Waveforms vs Wavelets

Starting point: given a signal, sampled at N points

Fourier — assumed periodic

Wavelet — not necessarily periodic

Waveforms (Fourier) are nonzero everywhere — always magnitude 1

$$E_k[j] = \omega^{jk}$$

Wavelets can be supported on smaller sets — have 0 value outside of some interval.

at least mostly in one place.

Fourier

low vs high frequencies

frequency spectrum

Wavelet

large vs small support

where wavelet is ≠0 or large.

feature scale localization.

(edge detection)

More precisely: Looking for alternate bases for $\mathbb{C}^N$ (or $\mathbb{R}^N$) given by "wavelet basis" $w_0, \ldots, w_N$ and describe lin trans. changing basis.

Also want computations to be reasonably efficient.

want lin. trans. in terms of easy matrices

Easy Matricies

- Permutation.
- Diagonal
- Upper & lower triangular. - easy to invert
- "Sparse" - most entries are 0.

Ex: Fast Fourier $\quad F_N f[k] = F_M f_{even}[k] + \omega^{-k} F_M f_{odd}[k]$

$N = 2M$

$$\vec{f} = \begin{bmatrix} f[0] \\ f[1] \\ \vdots \\ f[2M-1] \end{bmatrix} \qquad \boxed{2\downarrow}\,\vec{f} = \begin{bmatrix} f[0] \\ f[2] \\ \vdots \\ f[2M-2] \end{bmatrix}$$

down sampling by 2

$$\vec{f}_{even} = \boxed{2\downarrow}\,\vec{f}$$

$$\boxed{2\downarrow} = \begin{array}{c} \\ \left[\begin{array}{cccccc} 1 & 0 & 0 & 0 & \cdots & 0 \;\; 0 \\ 0 & 0 & 1 & 0 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \;\; 0 \end{array}\right] M \\ N \end{array}$$

$$S\vec{f} = \begin{bmatrix} f[2M-1] \\ f[0] \\ f[1] \\ \vdots \\ f[2M-2] \end{bmatrix} \qquad \vec{f}_{odd} = \boxed{2\downarrow}\, S^{-1}\vec{f}$$

$$\begin{bmatrix} \vec{f}_{even} \\ \hline \vec{f}_{odd} \end{bmatrix} = \begin{bmatrix} \boxed{2\downarrow} \\ \hline \boxed{2\downarrow}\, S^{-1} \end{bmatrix} \vec{f}$$

$$\begin{bmatrix} F_M & 0 \\ 0 & F_M \end{bmatrix} \begin{bmatrix} \boxed{2\downarrow} \\ \hline \boxed{2\downarrow}\, S^{-1} \end{bmatrix} \vec{f} = \begin{bmatrix} F_M \vec{f}_{even} \\ \hline F_M \vec{f}_{odd} \end{bmatrix}$$

root of 1

$$\begin{bmatrix} I_M & \overset{D_M}{\cancel{I_M}} \\ I_M & \underset{-D_M}{\cancel{I_M}} \end{bmatrix} \begin{bmatrix} F_M \vec{f}_{even} \\ \hline F_M \vec{f}_{odd} \end{bmatrix} = \cancel{\begin{bmatrix} F_M f_{even} + F_M f_{odd} \\ F_M f_{even} + F_M f_{odd} \end{bmatrix}}$$

$$D_M = \begin{bmatrix} 1 & & & 0 \\ & \omega^{-1} & & \\ & & \omega^{-2} & \\ & & & \ddots \\ 0 & & & & \omega^{-(M-1)} \end{bmatrix}$$

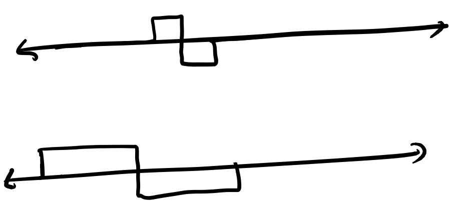
$$\begin{bmatrix} \omega^{-M} & & & \\ & \omega^{-(M+1)} & & \\ & & \ddots & \\ & & & \omega^{-(2M-1)} \end{bmatrix} = \omega^{-M} D_M = -D_M$$

$$F_N = \begin{bmatrix} I_M & D_M \\ I_M & -D_M \end{bmatrix} \begin{bmatrix} F_M & 0 \\ 0 & F_M \end{bmatrix} \begin{bmatrix} \boxed{2\downarrow} \\ \hline \boxed{2\downarrow} S^{-1} \end{bmatrix}$$

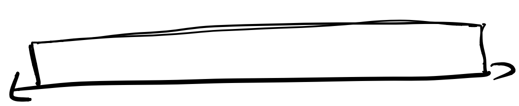
The Haar Wavelet Transform

Assume : $N = 2^n$

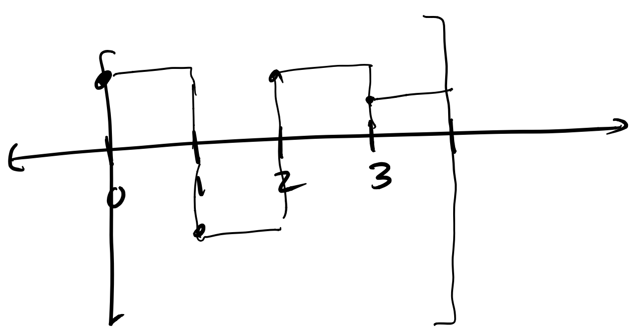
Basic wavelets "pulses"

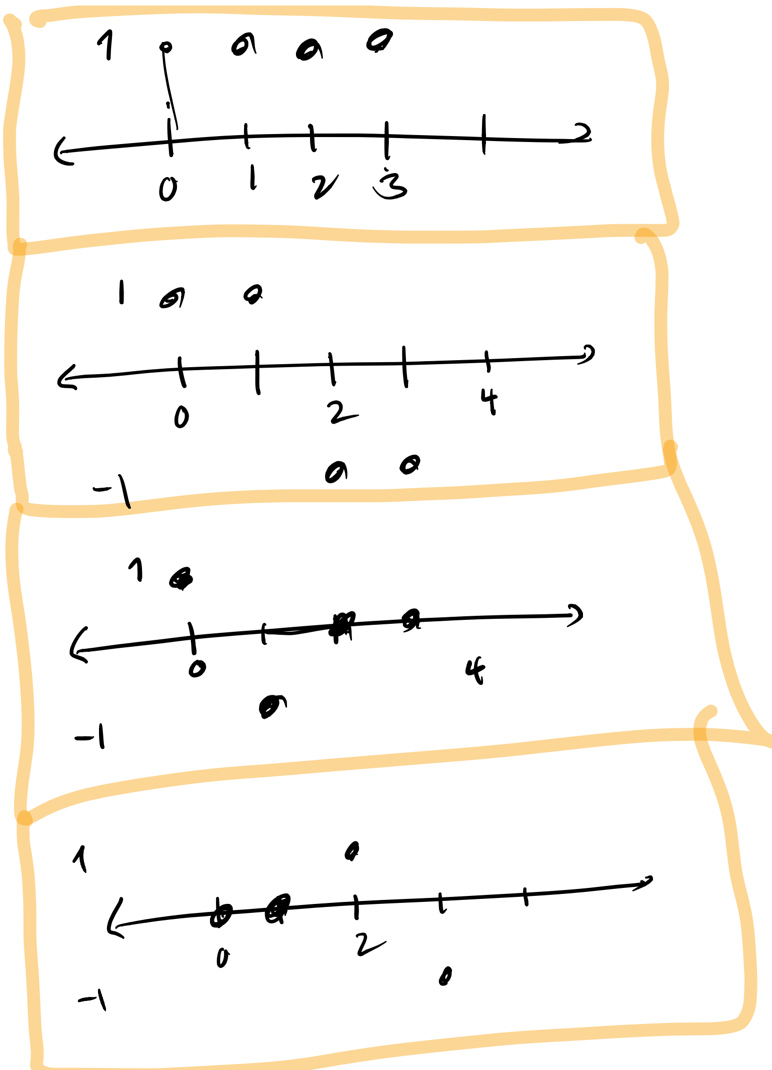


(+ constant pulse



$N = 4$





Idea of working in this basis: Inductive procedure

At each iteration, break up signal into 2 pieces "trend" & "fluctuation"

s_n = original signal n – number that $N = 2^n$.

trend : $s_{n-1}[k] =$ average of s_n at $2k$ & $2k+1$.

$$= \frac{1}{2}\left(s_n[2k] + s_n[2k+1]\right)$$

loase information

"fluctuation"

$$= \frac{1}{2}\left(\underbrace{\boxed{2\downarrow} s_n[k]}_{s_n^{even}[k]} + \underbrace{\boxed{2\downarrow} S^{-1} s_n[k]}_{s_n^{odd}[k]}\right)$$

$$d_{n-1}[k] = s_n^{even}[k] - s_{n-1}[k]$$

given s_{n-1}, d_{n-1}, can get back s_n

$$s_{n-1}[k] + d_{n-1}[k] = s_n^{even}[k]$$

$$s_{n-1}[k] = \frac{1}{2}\left(s_n^{even}[k] + s_n^{odd}[k]\right)$$

$$2 s_{n-1}[k] - s_n^{even}[k] = s_n^{odd}[k]$$

$$s_n \rightsquigarrow \begin{bmatrix} s_{n-1} \\ d_{n-1} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{1}{2}\left(\boxed{2\downarrow} + \boxed{2\downarrow} S^{-1}\right) \\ \hline \boxed{2\downarrow} - \frac{1}{2}\left(\boxed{2\downarrow} + \boxed{2\downarrow} S^{-1}\right) \end{bmatrix}} \begin{bmatrix} s_n \end{bmatrix}$$

$$\begin{bmatrix} s_n^{even} \\ \hline s_n^{odd} - s_n^{even} \end{bmatrix} = \begin{bmatrix} I & 0 \\ -I & I \end{bmatrix} \underbrace{\begin{bmatrix} \boxed{2\downarrow} \\ \hline \boxed{2\downarrow} S^{-1} \end{bmatrix}}_{= \boxed{\text{split}}}$$

$$\cdots\cdots \quad \begin{bmatrix} s_n^{even} \\ s_n^{odd} - s_n^{even} \end{bmatrix}$$

and get

$$\begin{bmatrix} s_{n-1} \\ d_{n-1} \end{bmatrix} = D\,U\,P\,\boxed{\text{split}}\,\begin{bmatrix} s_n \end{bmatrix}$$

$$\begin{bmatrix} I & 0 \\ -I & I \end{bmatrix}$$

$$\begin{bmatrix} I & 1/2\,I \\ 0 & I \end{bmatrix}$$

$$\begin{bmatrix} I & 0 \\ 0 & \frac{1}{2}I \end{bmatrix}$$

P – prediction

U – update

D – normalization.

Continue, apply procedure to $s_{n-1} \rightsquigarrow s_{n-2}, d_{n-2}$

Ex: $N = 4 = 2^2$

s_2

0	1	2	3
4	2	7	6

s_1

0	1
3	6.5

avg of 4,2

avg of 7,6

d_1

0	1
1	0.5

4-3

7-6.5

s_0

0
4.75

avg of 3,6.5

d_0

0
-1.75

3-4.75

"answer" $s_0, \partial_0, \partial_1$

const signal

right left diff.

var between
2 pts on left
& 2 pts on right