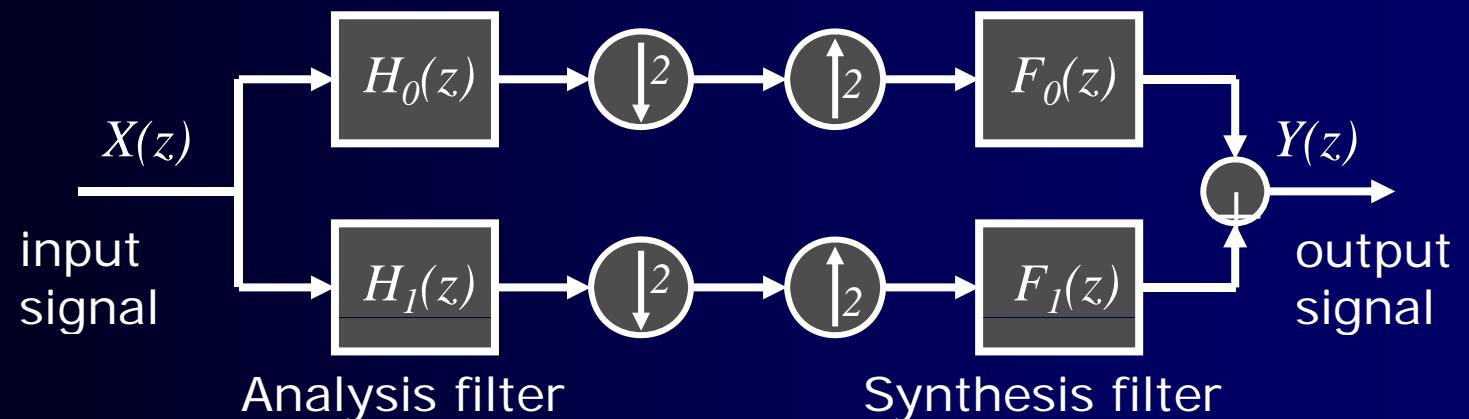


■ 完全再構成の条件

$$\begin{aligned} H_0(-z)F_0(z) + H_1(-z)F_1(z) &= 0 \\ H_0(z)F_0(z) + H_1(z)F_1(z) &= 2z^{-L} \end{aligned}$$

Analysis and Synthesis

- Analysis filters and Synthesis filters



- Condition for Perfect Reconstruction

$$\begin{array}{l} H_0(-z)F_0(z) + H_1(-z)F_1(z) = 0 \\ \hline H_0(z)F_0(z) + H_1(z)F_1(z) = 2z^{-L} \end{array}$$

2バンドフィルタ

- QMF (Quadrature Mirror Filter)
 - 完全再構成ではない直線位相フィルタ
 - 周波数分離特性に優れる

$$\begin{aligned}H_1(z) &= H_0(-z) \\F_0(z) &= H_1(-z) \quad F_1(z) = -H_0(-z)\end{aligned}$$

- CQF (Conjugate Quadrature Filter)
 - 完全再構成の非直線位相フィルタ(偶数次係数0のHalfband Filter)

$$\begin{aligned}H_1(z) &= z^{-(N-1)} H_0(-z^{-1}) \\F_0(z) &= H_1(-z) \quad F_1(z) = H_0(-z)\end{aligned}$$

2-band filter

- QMF (Quadrature Mirror Filter)
 - Linear phase filter, but not Perfect Reconstruction
 - Better band splitting

$$\begin{aligned} H_1(z) &= H_0(-z) \\ F_0(z) &= H_1(-z) \quad F_1(z) = -H_0(-z) \end{aligned}$$

- CQF (Conjugate Quadrature Filter)
 - Non linear phase, Perfect Reconstruction (even order coefficients are 0 in Halfband Filter)

$$\begin{aligned} H_1(z) &= z^{-(N-1)} H_0(-z^{-1}) \\ F_0(z) &= H_1(-z) \quad F_1(z) = H_0(-z) \end{aligned}$$

2バンドフィルタ(2)

- SSKF (Symmetric Short Kernel Filter)
 - 直線位相の非直交フィルタ

$$F_0(z) = H_1(-z)$$
$$F_1(z) = -H_0(-z)$$

- SSKF(5,3)フィルタの係数の例

	n	係数値
$H_0(z)$	0,4	-0.125
	1,3	0.25
	2	0.75
$H_1(z)$	0,2	0.5
	1	-1.0

2-band filter(2)

- SSKF (Symmetric Short Kernel Filter)
 - Linear phase filter, not orthogonal

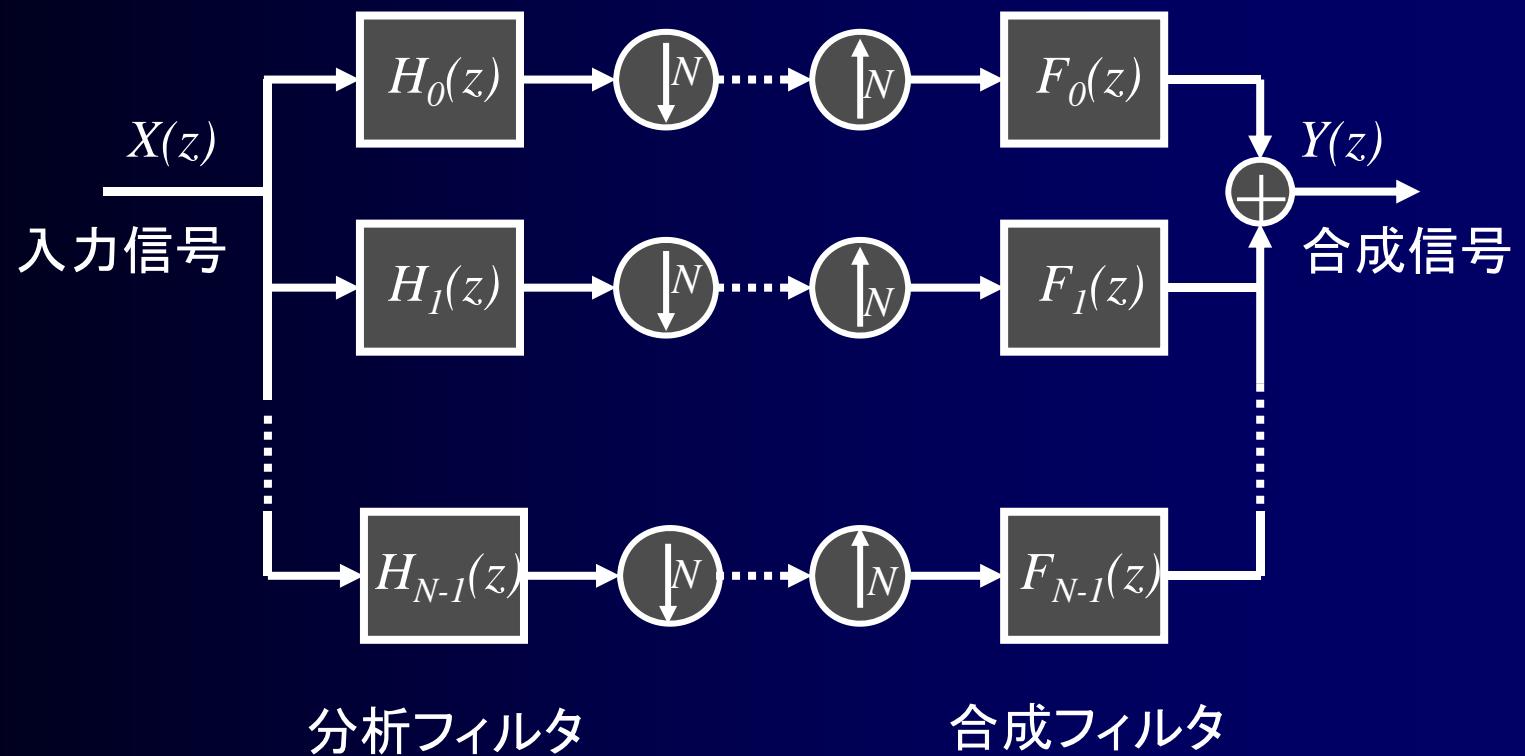
$$F_0(z) = H_1(-z)$$
$$F_1(z) = -H_0(-z)$$

- Example of SSKF(5,3) filter

	n	Coeff.
$H_0(z)$	0,4	-0.125
	1,3	0.25
	2	0.75
$H_1(z)$	0,2	0.5
	1	-1.0

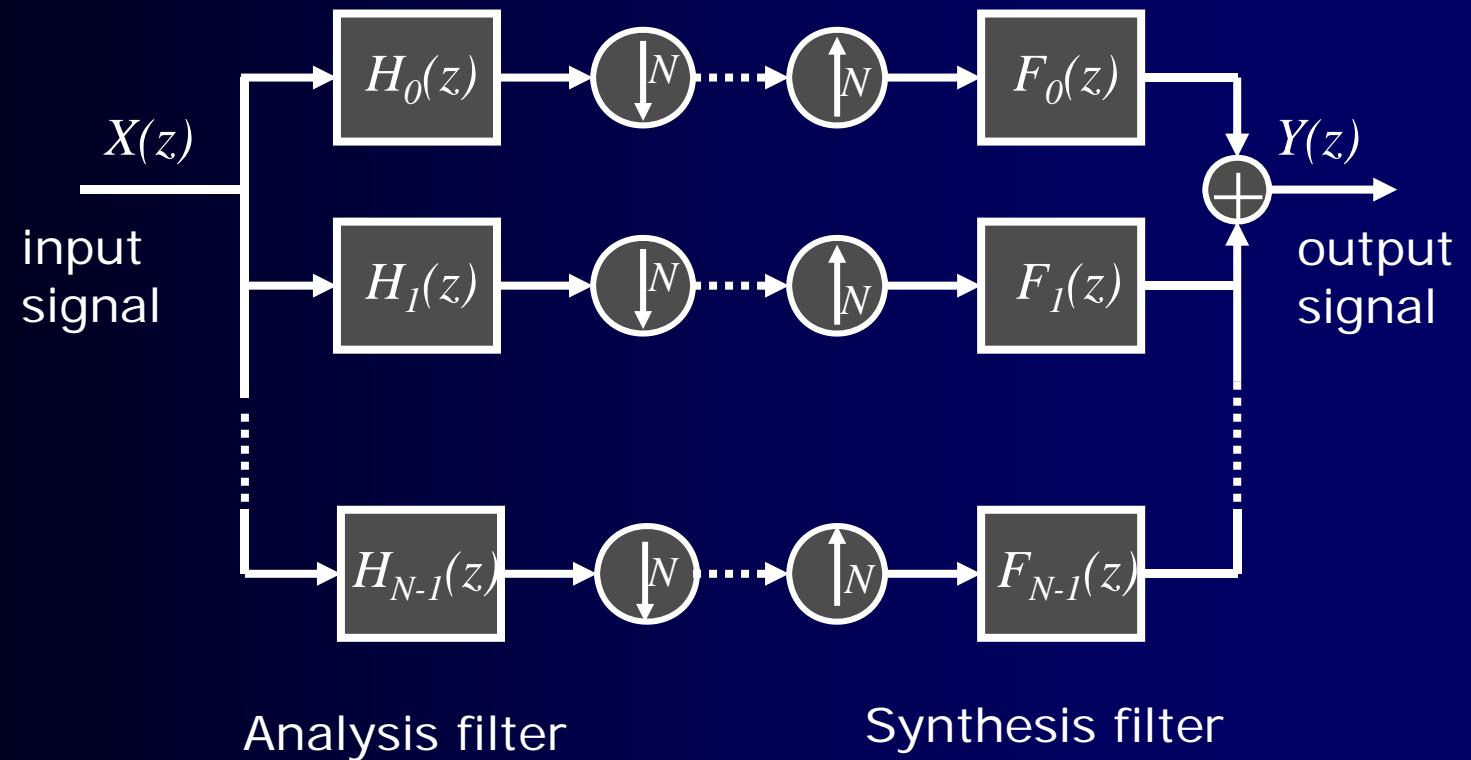
Nバンドフィルタ

■ Nバンドフィルタバンクの構成



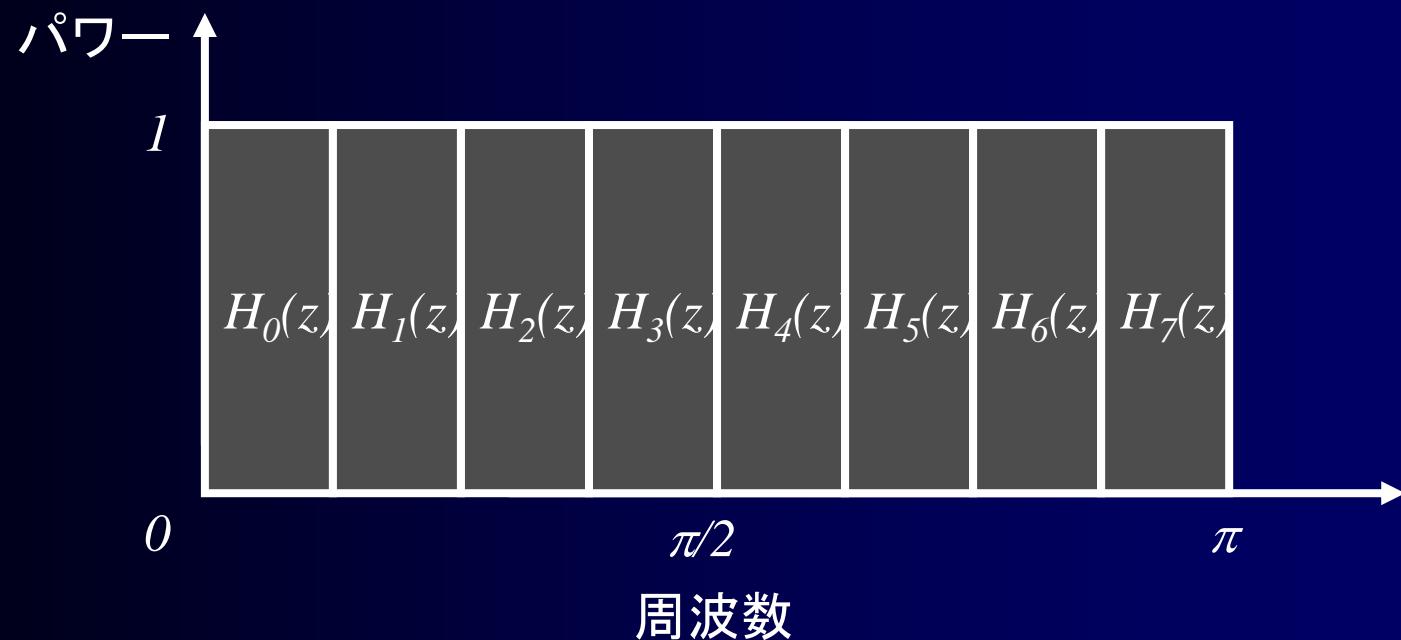
N-band filter

■ Structure of N-band filterbank



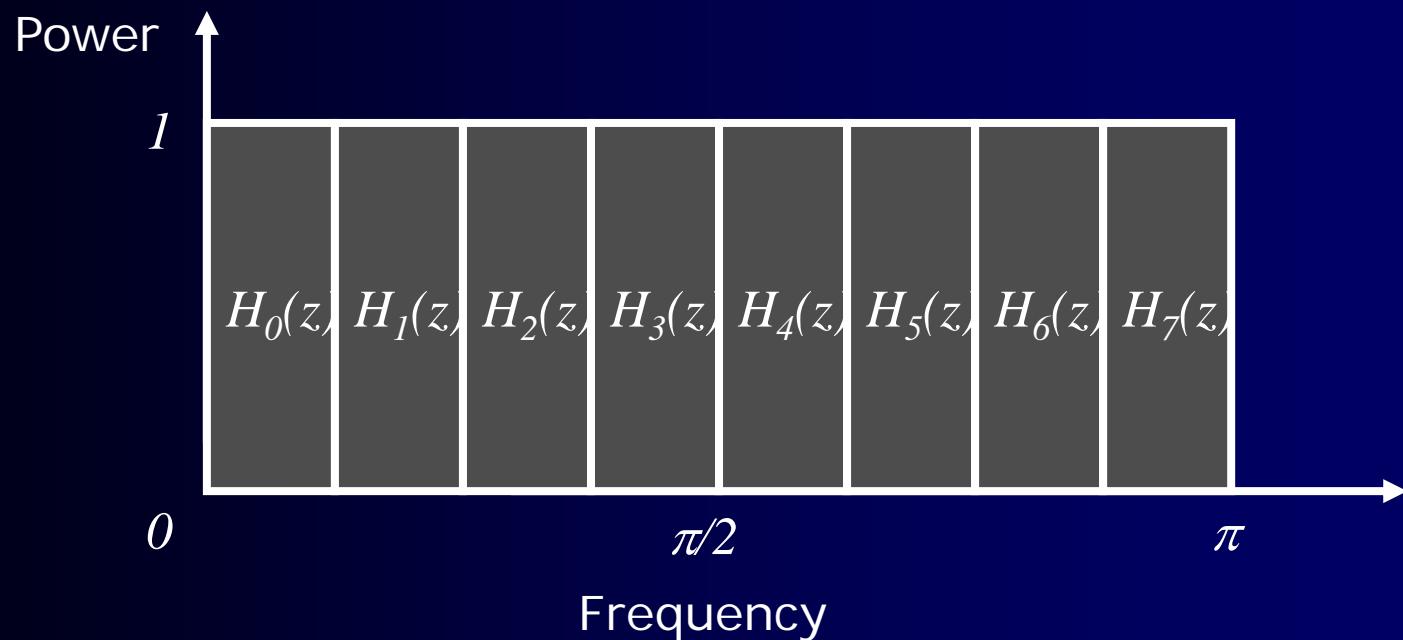
Nバンドフィルタ(2)

■ 理想フィルタバンクの周波数特性(8band)



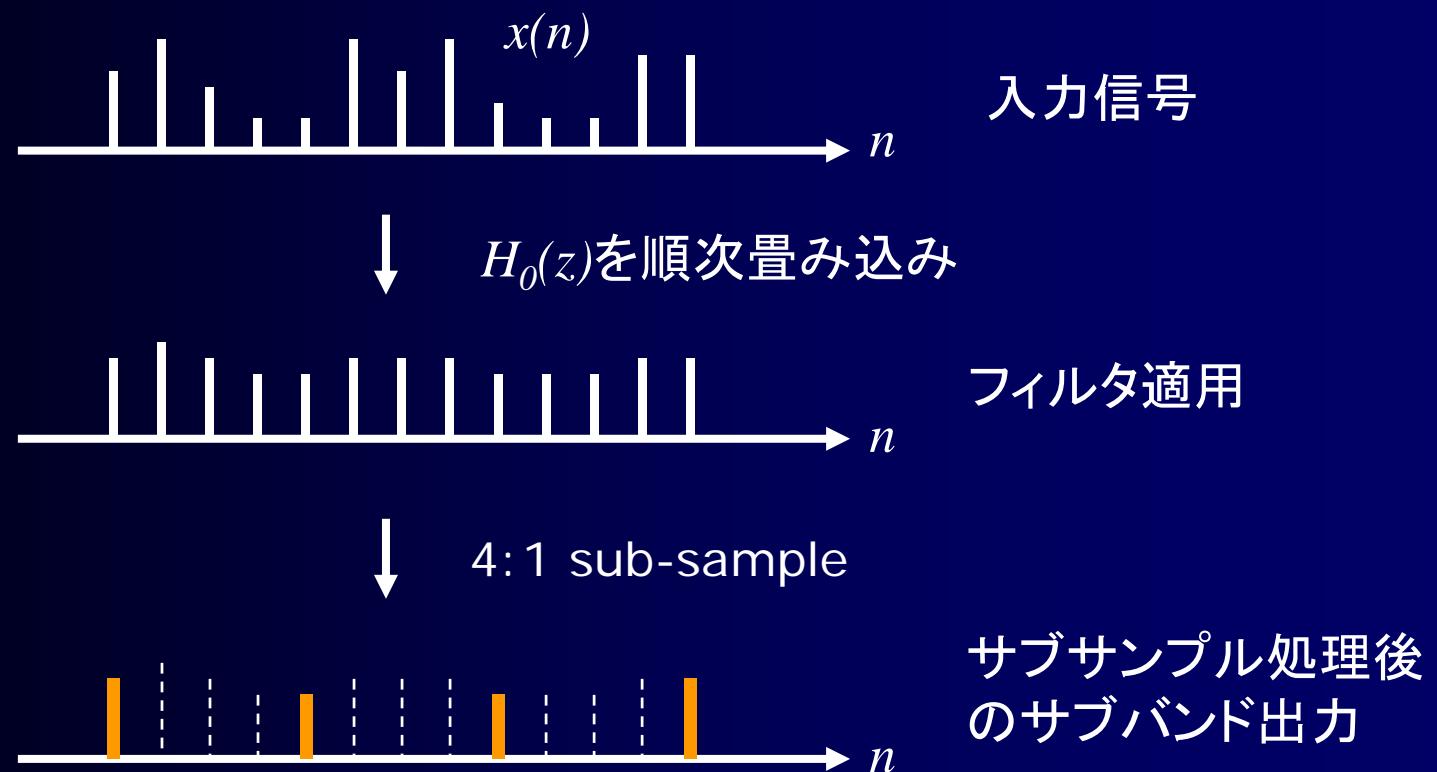
N-band filter(2)

- Ideal frequency response of filterbank (8-band)



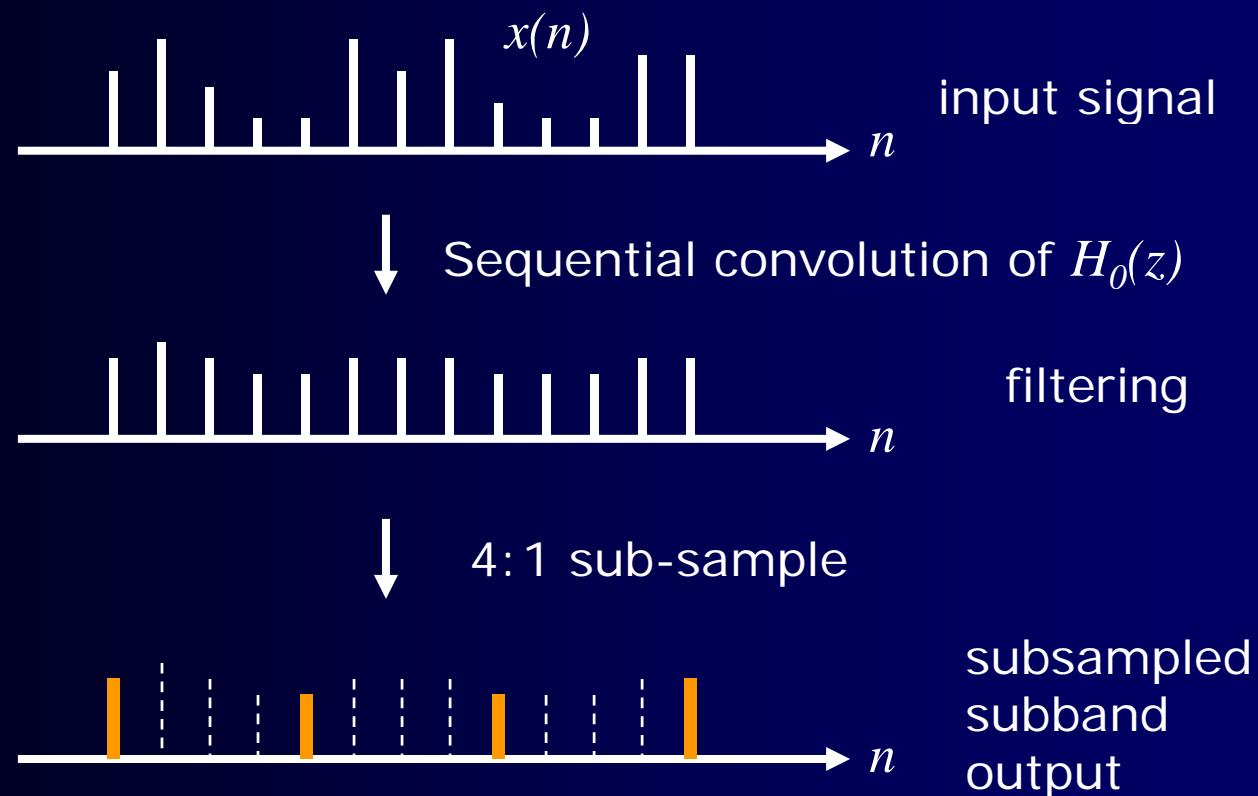
Nバンドフィルタ(3)

- 4バンドの場合のサブバンド出力



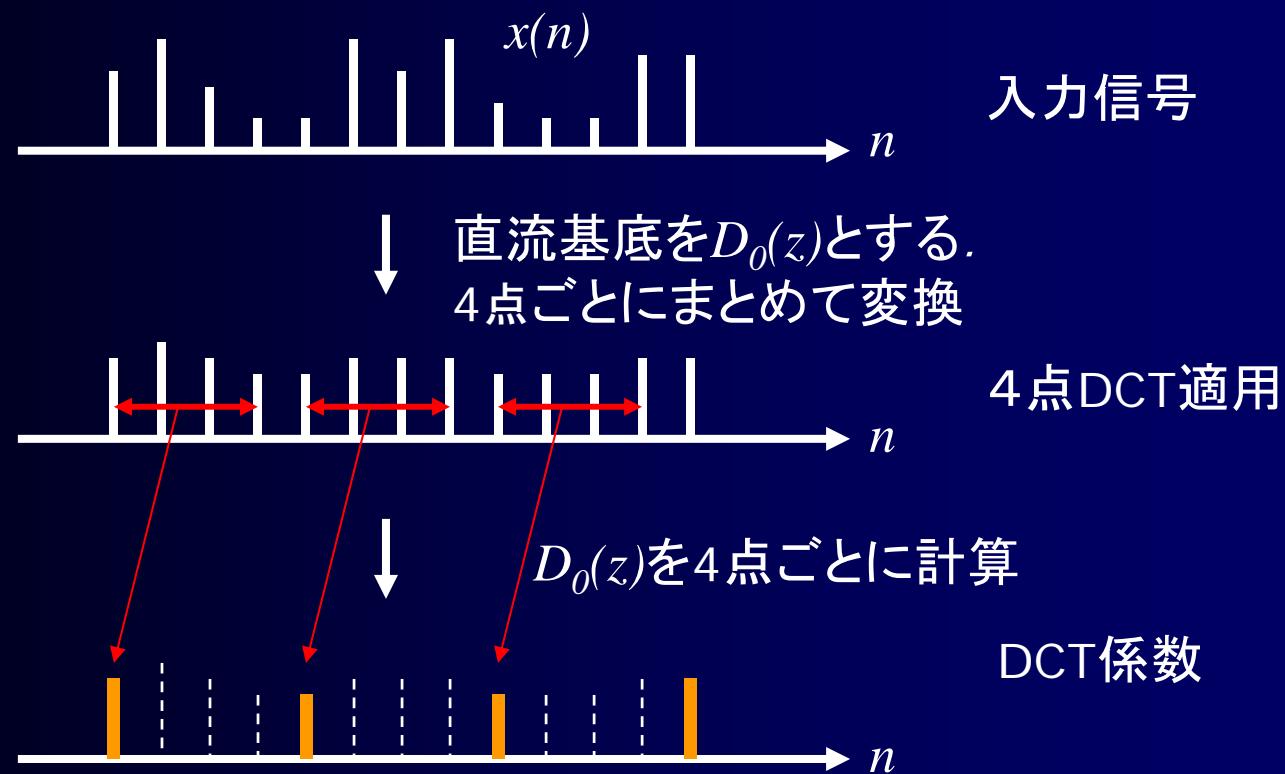
N-band filter(3)

■ Subband output at 4-band



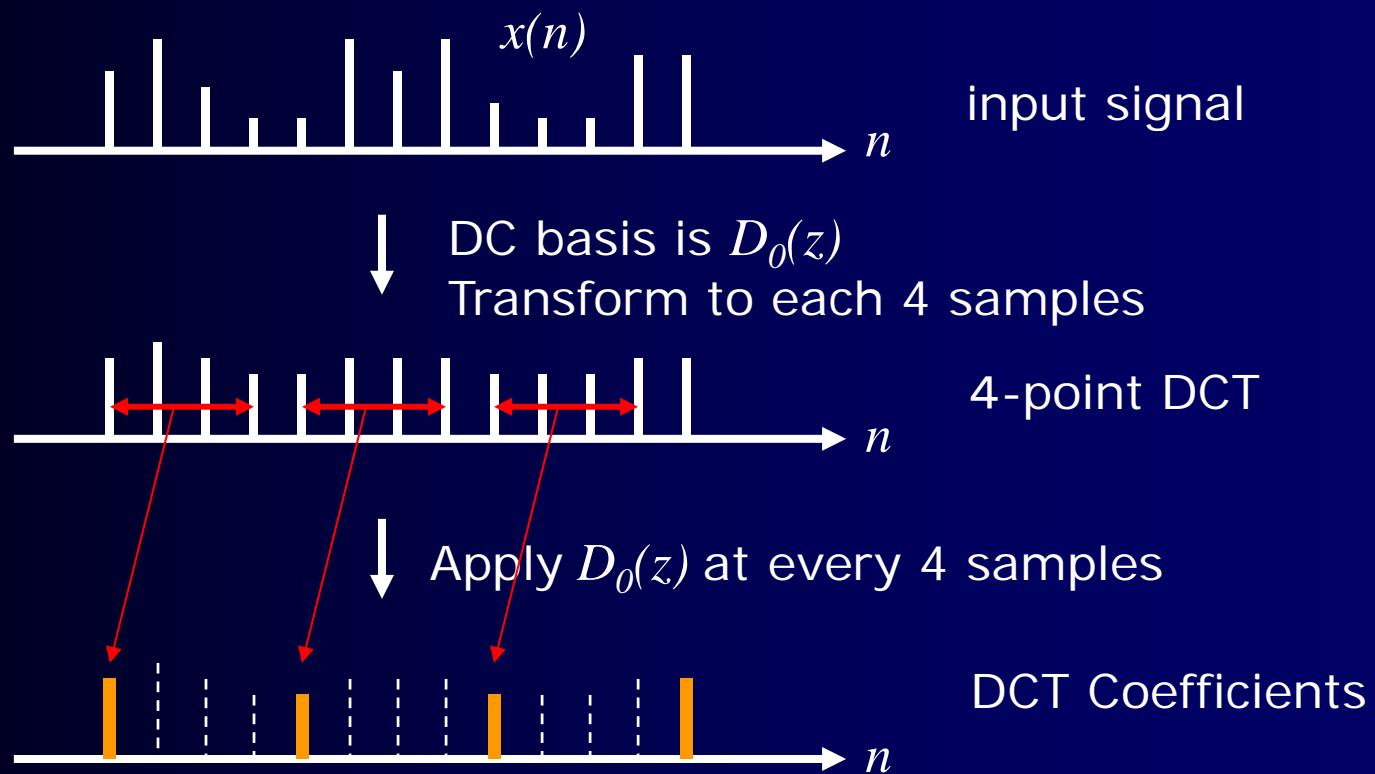
Nバンドフィルタ(4)

■ 4点DCTの場合の係数



N-band filter(4)

- Coefficient of 4-point DCT



画像圧縮技術

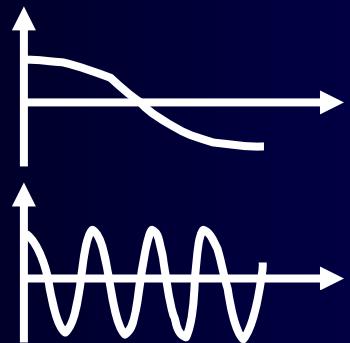
- 色信号多重分離
- 直交変換
- サブバンド
- ウェーブレット
- 係数量子化
- エントロピー符号化

Image Compression Technology

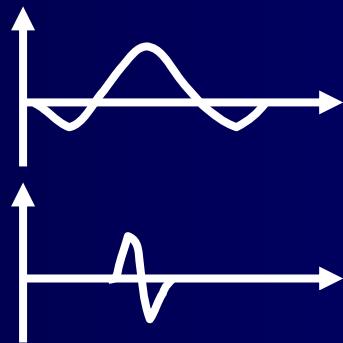
- Chrominance Signal Mux/DeMux
- Orthogonal Transform
- Subband
- Wavelet
- Coefficient Quantization
- Entropy Coding

ウェーブレット

- ウェーブレットとは、基本フィルタの周波数解像度と時間解像度を変化されたフィルタ群を用いて、信号を表現する手法
- DFT(離散フーリエ変換)は長時間の信号を観察し、高い周波数解像度により信号を分析しようとする手法
- サブバンド分析は、一定の時間解像度および一定の周波数解像度を持つフィルタ群を用いて、信号を表現する手法



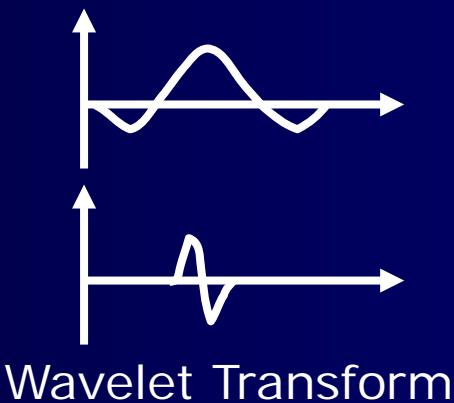
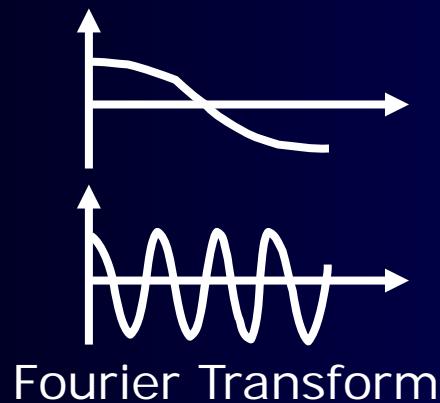
フーリエ変換



ウェーブレット変換

Wavelet

- Wavelet is a technique to describe signal using a set of filters having varied frequency and time axis resolutions based on a basic filter
- DFT(Discrete Fourier Transform) is a technique to analyze signal using maximum frequency resolution in a long term
- Subband analysis is a technique to describe signal using filters having certain frequency and time axis resolution



ウェーブレット(2)

■ 連続系のウェーブレット信号

- dilation: スケールを変えるパラメータ a による操作
- translation: シフトパラメータ b による操作

$$x_{a,b}(t) = \frac{1}{\sqrt{a}} x\left(\frac{t-b}{a}\right)$$

■ ウェーブレットのフーリエ変換

$$\overline{X}_{a,b}(\omega) = \sqrt{a} X(a\omega) \exp(-j\omega b)$$

Wavelet(2)

- Continuous Wavelet signal
 - dilation: operation by scaling parameter a
 - translation: operation by shift parameter b

$$x_{a,b}(t) = \frac{1}{\sqrt{a}} x\left(\frac{t-b}{a}\right)$$

- Fourier transform of Wavelet

$$X_{a,b}(\omega) = \sqrt{a} X(a\omega) \exp(-j\omega b)$$

復習

■ フーリエ変換

$$X(\omega) \equiv \int_{-\infty}^{\infty} x(t) \exp(-j\omega t) dt$$

■ 信号の拡大・縮小に対するフーリエ変換

$$\begin{aligned} \int_{-\infty}^{\infty} x\left(\frac{t}{a}\right) \exp(-j\omega t) dt &= \int_{-\infty}^{\infty} x(s) \exp(-j\omega as) ads \\ &= a \int_{-\infty}^{\infty} x(s) \exp(-j(\omega a)s) ds \\ &= aX(a\omega) \end{aligned}$$

$$\begin{aligned} s &\equiv \frac{t}{a} \\ ds &= \frac{1}{a} dt \end{aligned}$$

Review

- Fourier Transform

$$X(\omega) \equiv \int_{-\infty}^{\infty} x(t) \exp(-j\omega t) dt$$

- Fourier Transform for expanded (shrunk) signal

$$\begin{aligned} \int_{-\infty}^{\infty} x\left(\frac{t}{a}\right) \exp(-j\omega t) dt &= \int_{-\infty}^{\infty} x(s) \exp(-j\omega as) ads \\ &= a \int_{-\infty}^{\infty} x(s) \exp(-j(\omega a)s) ds \\ &= aX(a\omega) \end{aligned}$$

$$\begin{aligned} s &\equiv \frac{t}{a} \\ ds &= \frac{1}{a} dt \end{aligned}$$

復習(2)

■ 信号のシフトに対するフーリエ変換

$$\begin{aligned} \int_{-\infty}^{\infty} x(t-b) \exp(-j\omega t) dt &= \int_{-\infty}^{\infty} x(s) \exp(-j\omega(s+b)) ds \\ &= \int_{-\infty}^{\infty} x(s) \exp(-j\omega s) \exp(-j\omega b) ds \\ &= \exp(-j\omega b) \int_{-\infty}^{\infty} x(s) \exp(-j\omega s) ds \\ &= \exp(-j\omega b) X(\omega) \end{aligned}$$

$$\begin{aligned} s &\equiv t - b \\ ds &= dt \end{aligned}$$

Review(2)

■ Fourier Transform for shifted signal

$$\begin{aligned}\int_{-\infty}^{\infty} x(t-b) \exp(-j\omega t) dt &= \int_{-\infty}^{\infty} x(s) \exp(-j\omega(s+b)) ds \\&= \int_{-\infty}^{\infty} x(s) \exp(-j\omega s) \exp(-j\omega b) ds \\&= \exp(-j\omega b) \int_{-\infty}^{\infty} x(s) \exp(-j\omega s) ds \\&= \exp(-j\omega b) X(\omega)\end{aligned}$$

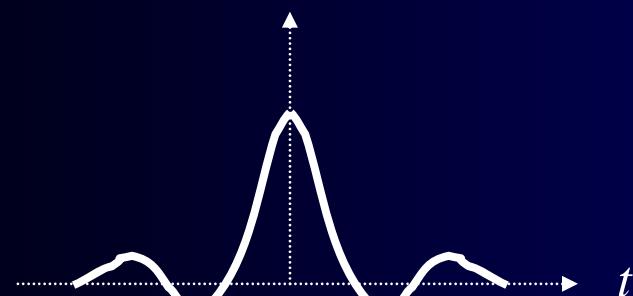
$$s \equiv t - b$$

$$ds = dt$$

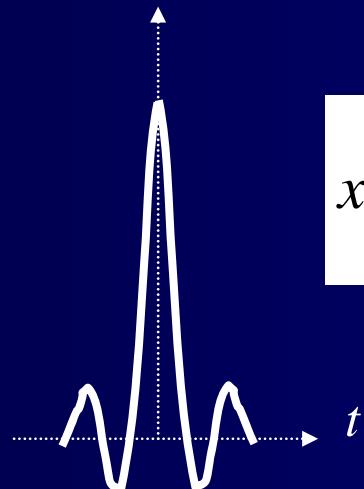
ウェーブレット(3)

- dilationによる信号波形の変化

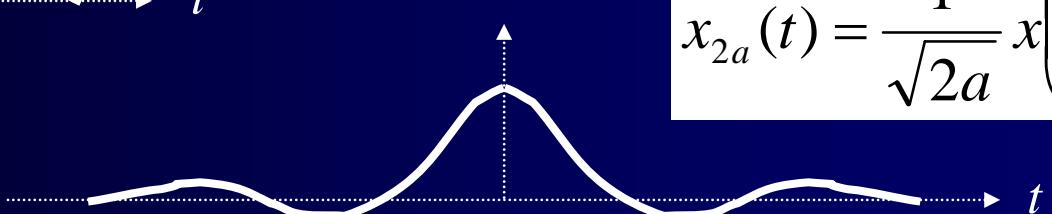
$$x_a(t) = \frac{1}{\sqrt{a}} x\left(\frac{t}{a}\right)$$



$$x_{\frac{a}{2}}(t) = \sqrt{\frac{2}{a}} x\left(\frac{2t}{a}\right)$$



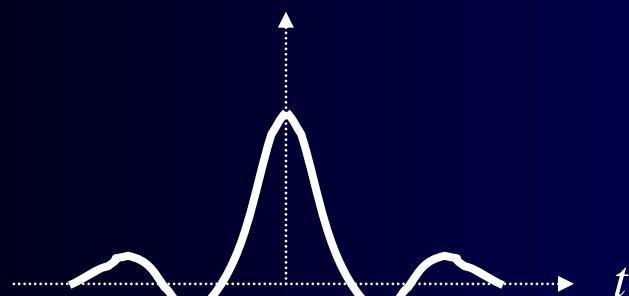
$$x_{2a}(t) = \frac{1}{\sqrt{2a}} x\left(\frac{t}{2a}\right)$$



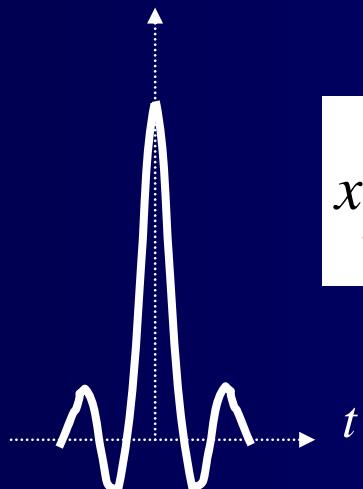
Wavelet(3)

- Signal variation by dilation

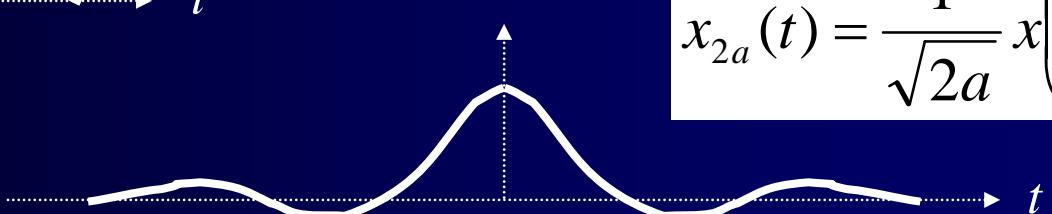
$$x_a(t) = \frac{1}{\sqrt{a}} x\left(\frac{t}{a}\right)$$



$$x_{\frac{a}{2}}(t) = \sqrt{\frac{2}{a}} x\left(\frac{2t}{a}\right)$$



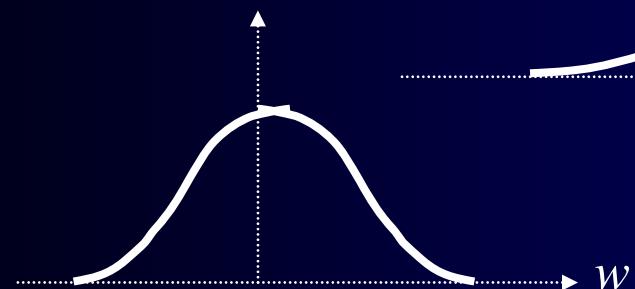
$$x_{2a}(t) = \frac{1}{\sqrt{2a}} x\left(\frac{t}{2a}\right)$$



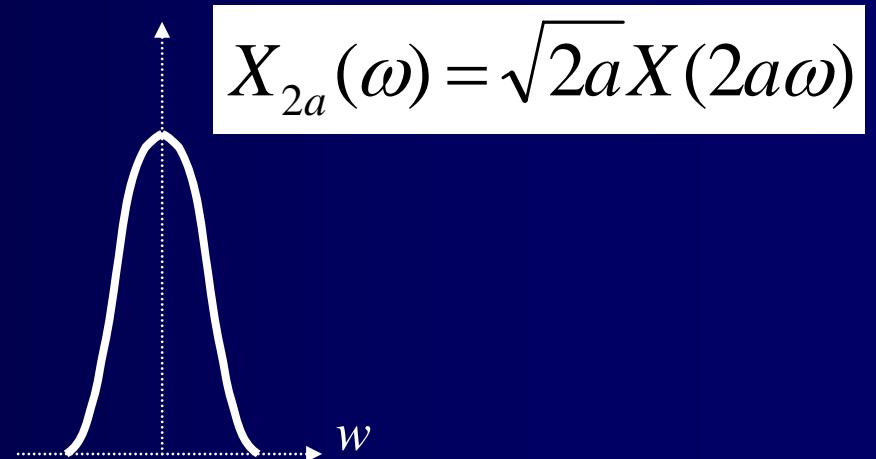
ウェーブレット(4)

- dilationによるスペクトルの変化

$$X_a(\omega) = \sqrt{a} X(a\omega)$$



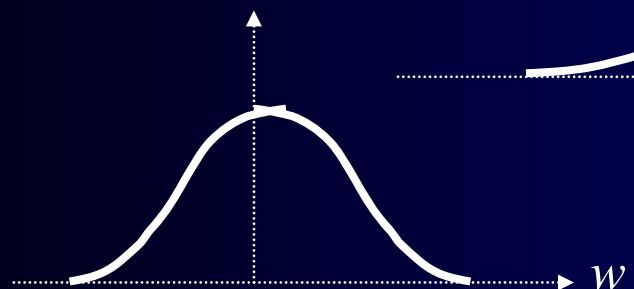
$$X_{\frac{a}{2}}(\omega) = \sqrt{\frac{a}{2}} X\left(\frac{a\omega}{2}\right)$$



Wavelet(4)

- Spectrum variation by dilation

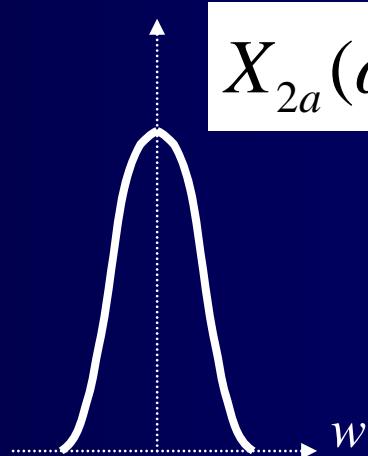
$$X_a(\omega) = \sqrt{a} X(a\omega)$$



$$X_{\frac{a}{2}}(\omega) = \sqrt{\frac{a}{2}} X\left(\frac{a\omega}{2}\right)$$

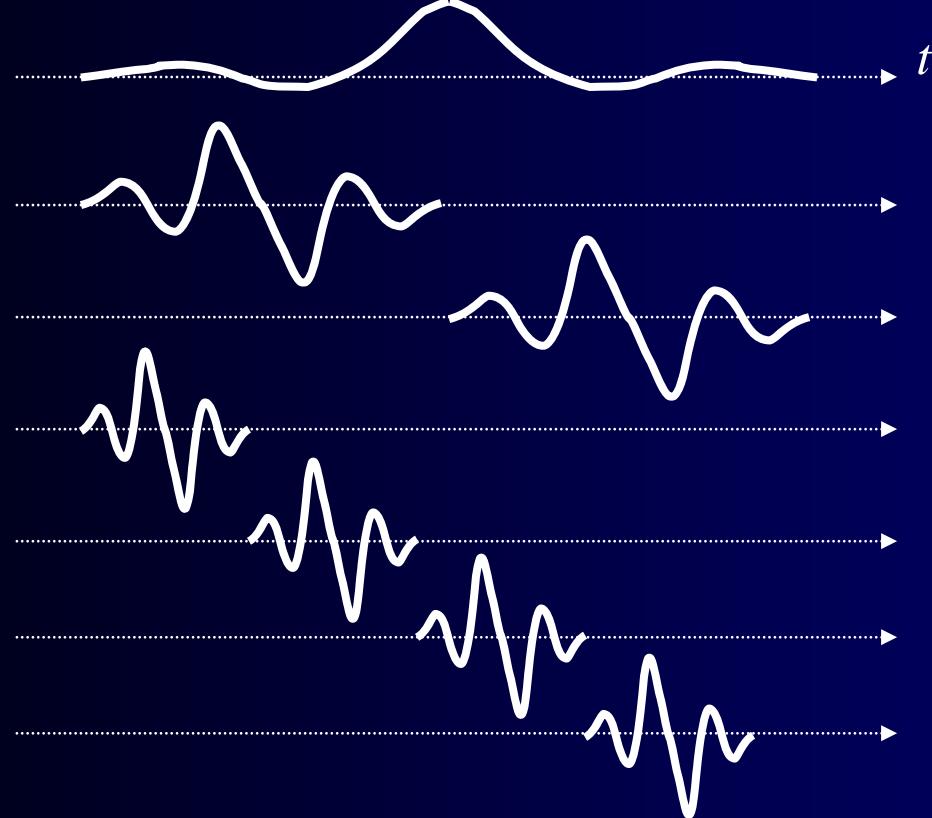


$$X_{2a}(\omega) = \sqrt{2a} X(2a\omega)$$



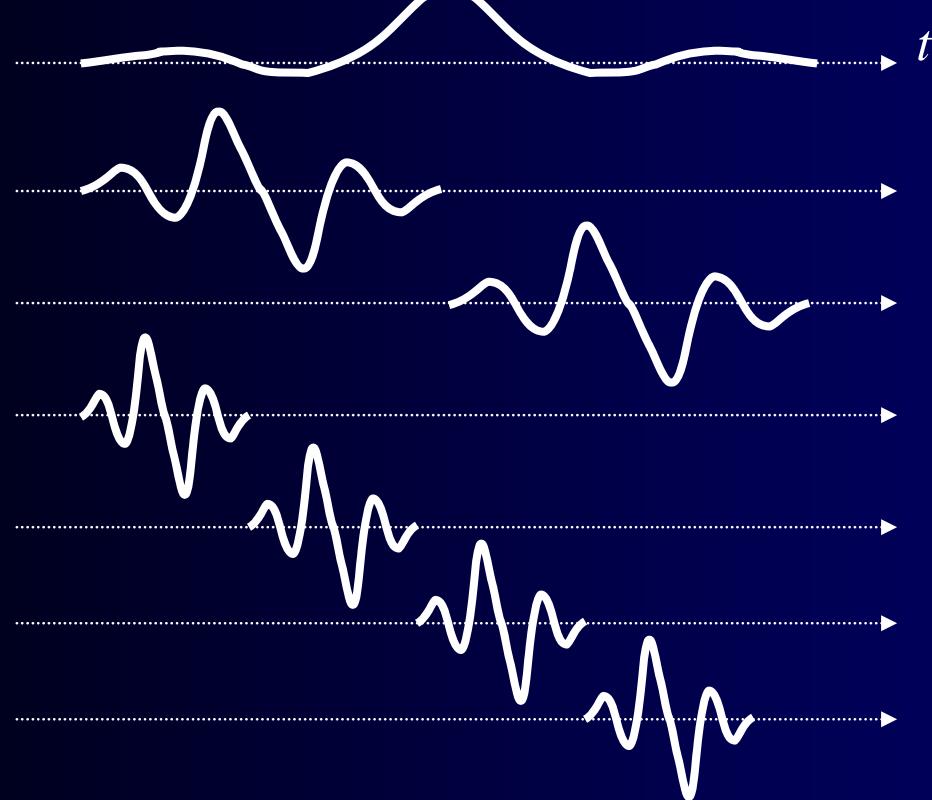
ウェーブレット変換

- Translation パラメータを用いて直交基底を作成



Wavelet Transform

- Create orthogonal basis using Translation parameter



離散ウェーブレット

- dilation と translation のパラメータを離散化

$$a = 2^m, \quad b = n2^m$$

- 离散ウェーブレット基底関数

$$x_{m,n}(t) = 2^{-\frac{m}{2}} x(2^{-m}t - n)$$

- $x(t)$ を正規直交化した関数 $f(t)$ を定義すると、 $H_0(z)$, $H_1(z)$ のインパルス応答 $h_0(n)$, $h_1(n)$ に対して次式が成立つ

$$\phi(t) = 2 \sum_{n=0}^{N-1} h_0(n) \phi(2t - n) \quad \varphi(t) = 2 \sum_{n=0}^{N-1} h_1(n) \phi(2t - n)$$

Discrete Wavelet

- Discretized dilation and translation parameters

$$a = 2^m, \quad b = n2^m$$

- Basis function of discrete wavelet

$$x_{m,n}(t) = 2^{-\frac{m}{2}} x(2^{-m}t - n)$$

- Define function $f(t)$ which is orthogonalized $x(t)$, consider $h_0(n), h_1(n)$ which are impulse responses of $H_0(z), H_1(z)$

$$\phi(t) = 2 \sum_{n=0}^{N-1} h_0(n) \phi(2t - n) \quad \varphi(t) = 2 \sum_{n=0}^{N-1} h_1(n) \phi(2t - n)$$

ハール変換

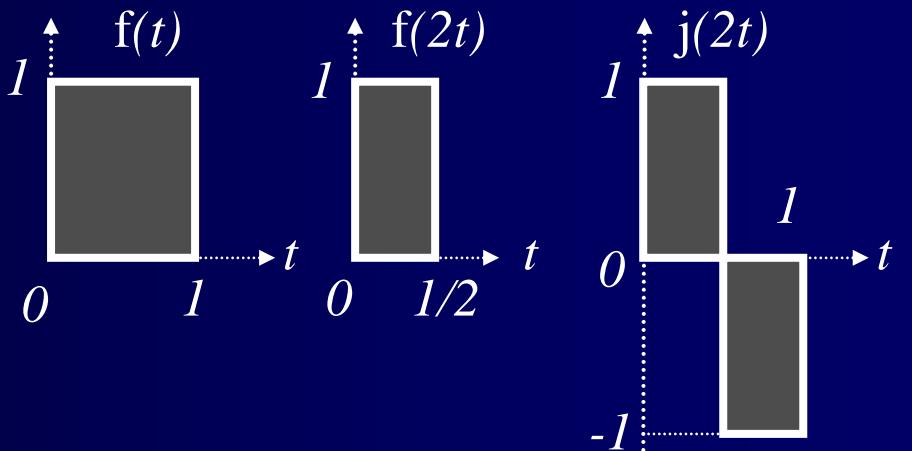
- 最も簡単なウェーブレット
- 低域フィルタ $H_0(z)$, 高域フィルタ $H_1(z)$

$$H_0(z) = \frac{1+z^{-1}}{\sqrt{2}} \quad H_1(z) = \frac{1-z^{-1}}{\sqrt{2}}$$

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & otherwise \end{cases}$$

$$\phi(t) = \phi(2t) + \phi(2t+1)$$

$$\varphi(t) = \phi(2t) - \phi(2t+1)$$

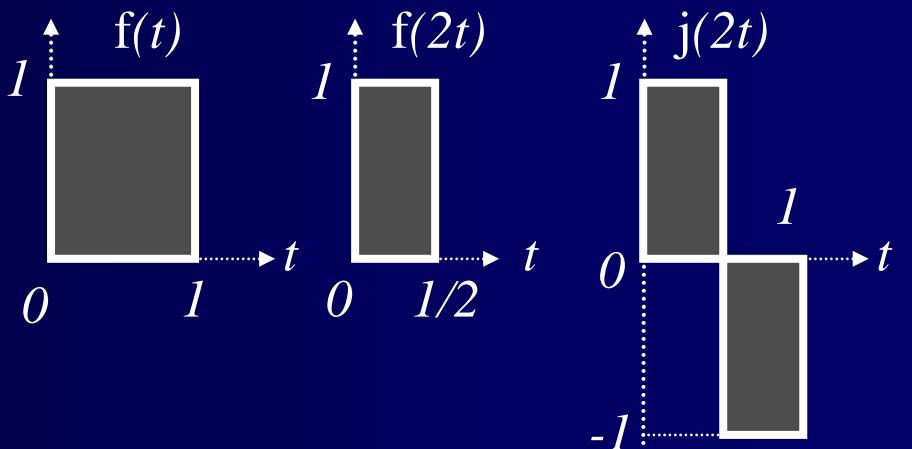


Harr Transform

- The simplest wavelet
- LPF: $H_0(z)$, HPF: $H_1(z)$

$$H_0(z) = \frac{1+z^{-1}}{\sqrt{2}} \quad H_1(z) = \frac{1-z^{-1}}{\sqrt{2}}$$

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & otherwise \end{cases}$$
$$\phi(t) = \phi(2t) + \phi(2t+1)$$
$$\varphi(t) = \phi(2t) - \phi(2t+1)$$



ハール変換(2)

■ 1次元8点ハール変換行列の例

$$\begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(7) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(7) \end{bmatrix}$$

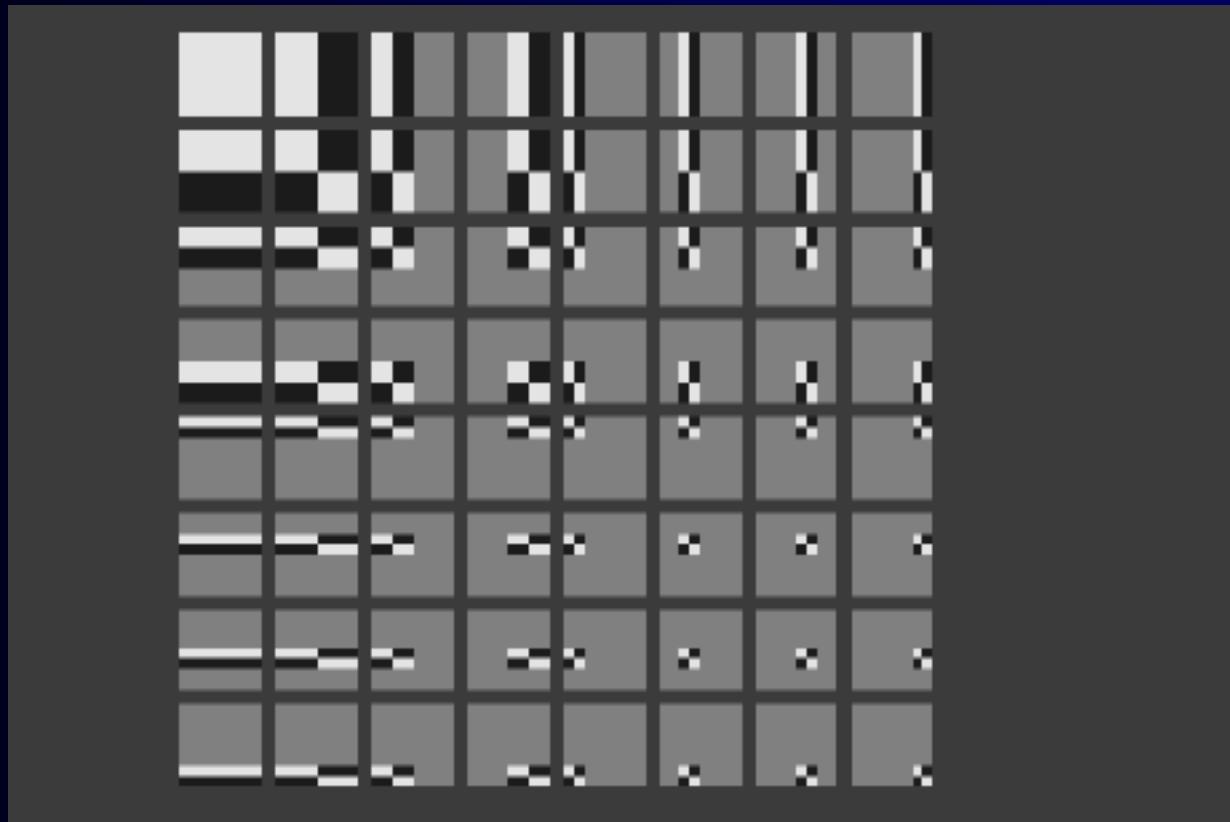
Harr Transform(2)

- 1-D 8-point Harr transform matrix

$$\begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(7) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(7) \end{bmatrix}$$

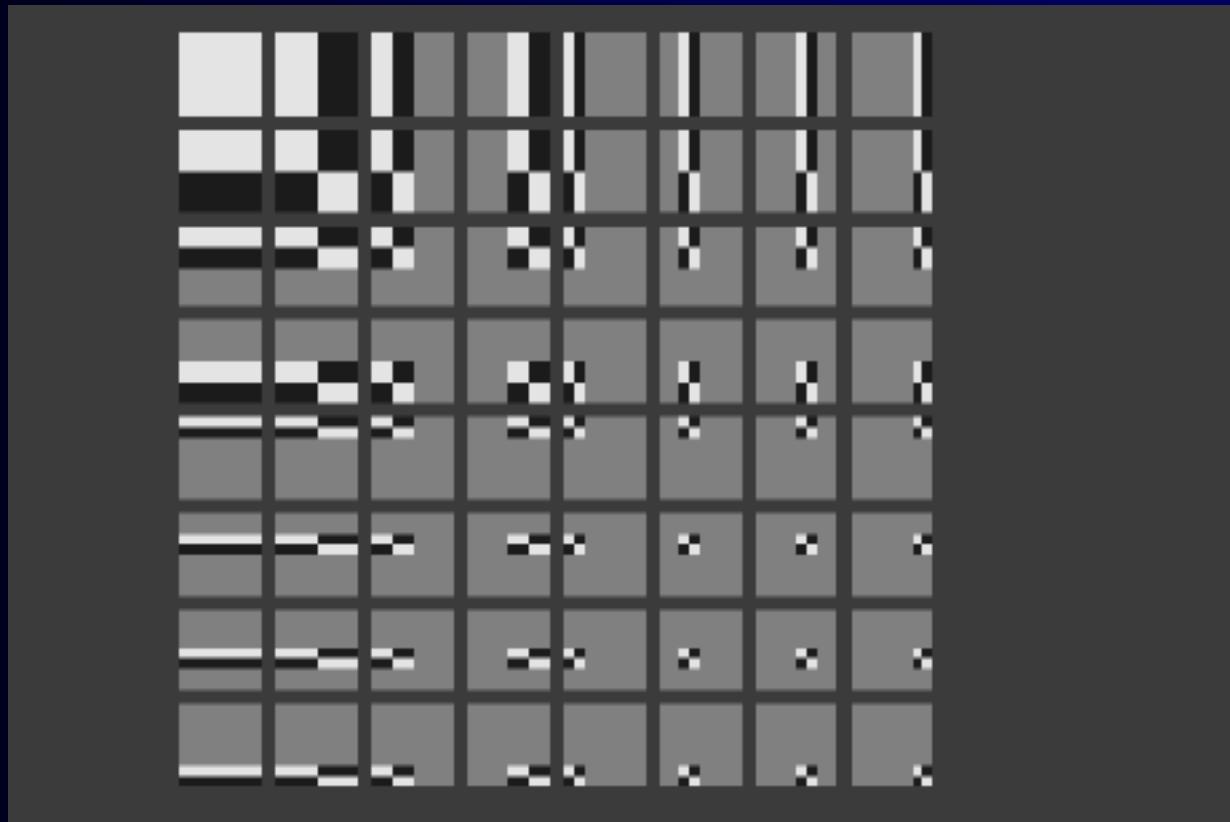
ハール変換基底

- 8x8 ハール変換の基底ベクトル



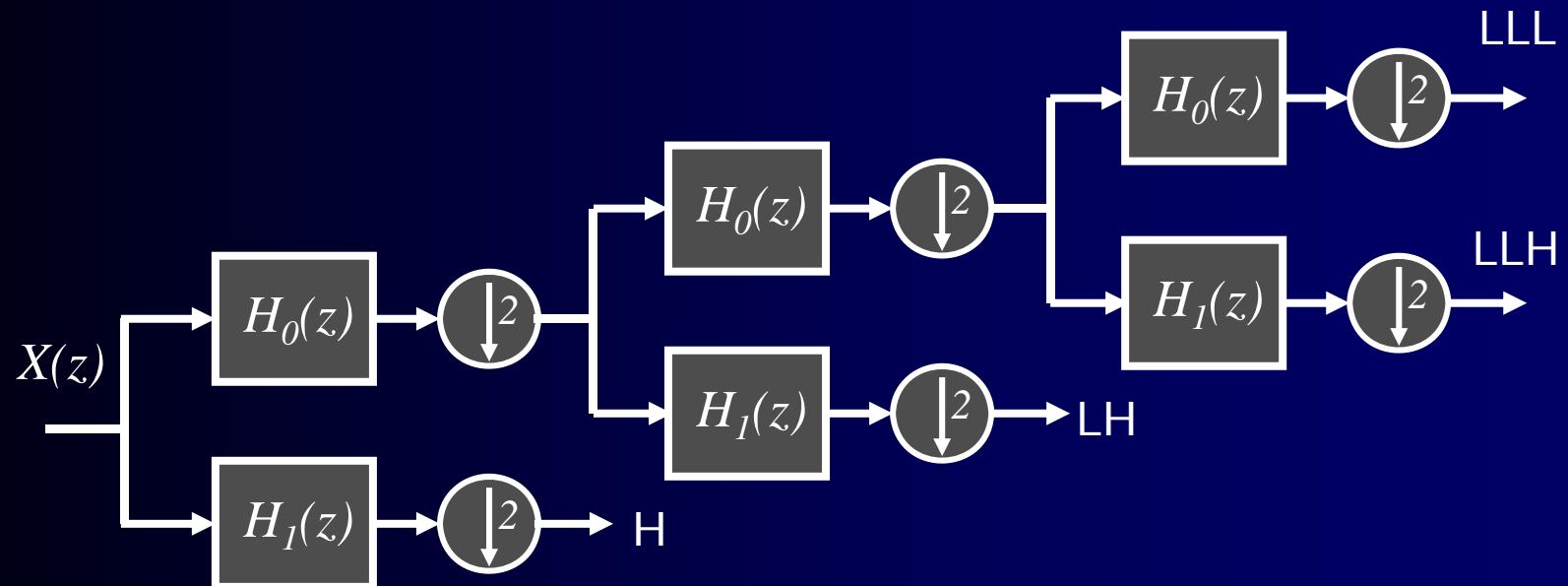
Harr Transform Basis

- 8x8 Harr transform basis



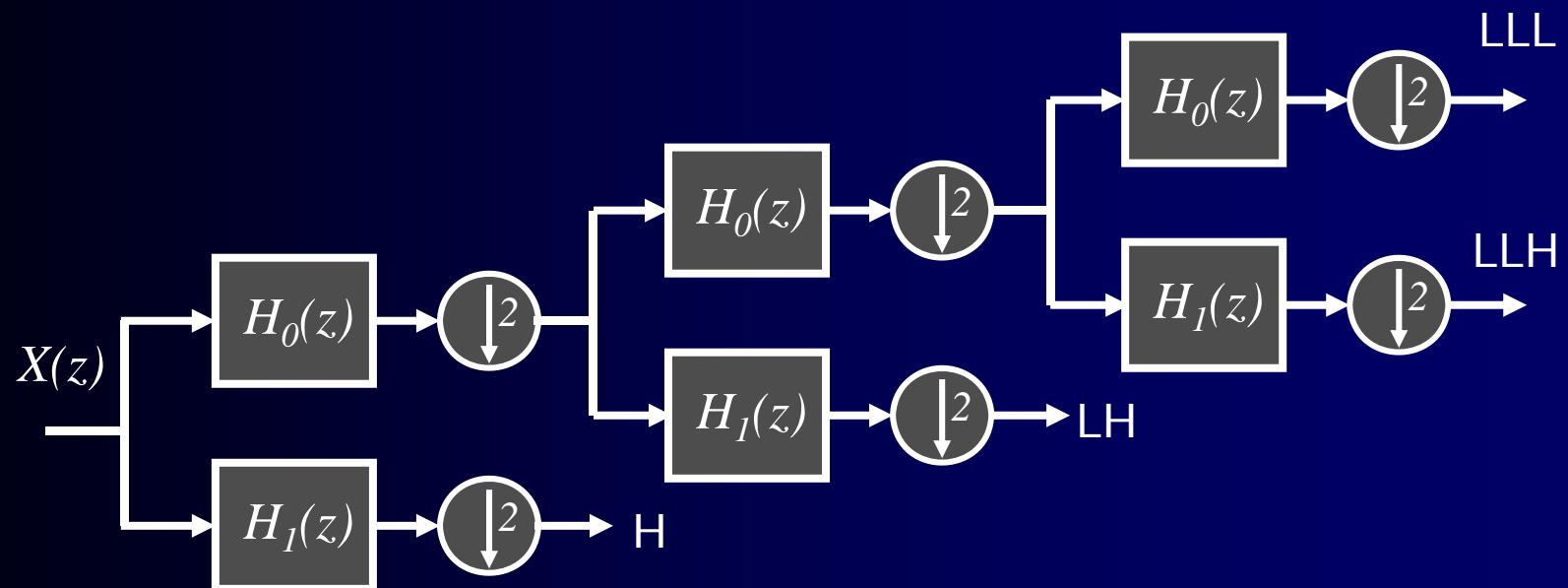
オクターブ分割

■ 4バンドフィルタバンクの構成



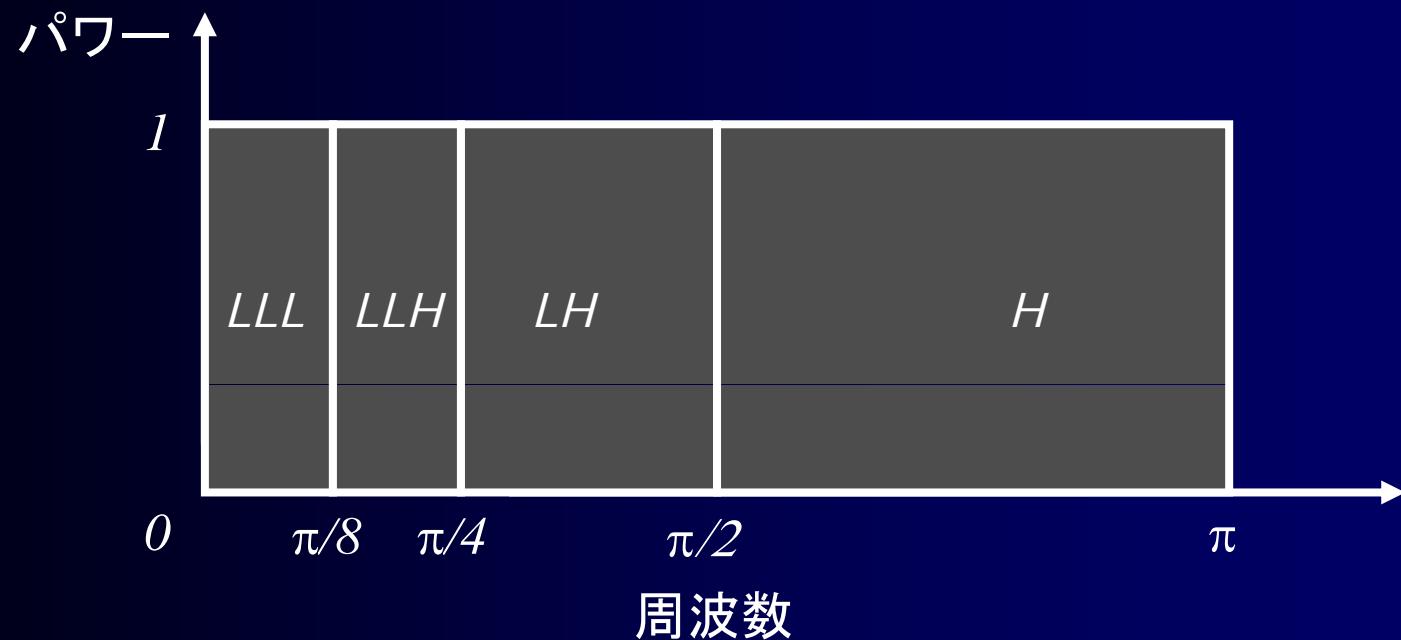
Octave Splitting

- 4-band (diadic) filterbank structure



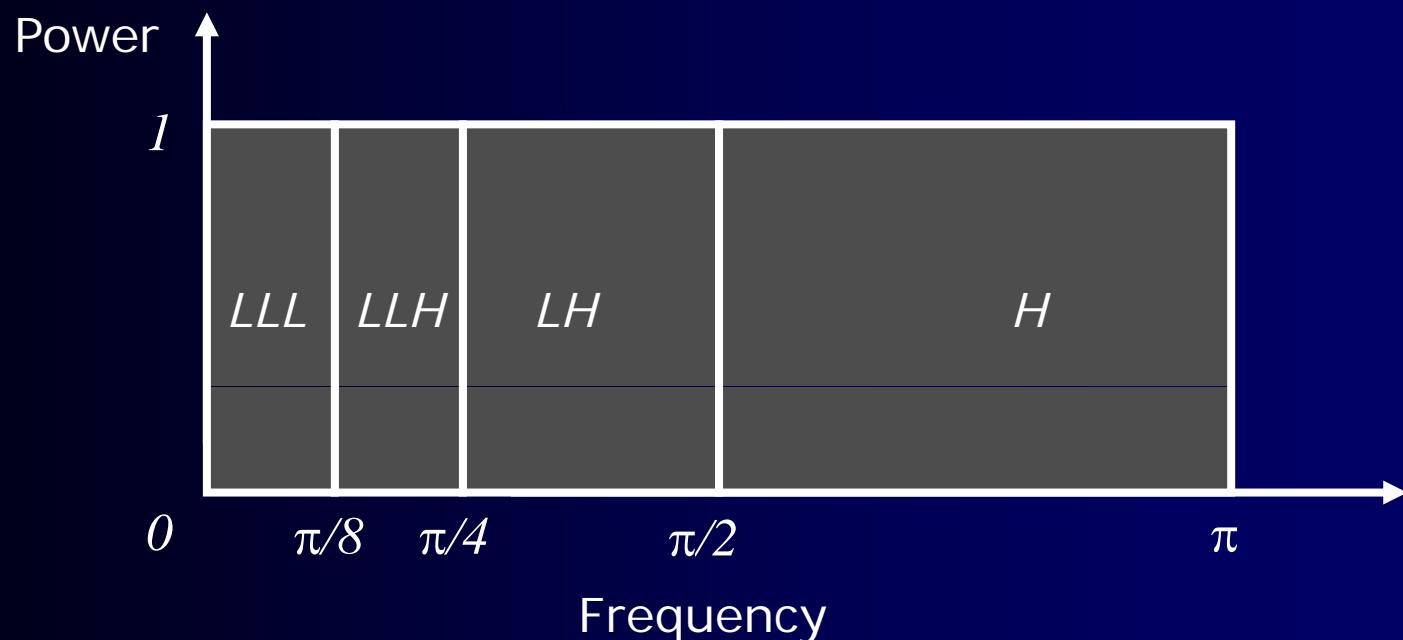
オクターブ分割(2)

- オクターブ分割のフィルタバンクの周波数特性(4band)

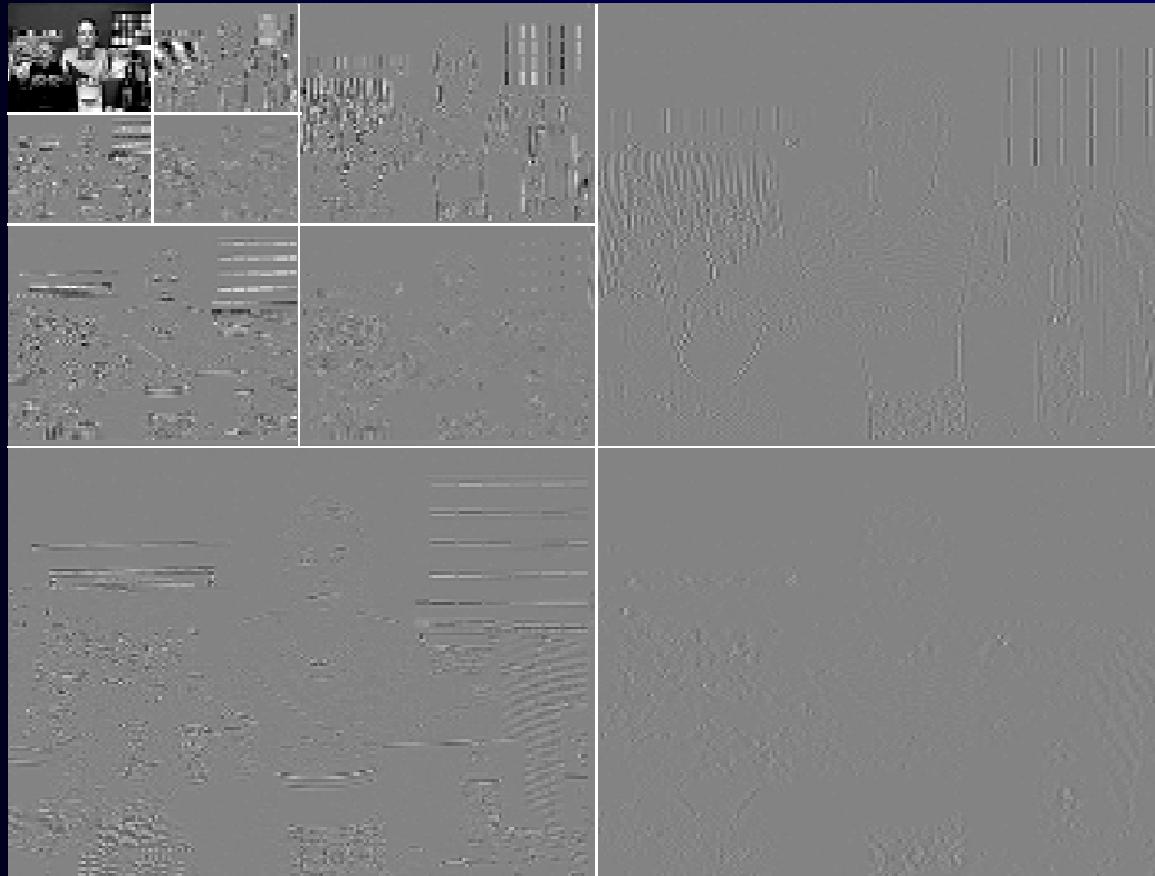


Octave Splitting(2)

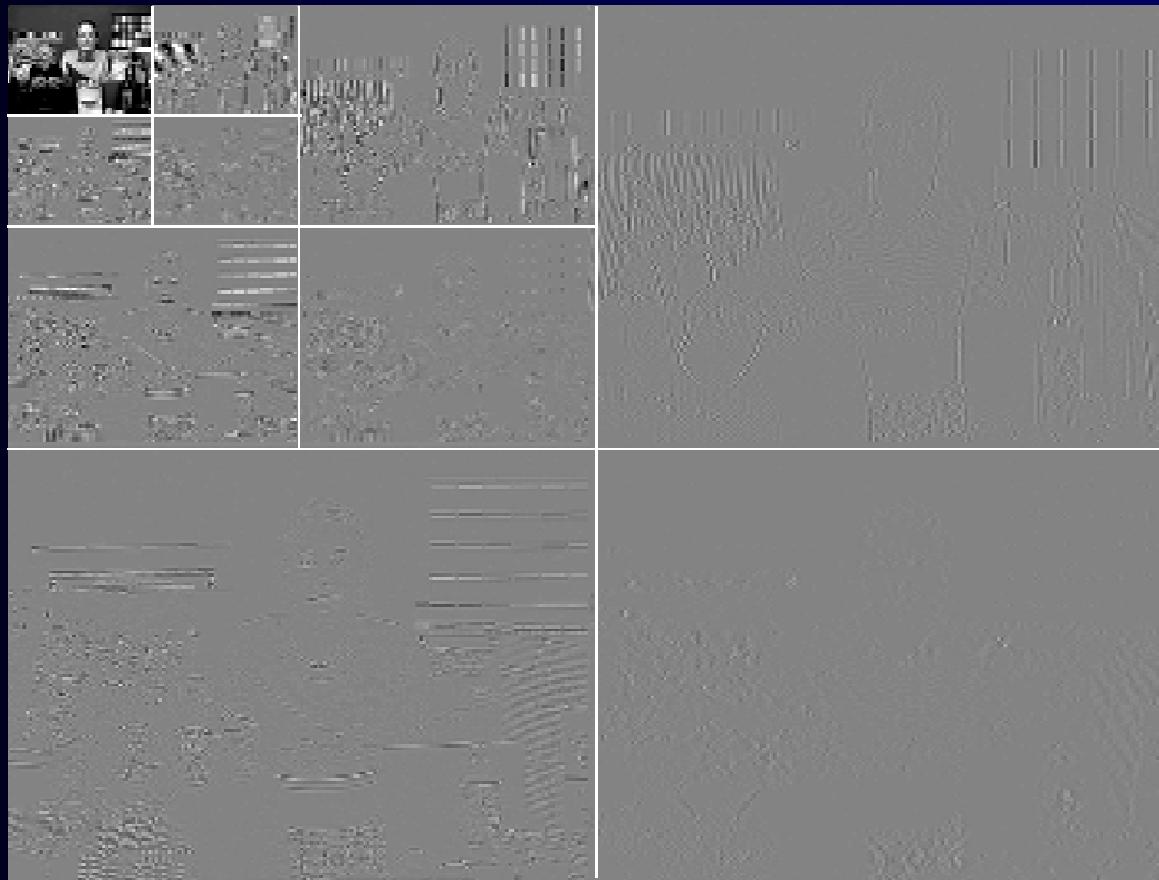
- Frequency responses of ideal octave splitting filterbank (4-band diadic)



ウェーブレット変換画像



Wavelet Transform Image

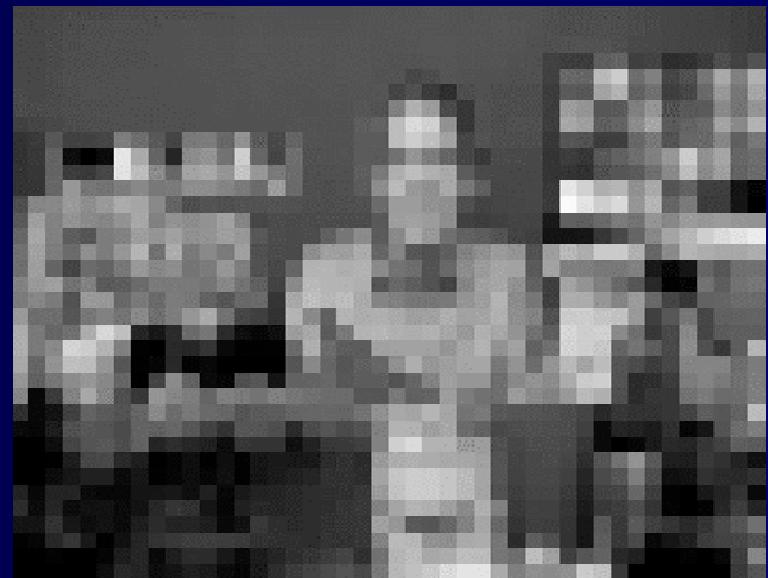


ウェーブレット変換画像(2)

- DWTとDCTの最低周波数成分(1/64)



DWT



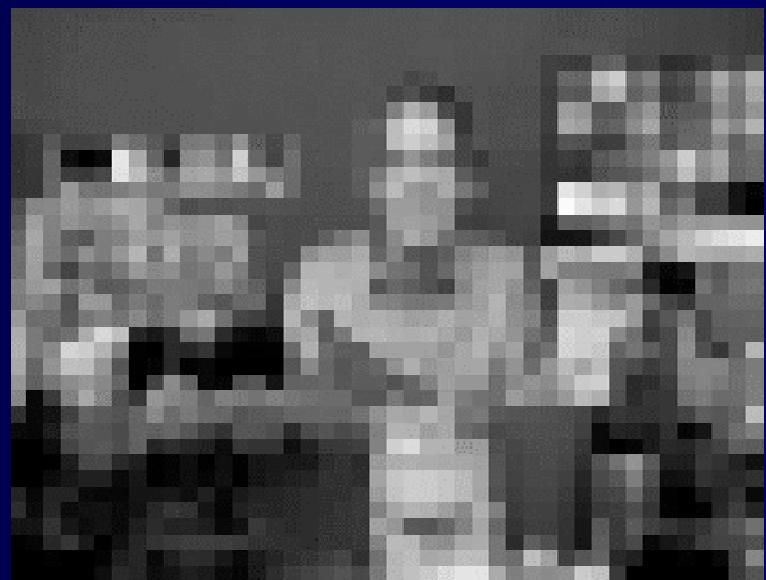
DCT

Wavelet Transform Image(2)

- The lowest frequency component of DWT and DCT
(1/64)



DWT



DCT

画像圧縮技術

- 色信号多重分離
- 直交変換
- サブバンド
- ウエーブレット
- 係数量子化
- エントロピー符号化

Image Compression Technology

- Chrominance Signal Mux/DeMux
- Orthogonal Transform
- Subband
- Wavelet
- Coefficient Quantization
- Entropy Coding

量子化

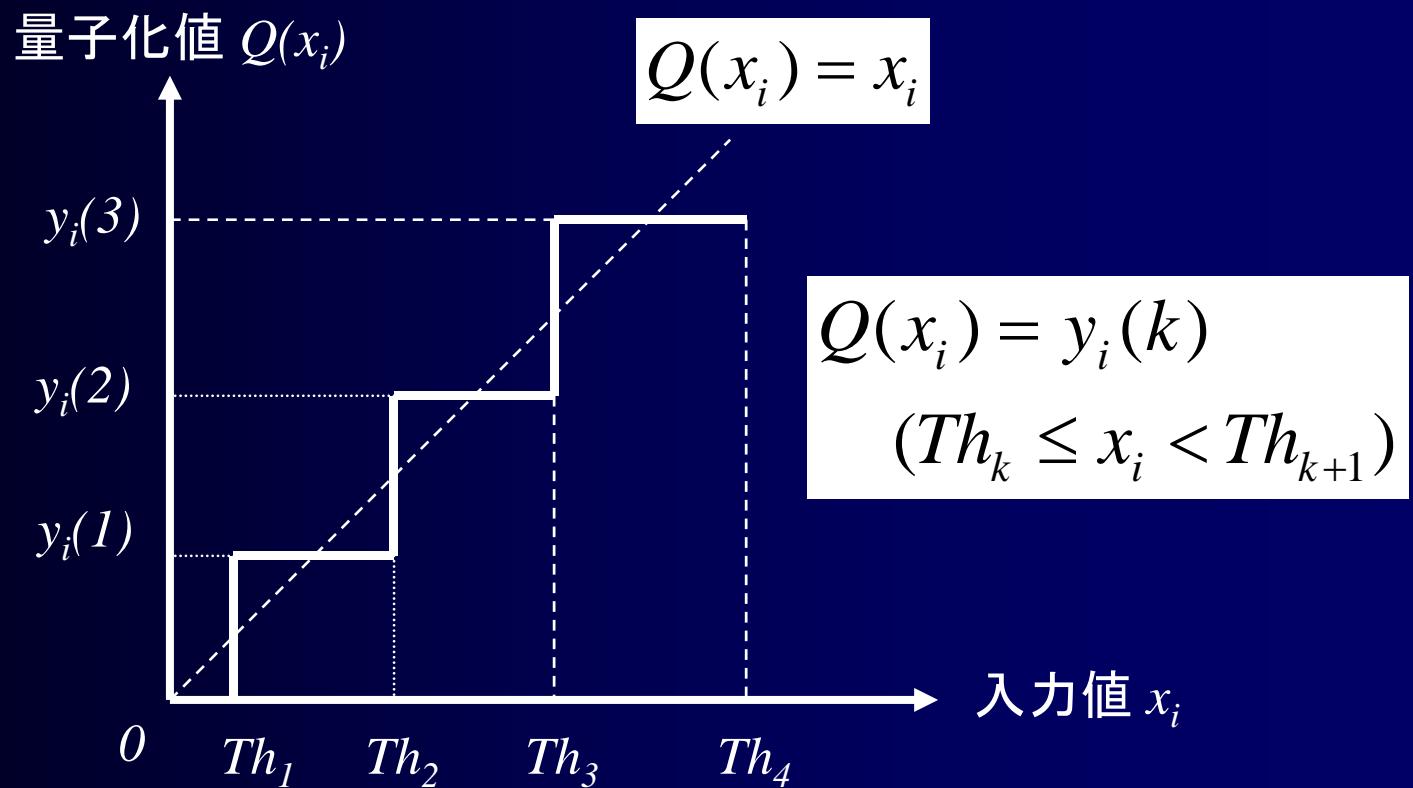
- 量子化の役割
 - 状態数の削減によるデータ圧縮
 - 量子化歪: 量子化することにより生じる誤差
- 量子化の種類
 - スカラー量子化、ベクトル量子化
 - 線形量子化、非線形量子化
- 量子化の最適化
 - Max量子化器
- 画像符号化における量子化
 - JPEG, H.261, MPEG-1, MPEG-2

Quantization

- Role of Quantization
 - Data compression by reducing the number of status
 - Quantization Error: Error caused by quantization
- Types of Quantization
 - Scalar Quantization, Vector Quantization
 - Linear Quantization, Non-linear Quantization
- Optimization of Quantization
 - Max Quantizer
- Quantizer used at Image Coding
 - JPEG, H.261, MPEG-1, MPEG-2

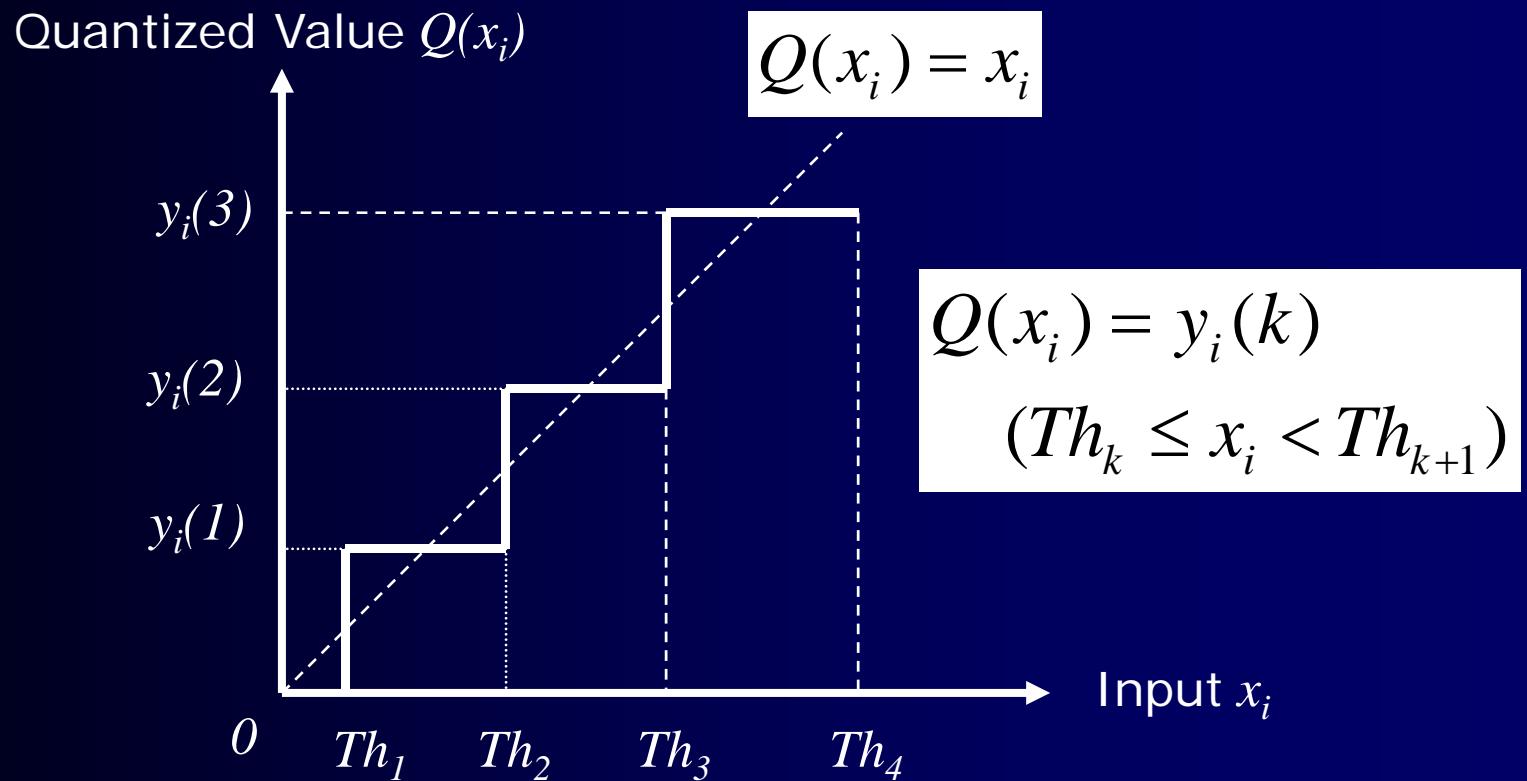
量子化とは？

■ 离散レベル化



What's Quantization?

■ Discrete Level



量子化歪

- スカラの場合の量子化歪と距離尺度(L^p ノルム)

$$d(x, y) = \|x - y\|_p$$
$$\|x - y\|_p = \left(|x - y|^p \right)^{1/p}$$

- ベクトルの場合の量子化歪とノルム

$$d(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X} - \mathbf{Y}\|_p$$
$$\|\mathbf{X} - \mathbf{Y}\|_P = \left(|x_1 - y_1|^p + |x_2 - y_2|^p + \cdots + |x_n - y_n|^p \right)^{1/p}$$

Quantization Distortion

- Quantization Error and Distortion Measure for Scalar's Case (L^p -norm)

$$d(x, y) = \|x - y\|_p$$
$$\|x - y\|_p = \left(|x - y|^p \right)^{1/p}$$

- Quantization Error and Distortion Measure for Vector's Case (L^p -norm)

$$d(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X} - \mathbf{Y}\|_p$$
$$\|\mathbf{X} - \mathbf{Y}\|_p = \left(|x_1 - y_1|^p + |x_2 - y_2|^p + \cdots + |x_n - y_n|^p \right)^{1/p}$$

ノルム

■ 代表的なノルム

$$\|x\|_p = \left(|x_1|^p + |x_2|^p + \cdots + |x_n|^p \right)^{1/p} \quad (\infty > p \geq 1)$$

$$\|x\|_1 = |x_1| + |x_2| + \cdots + |x_n|$$

$$\|x\|_2 = \left(|x_1|^2 + |x_2|^2 + \cdots + |x_n|^2 \right)^{1/2} \quad (\text{Euclidean norm})$$

$$\|x\|_\infty \equiv \max_i |x_i|$$

Norm

■ Typical Norms

$$\|x\|_p = \left(|x_1|^p + |x_2|^p + \cdots + |x_n|^p \right)^{1/p} \quad (\infty > p \geq 1)$$

$$\|x\|_1 = |x_1| + |x_2| + \cdots + |x_n|$$

$$\|x\|_2 = \left(|x_1|^2 + |x_2|^2 + \cdots + |x_n|^2 \right)^{1/2} \quad (\text{Euclidean norm})$$

$$\|x\|_\infty \equiv \max_i |x_i|$$

スカラ量子化とベクトル量子化

■ スカラ量子化

- N 個の要素を持つ入力信号 (N 次元ベクトル) X に対して、 N 個の量子化値を要素として持つ出力信号は、個々の要素の量子化値を並べたもの

$$\mathbf{X} = [x_1 \quad x_2 \quad \cdots \quad x_N]^t$$

$$Q_s(\mathbf{X}) = [Q_s(x_1) \quad Q_s(x_2) \quad \cdots \quad Q_s(x_N)]^t$$

$$Q_s(x_i) = (y_i(k_i) \mid k_i = \arg \min_j d(x_i, y_i(j))) \quad (j = 1, 2, \dots, r_i)$$

$$d(x, y) = \|x - y\|_p$$

Scalar and Vector Quantization

■ Scalar Quantization

- For input signals (N -dimensional vector) having N elements, output signals have N elements which are quantized individually

$$\mathbf{X} = [x_1 \quad x_2 \quad \cdots \quad x_N]^t$$

$$Q_s(\mathbf{X}) = [Q_s(x_1) \quad Q_s(x_2) \quad \cdots \quad Q_s(x_N)]^t$$

$$Q_s(x_i) = (y_i(k_i) \mid k_i = \arg \min_j d(x_i, y_i(j))) \quad (j = 1, 2, \dots, r_i)$$

$$d(x, y) = \|x - y\|_p$$

スカラ量子化とベクトル量子化(2)

■ ベクトル量子化

- N 個の要素を持つ入力信号(N 次元ベクトル) X に対して、準備した代表ベクトル群(コードブック) $Y(i)$ ($i=1,\dots,M$)の中から最も距離の近いものを選び出す

$$\mathbf{X} = [x_1 \quad x_2 \quad \cdots \quad x_N]^t$$
$$Q_V(\mathbf{X}) = [y_1(k) \quad y_2(k) \quad \cdots \quad y_N(k)]^t$$

$$Q_V(\mathbf{X}) = (\mathbf{Y}(k) \mid k = \arg \min_i d(\mathbf{X}, \mathbf{Y}(i))) \quad (i = 1, 2, \dots, M)$$
$$\mathbf{Y}(i) = [y_1(i) \quad y_2(i) \quad \cdots \quad y_N(i)]^t$$
$$d(\mathbf{X}, \mathbf{Y}(i)) = \|\mathbf{X} - \mathbf{Y}(i)\|_p$$

Scalar and Vector Quantization(2)

■ Vector Quantization

- For input signals (N -dimensional vector) having N elements, a set of output signals giving the minimum distortion is selected from codevectors stored in a codebook $Y(i)$ ($i=1,\dots,M$)

$$\mathbf{X} = [x_1 \quad x_2 \quad \cdots \quad x_N]^t$$

$$Q_V(\mathbf{X}) = [y_1(k) \quad y_2(k) \quad \cdots \quad y_N(k)]^t$$

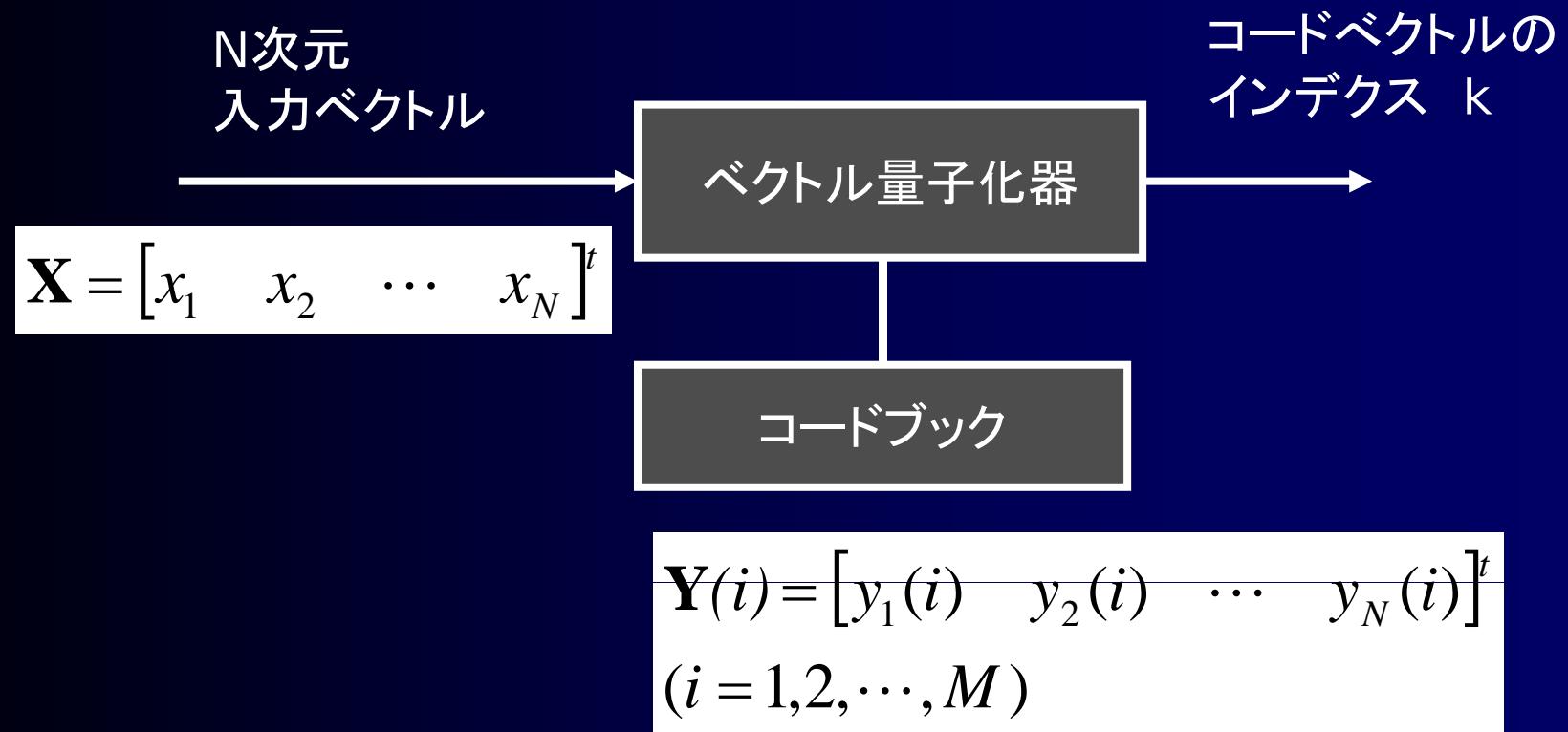
$$Q_V(\mathbf{X}) = (\mathbf{Y}(k) \mid k = \arg \min_i d(\mathbf{X}, \mathbf{Y}(i))) \quad (i = 1, 2, \dots, M)$$

$$\mathbf{Y}(i) = [y_1(i) \quad y_2(i) \quad \cdots \quad y_N(i)]^t$$

$$d(\mathbf{X}, \mathbf{Y}(i)) = \|\mathbf{X} - \mathbf{Y}(i)\|_p$$

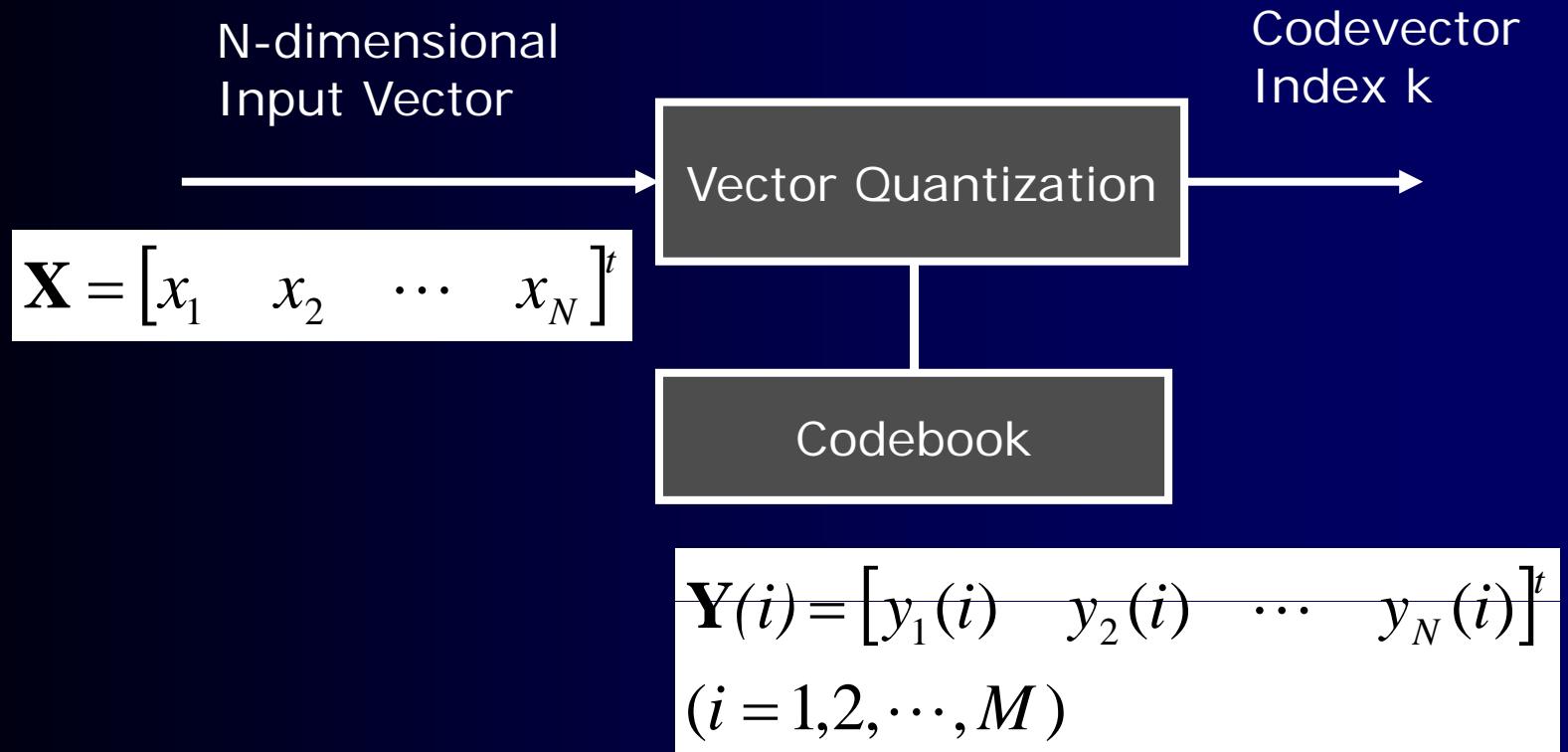
スカラ量子化とベクトル量子化(3)

■ ベクトル量子化の概念図



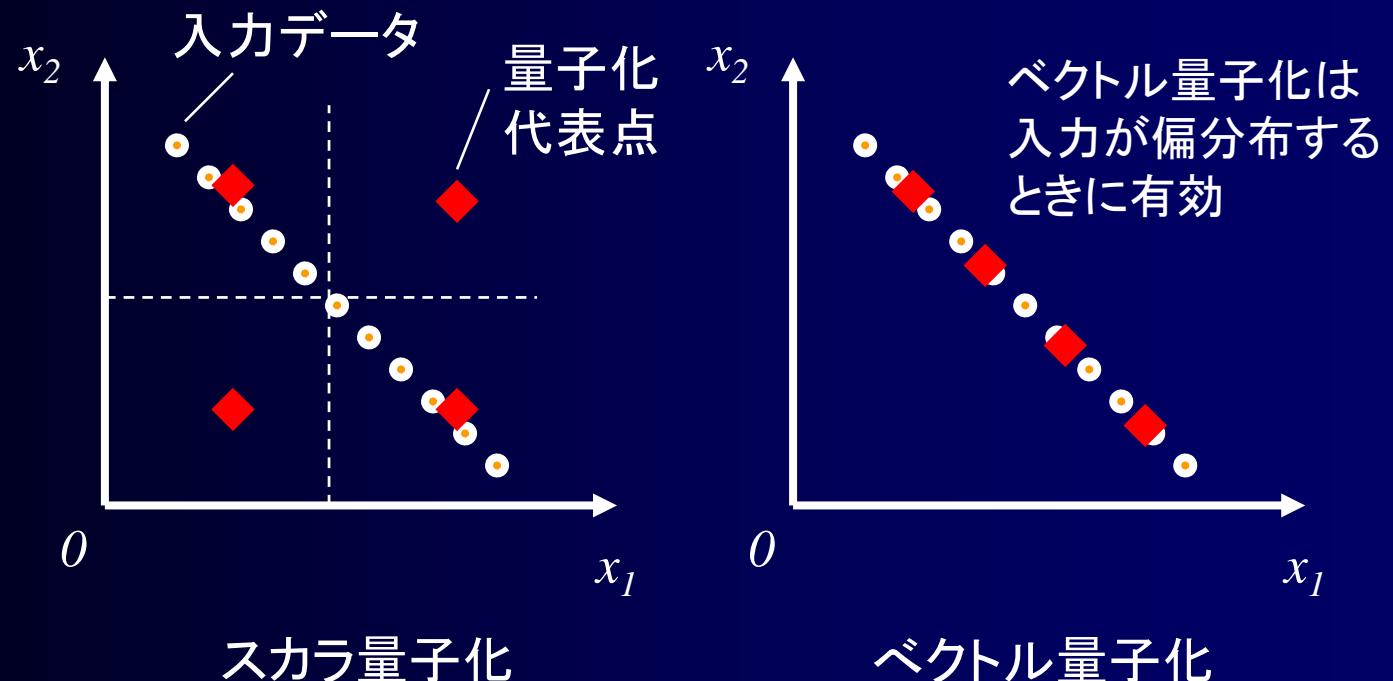
Scalar and Vector Quantization(3)

■ Block Diagram of Vector Quantization



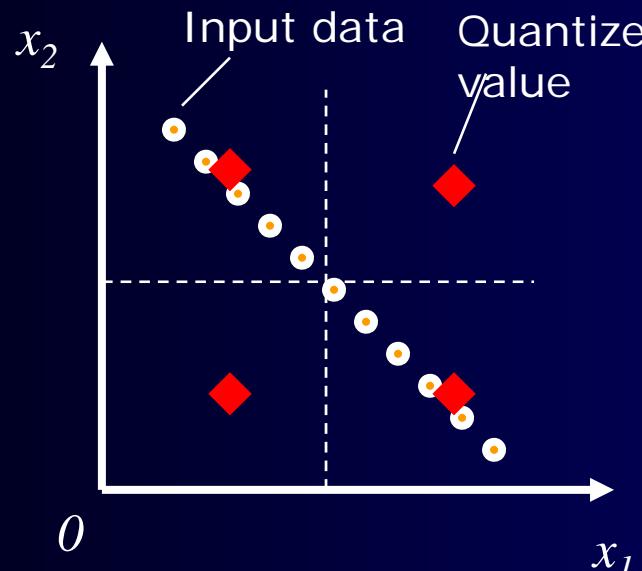
スカラ量子化とベクトル量子化(4)

■ 2次元入力のスカラ量子化とベクトル量子化

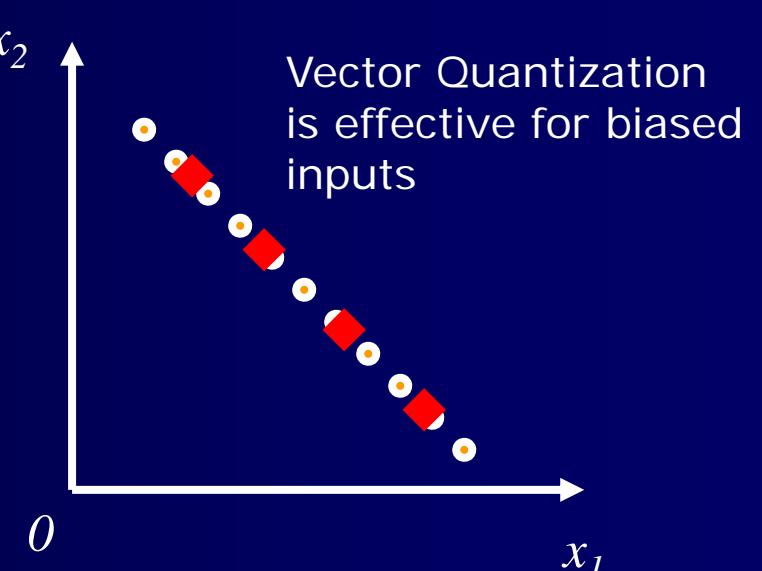


Scalar and Vector Quantization(4)

- Scalar and Vector Quantization for 2-D input



Scalar Quantization



Vector Quantization

ベクトル量子化

- コードブックの作成... LBGアルゴリズム (Linde, Buzo, Gray)
 - 初期コードブック $C_0 = \{Y_0(i)\} (i=1,2,\dots,N)$
 - トレーニング集合 T を N 個の集合 $R_0(i)$ に分割 (最近傍探索)

$$R_0(i) = \{\mathbf{X} \in \mathbf{T} : d(\mathbf{X}, \mathbf{Y}_0(i)) < d(\mathbf{X}, \mathbf{Y}_0(j)); \text{for } j \neq i\}$$

- セントロイドベクトルの計算
$$\mathbf{Y}_1(i) = E[\mathbf{X} | \mathbf{X} \in R_0(i)]$$
- 全体での誤差が一定以下になるまで、この処理の繰り返し

Vector Quantization

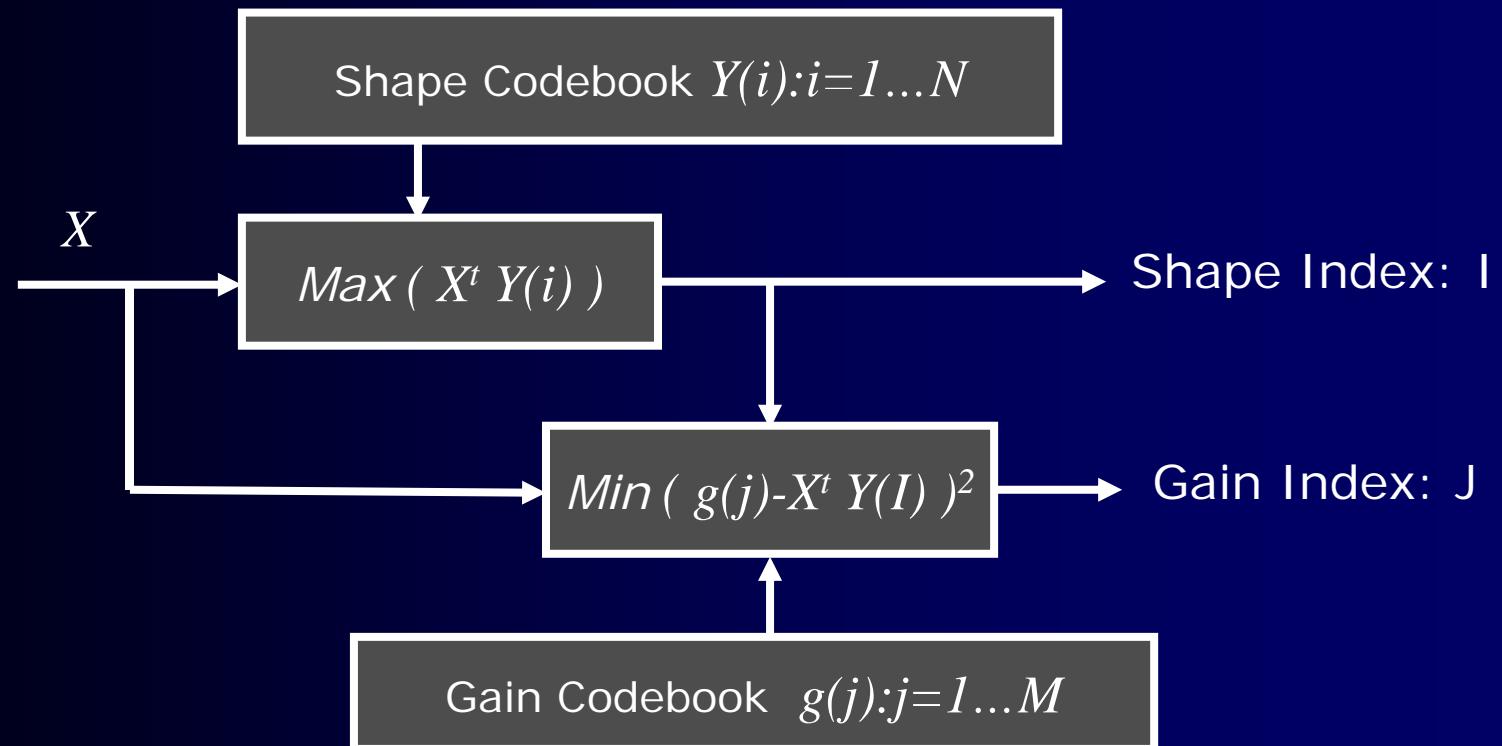
- Codebook Generation ... LBG Algorithm (Linde, Buzo, Gray)
 - Initial codebook $C_0 = \{Y_0(i)\} (i=1,2,\dots,N)$
 - Divide training set T to N sets $R_0(i)$ (Nearest Neighbor Search)

$$R_0(i) = \{\mathbf{X} \in \mathbf{T} : d(\mathbf{X}, \mathbf{Y}_0(i)) < d(\mathbf{X}, \mathbf{Y}_0(j)); \text{for } j \neq i\}$$

- Calculation of Centroid Vector
$$\mathbf{Y}_1(i) = E[\mathbf{X} | \mathbf{X} \in R_0(i)]$$
 - Repeat this procedure until total error will reach to a certain level

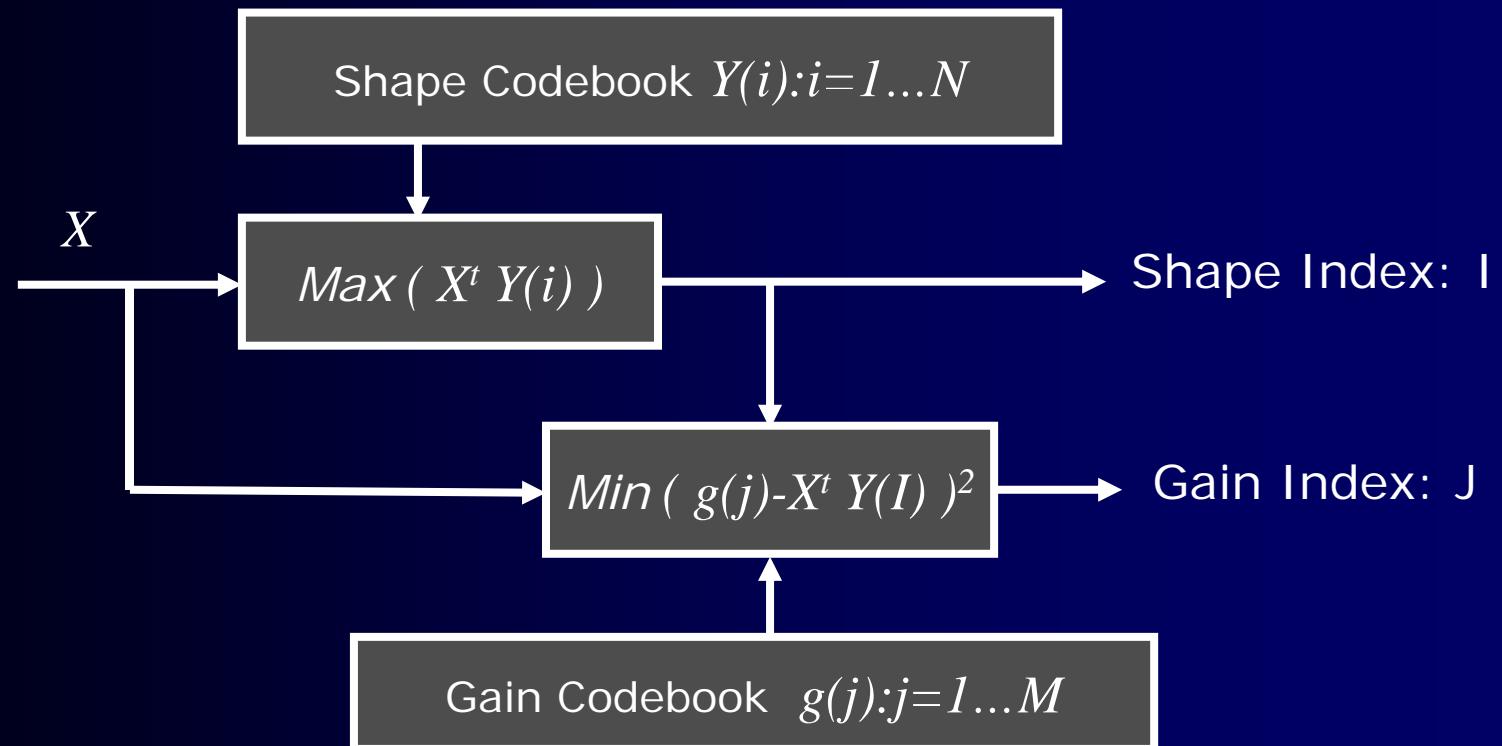
ベクトル量子化(2)

■ Shape-Gain ベクトル量子化



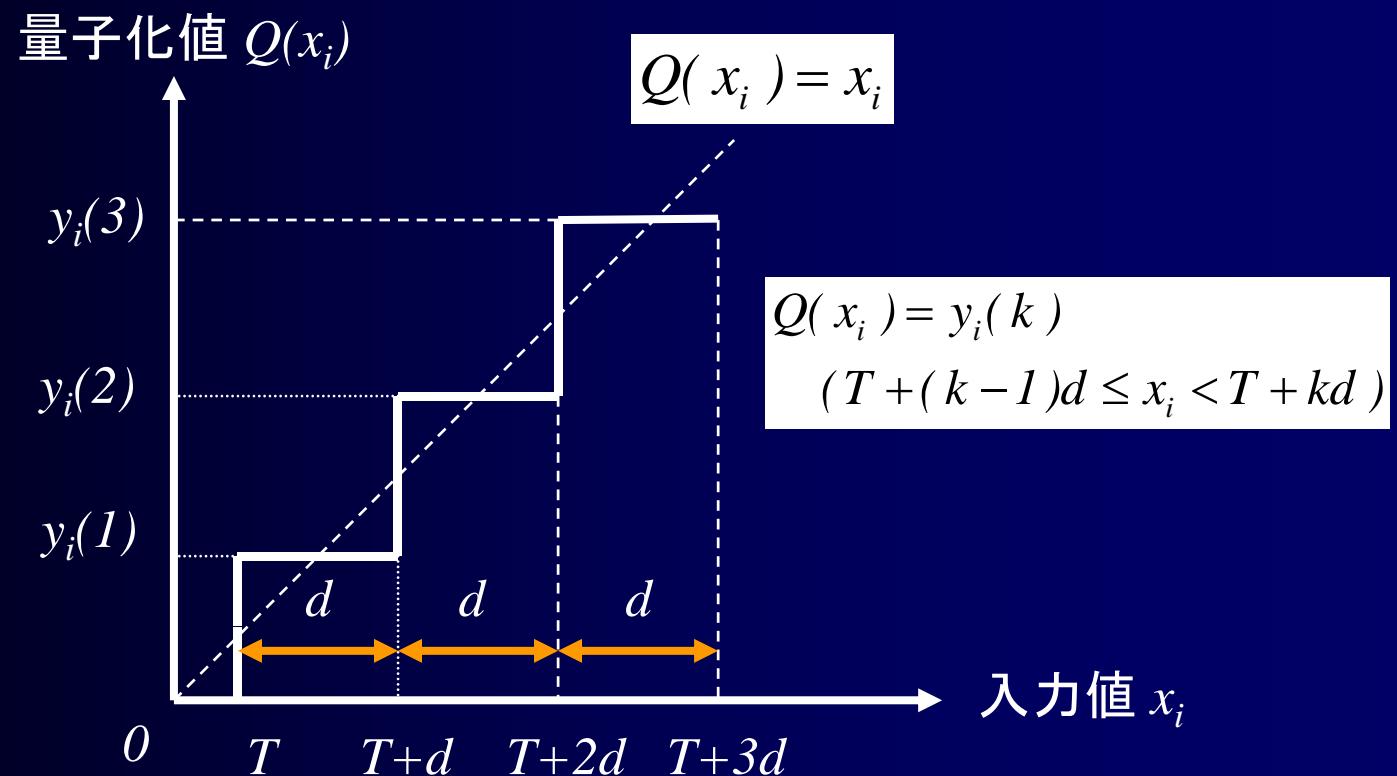
Vector Quantization(2)

■ Shape-Gain Vector Quantization



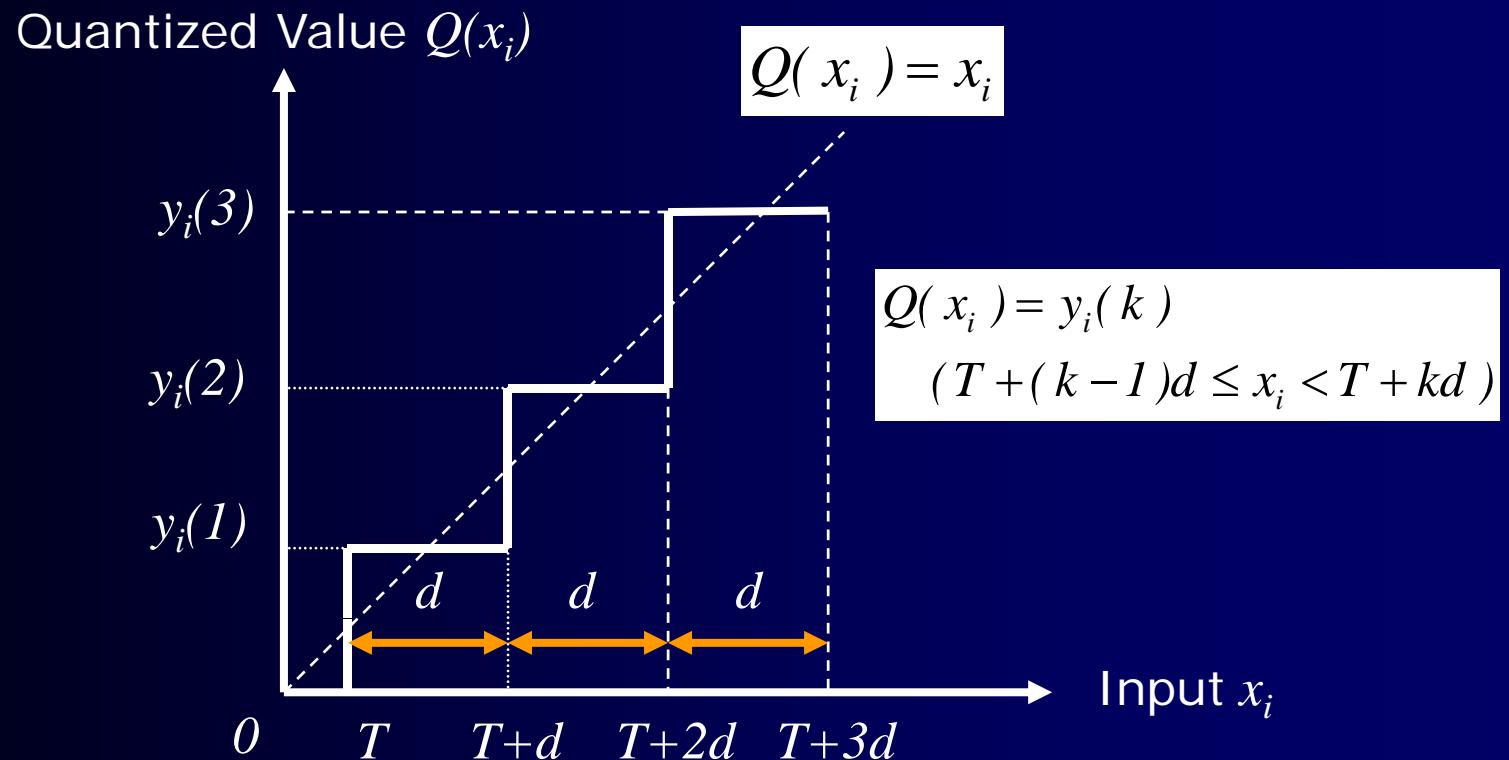
線形量子化

- 量子化間隔 d が一定(一様量子化とも呼ぶ)



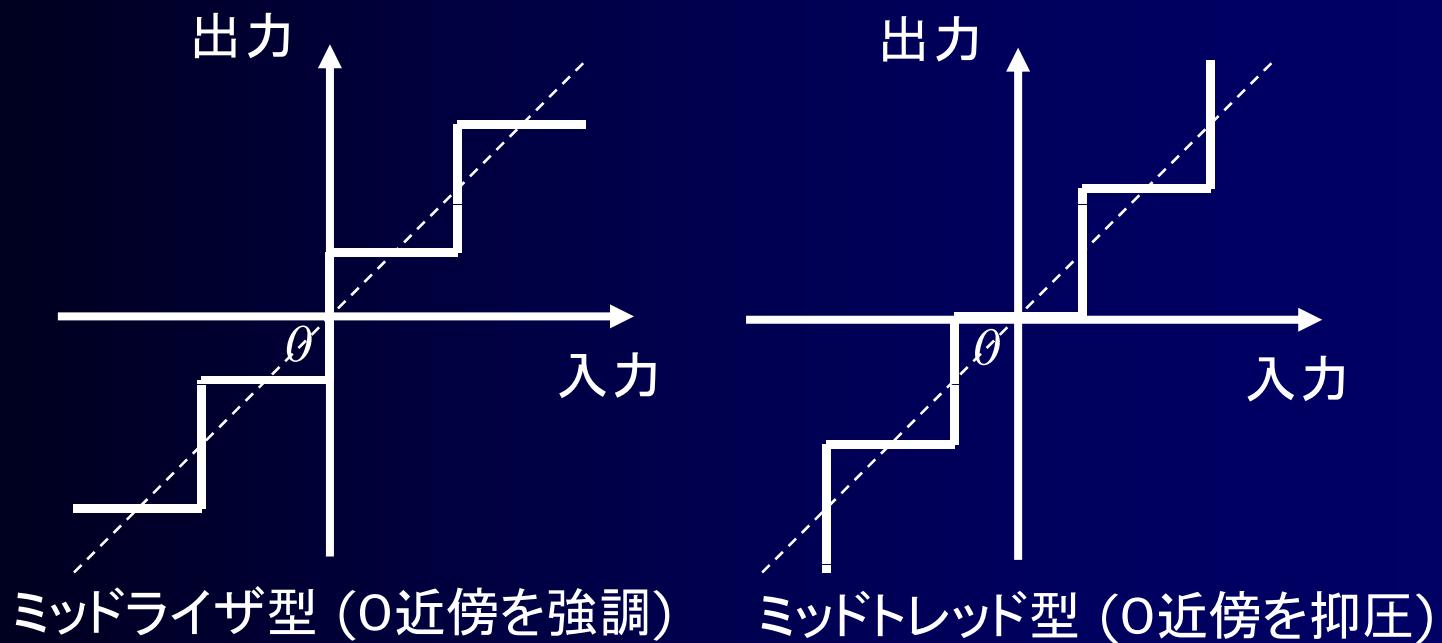
Linear Quantization

- Quantization Step d is constant (uniform quantization)



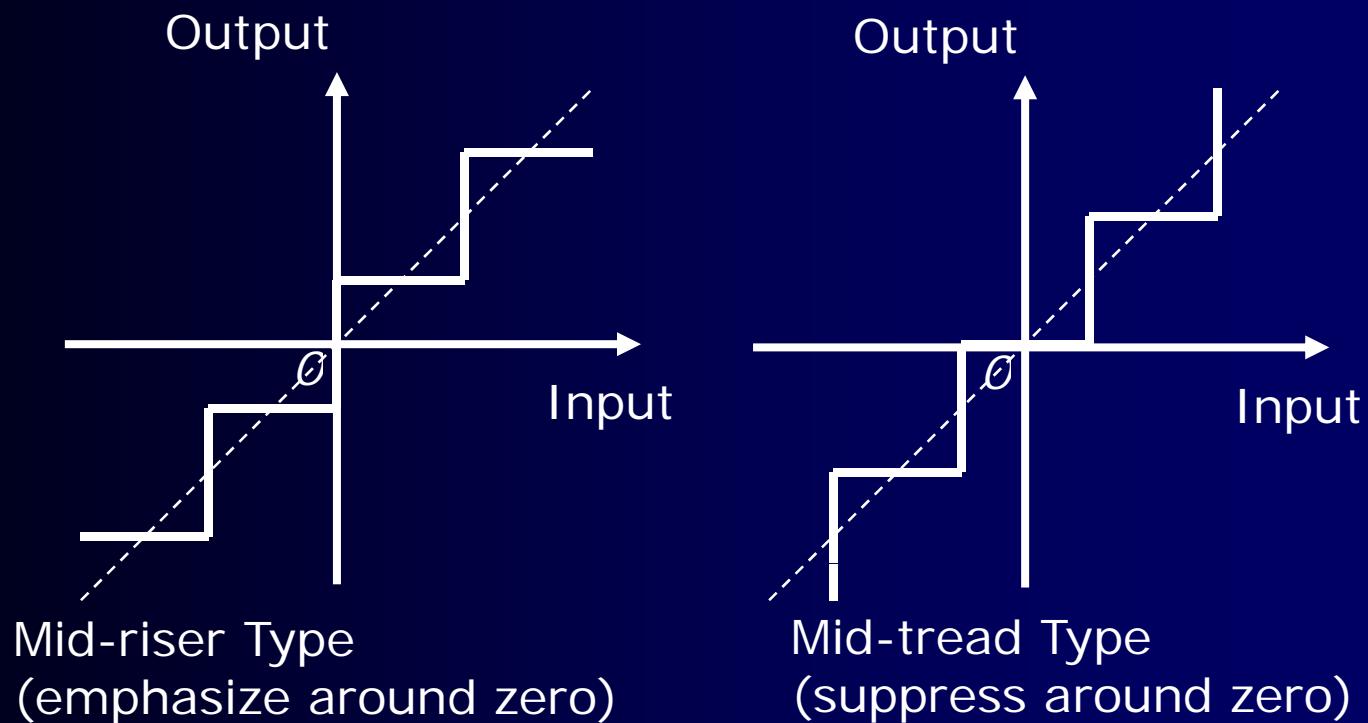
線形量子化(2)

■ ミッドライザ型とミッドトレッド型



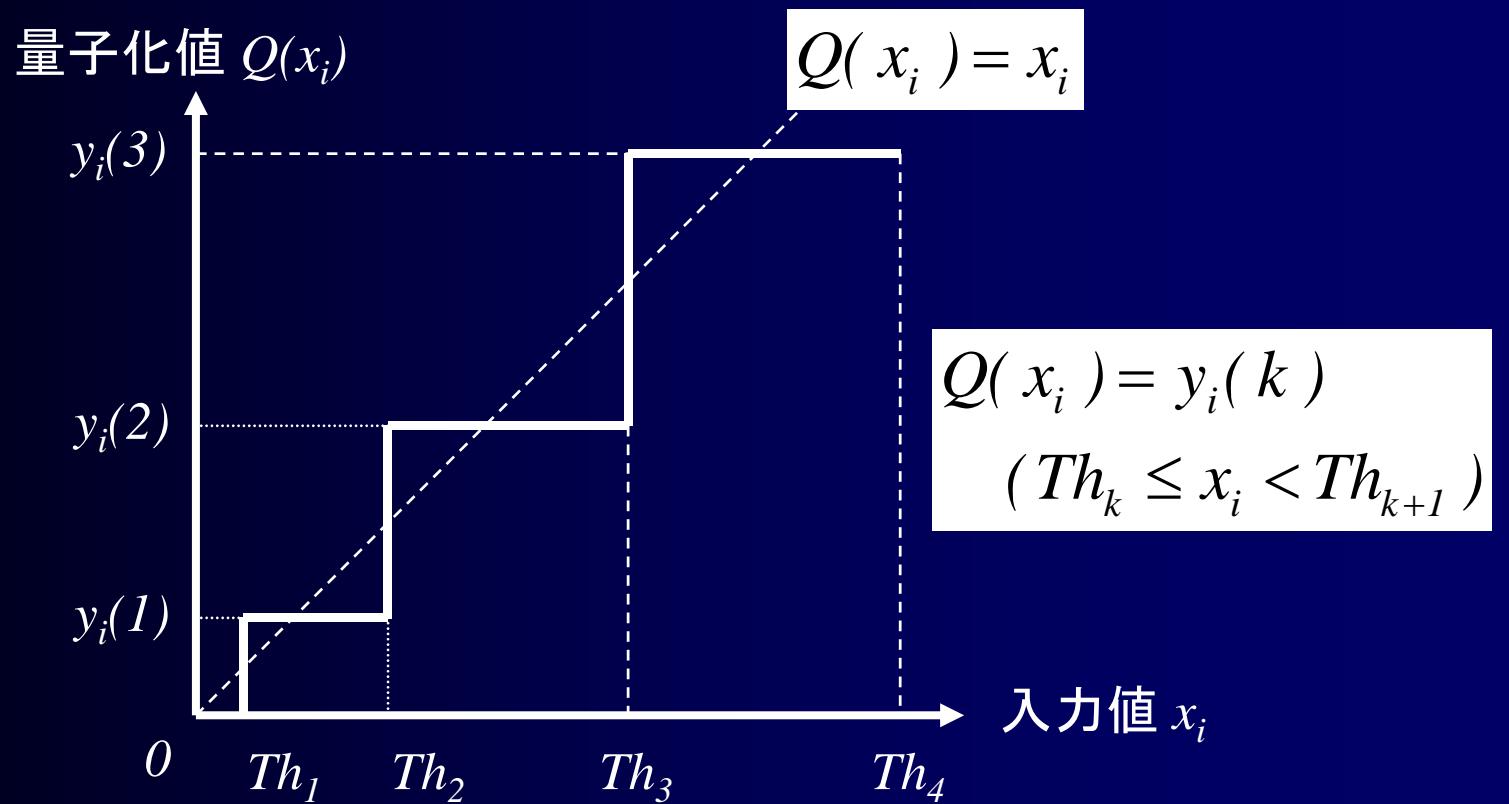
Linear Quantization(2)

■ Mid-riser and Mid-tread Type



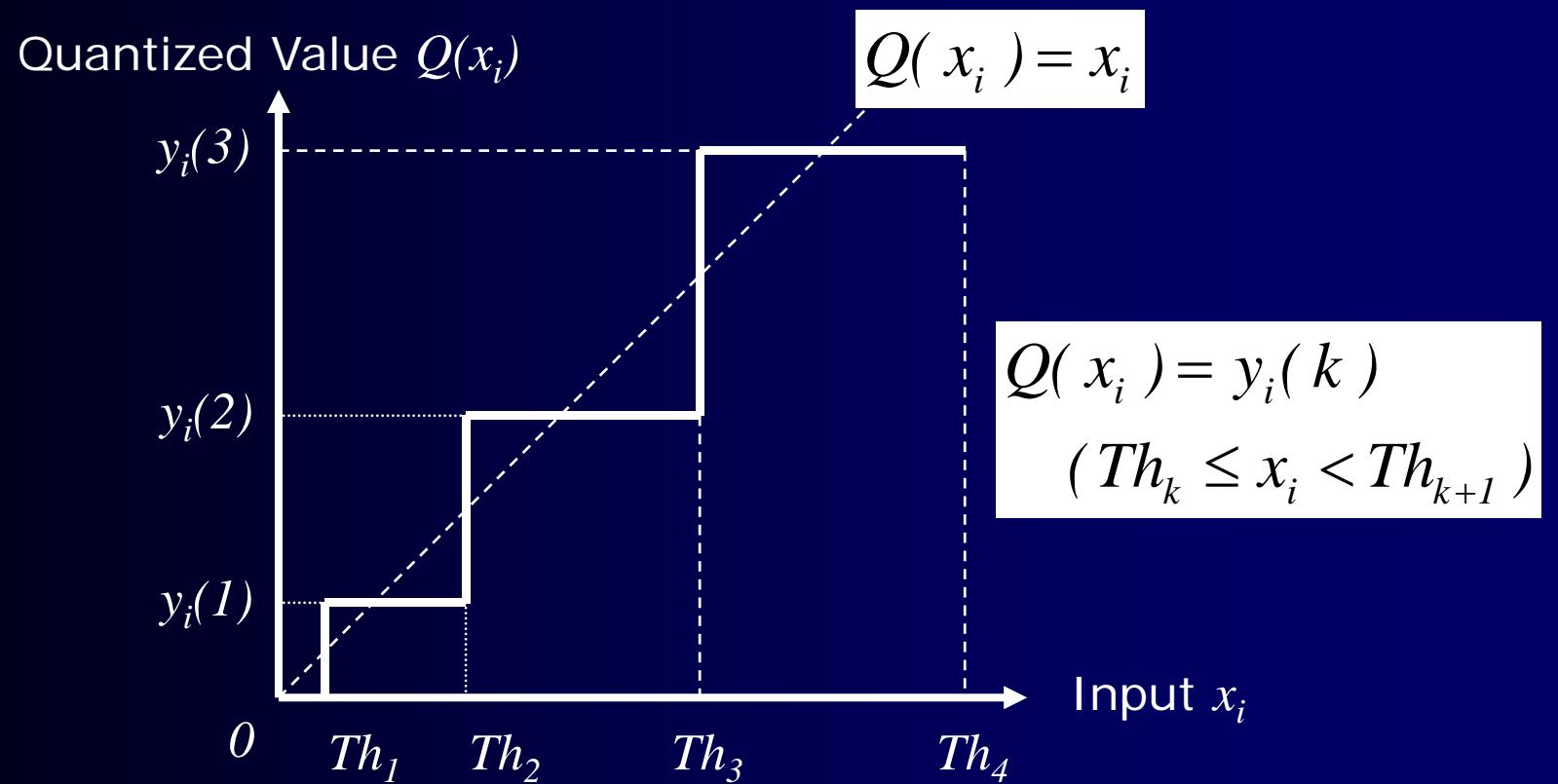
非線形量子化

- 量子化ステップが非一様



Non-linear Quantization

- Quantization Step in non-uniform



非線形量子化(2)

- 信号の特性と量子化ステップの関係
 - 入力信号の出現確率に合わせる
 - 均一な量子化に比べ、平均的な量子化誤差を低減できる
- DPCMのように0付近の信号の出現確率が高い場合
 - 0付近の入力値は出現頻度が高いので細かく量子化
 - 大きな入力値は、出現頻度が低いので粗く量子化
- 入力信号の出現確率と量子化ステップの関係
 - 平均的な量子化誤差の最適化(最小化)が可能

Non-linear Quantization(2)

- Relation of signal characteristics and quantization step
 - Fit to input signal's occurrence probability
 - Can reduce average distortion than linear quantization
- In case of high occurrence probability around zero such as DPCM
 - Fine quantization around zero
 - Coarse quantization to large inputs
- Relation of input's occurrence probability and quantization steps
 - Optimization (Minimization) of the average distortion is possible

画像圧縮技術

- 色信号多重分離
- 直交変換
- サブバンド
- ウエーブレット
- 係数量子化
- エントロピー符号化

Image Compression Technology

- Chrominance Signal Mux/DeMux
- Orthogonal Transform
- Subband
- Wavelet
- Coefficient Quantization
- Entropy Coding

固定長符号と可変長符号 (1)

- 固定長符号 (Fixed Length Code: FLC)
 - シンボルA,B,C,...を表現するバイナリ符号はすべて同じ長さ
 - シンボルA,B,C,...の生起確率が等しい場合に有効
 - 自然2進符号、Gray Code
- 可変長符号 (Variable Length Code: VLC)
 - バイナリ符号の長さがシンボルA,B,C,...ごとに異なる符号
 - シンボルA,B,C,...の生起確率が異なる場合に有効
 - ハフマン符号、算術符号
 - 可変長符号化を行うことを、エントロピー符号化 (Entropy Coding) と呼ぶこともある

Fixed Length Code and Variable Length Code (1)

- Fixed Length Code: FLC
 - Equal length binary code for Symbol A,B,C,...
 - Effective when occurrence probability for A,B,C,... is same
 - Natural binary code, Gray Code
- Variable Length Code: VLC
 - Unequal length binary code for Symbol A,B,C,...
 - Effective when occurrence probability for A,B,C,... is different
 - Huffman coding, Arithmetic coding
 - Variable length coding is sometimes called Entropy Coding

固定長符号と可変長符号(2)

シンボル	生起確率	固定長符号	ハフマン符号
A	0.4	000	0
B	0.2	001	100
C	0.15	010	101
D	0.1	011	110
E	0.05	100	11100
F	0.04	101	11101
G	0.03	110	11110
H	0.03	111	11111

エントロピー = 2.44 ビット 3 ビット/シンボル 2.5 ビット/シンボル

Fixed Length Code and Variable Length Code(2)

Symbol	Probability	FLC	Huffman Code
A	0.4	000	0
B	0.2	001	100
C	0.15	010	101
D	0.1	011	110
E	0.05	100	11100
F	0.04	101	11101
G	0.03	110	11110
H	0.03	111	11111

Entropy = 2.44 bit

3 bit/symbol

2.5 bit/symbol

符号の種類

- 非ブロック符号.....(算術符号)
- ブロック符号
 - 特異な符号
 - 非特異な符号
 - 一意復号不可能な符号
 - 一意復号可能な符号
 - 瞬時復号不可能な符号
 - 瞬時復号可能な符号.....(ハフマン符号)

Type of Codes

- Non-block Code..... (Arithmetic Code)
- Block Code
 - Singular Code
 - Non-singular Code
 - Uniquely undecodable Code
 - Uniquely Decodable Code
 - Non-instant decodable code
 - Instant decodable code..... (Huffman Code)

ハフマン符号

■ 構成法

- 各シンボルに対する枝を作り出現確率を与える
- 確率の最も小さい2つの枝を節点で結び、符号“0”“1”を与える
- 統合した枝の確率の和を求め、ひとつの枝とみなす
- 再度、確率も最も小さい2つを枝を探す
- この処理を枝がひとつになるまで繰り返す

Huffman Code

■ Structure

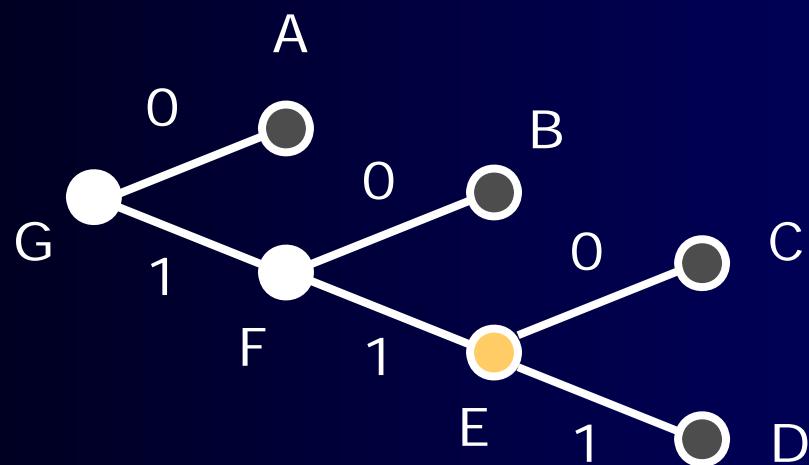
- Make branch to each symbol and assign probability
 - Combine two branches giving least two probability and assign "0""1"
 - Obtain probability of unified branch, and regarded as one branch
-
- Again, search branches giving least two probabilities
 - Repeat this procedure until branches unified to one

ハフマン符号(2)

■ ハフマン符号の構成例

- CとDを統合しE(0.15)とする
- BとEを統合しF(0.4)とする
- FとAを統合しG(1.0)とする

シンボル	確率	符号語
A	0.6	0
B	0.25	10
C	0.1	110
D	0.05	111

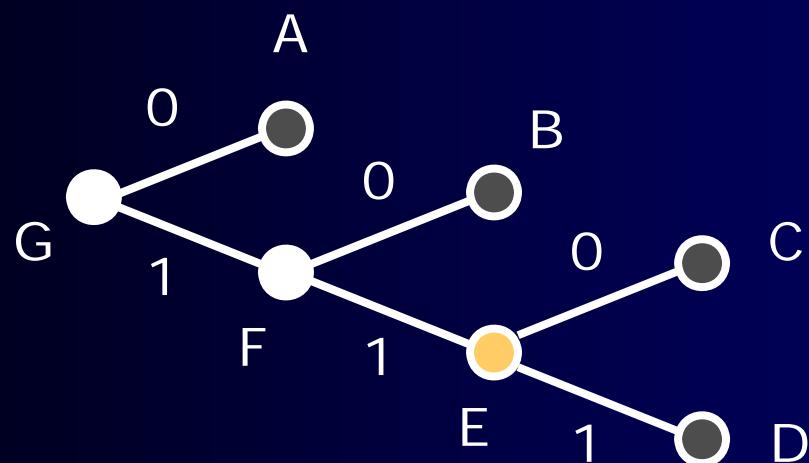


Huffman Code(2)

- Example of constructing Huffman code

- Unify C and D, make E(0.15)
- Unify B and E, make F(0.4)
- Unify F and A, make G(1.0)

Symbol	Probability	Code
A	0.6	0
B	0.25	10
C	0.1	110
D	0.05	111



ハフマン符号の応用

- 画像符号化におけるハフマン符号
 - 2値画像符号化におけるランレンジス符号化
 - H.261におけるDCT係数の2次元VLC
 - 動きベクトルのVLC
 - Coded Block PatternのVLC
 - Macroblock typeのVLC

Application of Huffman Code

- Huffman Code at Image Coding
 - Run-length coding at binary image coding
 - 2-D VLC of DCT coefficient at H.261
 - VLC for motion vectors
 - VLC for Coded Block Pattern
 - VLC for Macroblock type

ランレングス符号化

- 2値画像では、白画素と黒画素の連続する長さをシンボルとみなし、ハフマン符号化を行う
- 白画素を0、黒画素を1とすると、連続する0の個数 (Zero Run Length), 連続する1の個数に対してハフマン符号を設計

0 0 0 0 1 1 0 0 0 0 0 1 1 1 0 0
|← 4 →|← 2 →|← 5 →|← 3 →|

Run-length Coding

- Successive length of white and black pixels are regarded as symbol and Huffman coding is applied at binary image coding
- White pixel “0”, black pixel “1”, Huffman code is designed to successive number of “0” (Zero Run Length) and “1”

0 0 0 0 1 1 0 0 0 0 0 1 1 1 0 0
|← 4 →|← 2 →|← 5 →|← 3 →|

算術符号

- 2元情報源のシンボル(“0”, “1”) の系列を任意の長さに区切る
- 先頭に小数点を仮定し、コードワード(符号語)を2進数に見立てる
- 2進数を区間 $[0, 1]$ に射影
- 射影区間を一意に決定するための、区間決定ビット数が決まる
- 区間決定コードワードを出力する

Arithmetic Coding

- Bound sequence of binary source symbol (“0”, “1”) in arbitrary length
- Set fractional point at the head, regard codeword as a binary number
- Project binary number to section [0,1]
- Bit number is counted to determine the section
- Output codeword to specify the section

算術符号(2)

- Eliasの符号
 - 算術符号化の原理
 - 実用化には問題点
 - 演算精度の不足
 - リアルタイム復号が不可能
- LR(Langdon-Rissanen)型算術符号
 - 実用化の出発点
 - 演算精度の対策
 - リアルタイム復号可能

Arithmetic Coding(2)

- Elias Code
 - Principle of Arithmetic Coding
 - There exists problem for practical use
 - Insufficient operation accuracy
 - Unable to decode in realtime
- LR (Langdon-Rissanen) Type Arithmetic Coding
 - Starting point for practical application
 - Remedy for operational accuracy
 - Decodable in realtime

Eliasの符号

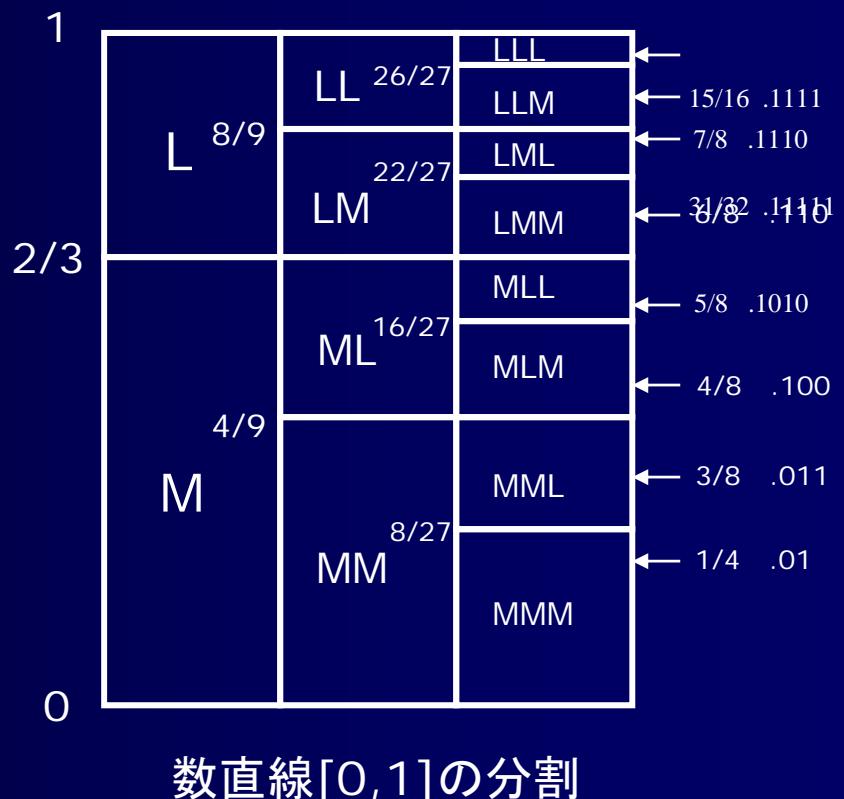
■ 符号化のプロセス

- 優勢シンボル (MPS) と劣勢シンボル (LPS) の生起確率比による数直線[0,1]の多段分割

MPS: More probable symbol

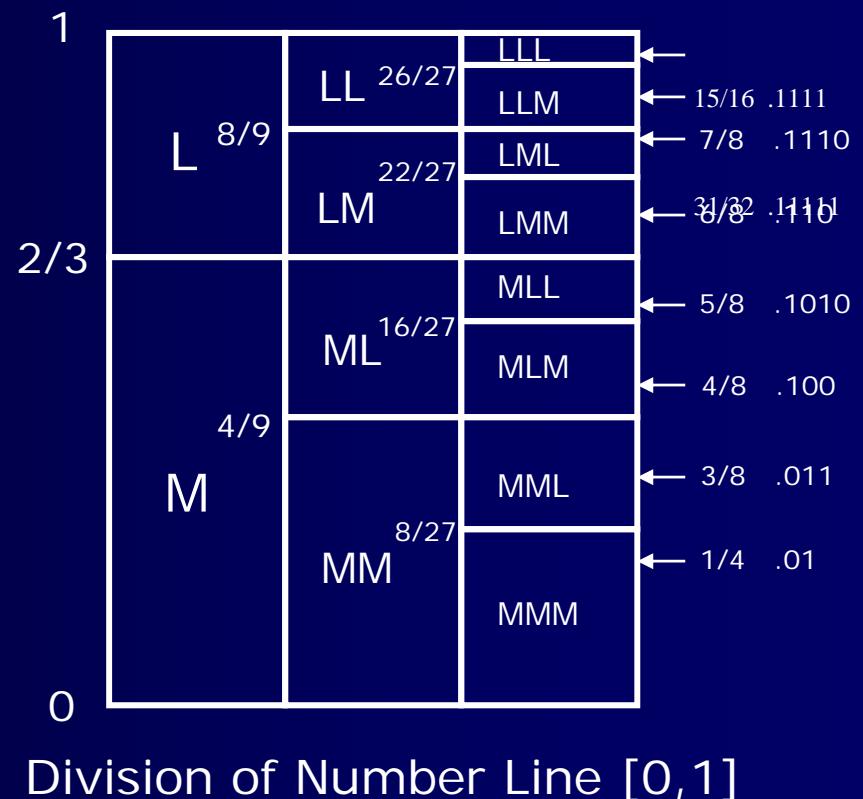
LPS: Less probable symbol

- $P(M)=2/3, P(L)=1/3$ の例
- 3個連続するシンボル系列は，“MMM”から“LLL”まで8種類の確率区間(オージェンド)に対応



Elias Code

- Coding Process
- Multistage division of number line [0,1] by the ratio of occurrence probability
 - MPS: More probable symbol
 - LPS: Less probable symbol
- Example of $P(M)=2/3$, $P(L)=1/3$
- Three successive symbol corresponds to the probability section (Augend) from “MMM” to “LLL”



Eliasの符号(2)

■ 算術符号化のプロセス

- シンボル系列“MMM”は区間[0, 8/27]に相当
“1”がMPSであれば、シンボル系列は“111”
“0”がMPSであれば、シンボル系列は“000”
- このオージェンドを表現する最も短い2進数は0.01
- 小数点以下の2ビットでオージェンドを指定できる
- 各オージェンドに対して、一意復号可能な符号を別途作成

Elias Code(2)

- Process of Arithmetic Coding
 - Symbols “MMM” corresponds to the section [0, 8/27]
 - If “1” is MPS, symbols are “111”
 - If “0” is MPS, symbols are “000”
 - The minimum binary number to specify this Augend is 0.01
 - We can specify the Augend by fractional two bits
 - Create uniquely decodable code to each Augend

Eliasの符号(3)

■ 分割された各オージェンドと符号長

Symbols	Augend	$-\log_2(A)$	Code Length (bit)
LLL	1/27	4.75	5
LLM	2/27	3.75	4
LML	2/27	3.75	4
LMM	4/27	2.75	3
MLL	2/27	3.75	4
MLM	4/27	2.75	3
MML	4/27	2.75	3
MMM	8/27	1.75	2

Elias Code(3)

■ Divided Augend and Code Length

Symbols	Augend	$-\log_2(A)$	Code Length (bit)
LLL	1/27	4.75	5
LLM	2/27	3.75	4
LML	2/27	3.75	4
LMM	4/27	2.75	3
MLL	2/27	3.75	4
MLM	4/27	2.75	3
MML	4/27	2.75	3
MMM	8/27	1.75	2

算術符号の原理

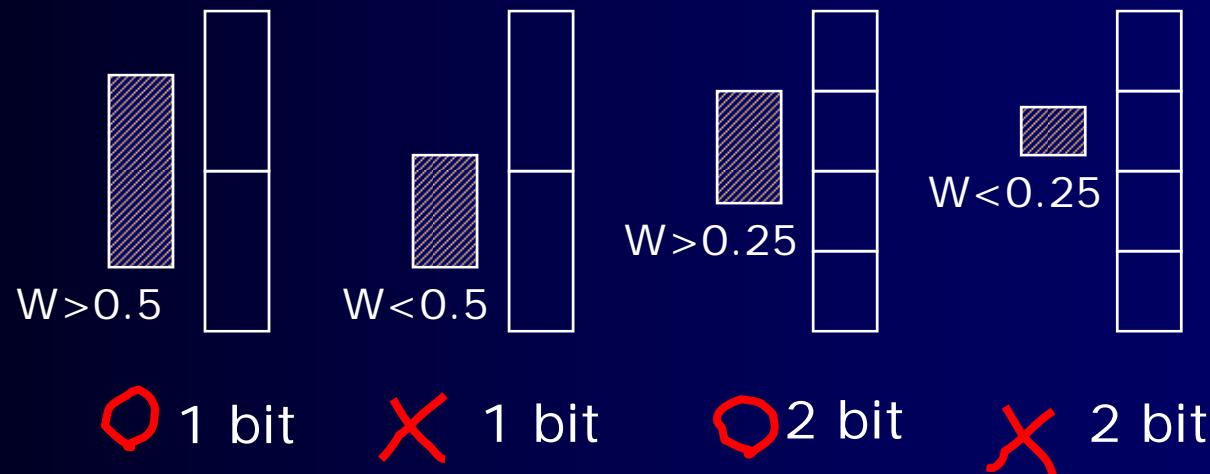
■ 符号化のプロセス

- MMMからLLLまでのオージェンドを計算
- オージェンドを表現するのに最低何ビット必要か?
 - オージェンド W は M が a 回, L が b 回であれば,
$$W=P(M)^aP(L)^b$$



Principle of Arithmetic Coding

- Coding Process
 - Calculate augends from MMM to LLL
 - At least how many bits required to specify Augend?
 - When M is a -times, L is b -times, Augend
 $W=P(M)^aP(L)^b$



算術符号の原理(2)

- オージェンド W の符号化に要するビット数

$$B = -\log_2 W + 1$$

- オージェンド W は

$$W = P(M)^a P(L)^b$$

であるから

$$\overline{B = -\log_2(P(M)^a P(L)^b) + 1}$$

Principle of Arithmetic Coding(2)

- Bit number required for coding of Augend W

$$B = -\log_2 W + 1$$

- Augend W is represented as

$$W = P(M)^a P(L)^b$$

Thus bit number is given by

$$\boxed{B = -\log_2(P(M)^a P(L)^b) + 1}$$

算術符号の原理(3)

- $(a+b)$ 個のシンボルに対する符号長

$$\begin{aligned}B &= -\log_2 W + 1 \\&= -\log_2 (P(M)^a P(L)^b) + 1 \\&= -\log_2 P(M)^a - \log_2 P(L)^b + 1 \\&= -a \log_2 P(M) - b \log_2 P(L) + 1 \\&= (a+b) \left(-\frac{a}{a+b} \log_2 P(M) - \frac{b}{a+b} \log_2 P(L) \right) + 1\end{aligned}$$

Principle of Arithmetic Coding(3)

- Code length for $(a+b)$ symbols

$$\begin{aligned}B &= -\log_2 W + 1 \\&= -\log_2 (P(M)^a P(L)^b) + 1 \\&= -\log_2 P(M)^a - \log_2 P(L)^b + 1 \\&= -a \log_2 P(M) - b \log_2 P(L) + 1 \\&= (a+b) \left(-\frac{a}{a+b} \log_2 P(M) - \frac{b}{a+b} \log_2 P(L) \right) + 1\end{aligned}$$

算術符号の原理(4)

- シンボルの個数 $N=a+b$ が無限大になれば以下が成り立つ

$$\lim_{N \rightarrow \infty} \frac{a}{N} = P(M) \quad \lim_{N \rightarrow \infty} \frac{b}{N} = P(L)$$

- このとき N 個のシンボルに対する符号長はエントロピー H を用いて以下のように書ける

$$\begin{aligned} B &= (a+b) \left(-\frac{a}{a+b} \log_2 P(M) - \frac{b}{a+b} \log_2 P(L) \right) + 1 \\ &= N \left(-P(M) \log_2 P(M) - P(L) \log_2 P(L) \right) + 1 \\ &= NH + 1 \end{aligned}$$

Principle of Arithmetic Coding(4)

- The next relation holds when symbol number $N=a+b$ becomes infinity

$$\lim_{N \rightarrow \infty} \frac{a}{N} = P(M) \quad \lim_{N \rightarrow \infty} \frac{b}{N} = P(L)$$

- Code length for N symbols can be written by entropy H

$$\begin{aligned} B &= (a+b) \left(-\frac{a}{a+b} \log_2 P(M) - \frac{b}{a+b} \log_2 P(L) \right) + 1 \\ &= N \left(-P(M) \log_2 P(M) - P(L) \log_2 P(L) \right) + 1 \\ &= NH + 1 \end{aligned}$$

算術符号の原理(5)

- 1シンボルあたりの平均符号長 b は

$$b = \frac{B}{N} = H + \frac{1}{N}$$

であるから、 N が無限大に近づけば、 $1/N$ は 0 に近づき、平均符号長 b はエントロピー H に漸近する

$$\lim_{N \rightarrow \infty} b = H + \lim_{N \rightarrow \infty} \frac{1}{N}$$

Principle of Arithmetic Coding(5)

- Average code length b per 1 symbol is given by

$$b = \frac{B}{N} = H + \frac{1}{N}$$

so that, when N approaches to infinity, $1/N$ approaches to 0 , and average code length b approaches to entropy H

$$\lim_{N \rightarrow \infty} b = H + \lim_{N \rightarrow \infty} \frac{1}{N}$$