Academic year
2025-2026

Faculty of Applied Engineering

Digital Signal Processing

Signal Processing Systems – Formula collection

Walter Daems

Master of Science in Electronics and ICT Engineering Technology
Master of Science in de industriële wetenschappen: elektronica-ICT

2210FTIESY I-Electronic Systems
2212FTIESY I-Elektronische Systemen

This document has been typeset using LaTeX and the `uantwerpendocs` package.
Calculations have been performed using Matlab/Octave.
Graphics have been composed using PGF, TiKZ and InkScape.
All this material has been prepared on a GNU/Linux workstation.

Typesetting of this document was enabled by:

Digital Manifold Waves
Independent
Publishing
Cooperation

DSP-SPS-2025-3.11

Collection 1

System architectures

Single-buffer setup

$$CPLF = \frac{f_s}{M} \cdot \left(1 - p + \frac{p}{Q}\right) \sum_{i=1}^{V} T_i$$

$$R = 3M + \sum_{i=1}^{V} \underbrace{(M + R_{i,extra})}_{\equiv R_i}$$

$$L = 2 \cdot M \cdot T_s = 2\frac{M}{f_s}$$

Double-buffer setup

$$CPLF = \frac{f_s}{M} \cdot \left(1 - p + \frac{p}{Q}\right) \sum_{i=1}^{V} T_i$$

$$R = 2M + \sum_{i=1}^{V} \underbrace{(2M + R_{i,extra})}_{\equiv R_i}$$

$$L = (1 + V) \cdot M \cdot T_s = (1 + V)\frac{M}{f_s}$$

Mixed-buffer setup see single-buffer setup

Collection 2

Convolution and FFT-convolution

2.1 Convolution

Generic discrete-time convolution Consider two signals $x[n]$ and $y[n]$. The convolution of these two signals results in a signal $z[n]$ defined as

$$z[n] = x[n] \star y[n] = \sum_{i=-\infty}^{+\infty} x[n-i]y[i]$$

Discrete-time convolution with a time-limited-signal Consider a time-unlimited signal $x[n]$ in combination with a time-limited signal $h[n]$ with length N, starting at $n = 0$ (i.e. causal). The convolution $x[n] \star h[n]$ is a signal $y[n]$ defined as:

$$y[n] = x[n] \star h[n] = \sum_{i=0}^{N-1} x[n-i]h[i]$$

or

$$= \sum_{i=n-N+1}^{n} h[n-i]x[i]$$

Discrete-time convolution of two time-limited signals Consider two time-limited signals $x[n]$ and $h[n]$ with length M and N respectively, both starting at $n = 0$. The convolution $x[n] \star h[n]$ is a signal $y[n]$ of length $N + M - 1$, defined as:

$$y[n] = x[n] \star h[n] = \sum_{i=\max(0,n-M+1)}^{\min(N-1,n)} x[n-i]h[i]$$

or

$$= \sum_{i=\max(n-N+1,0)}^{\min(n,M-1)} h[n-i]x[i]$$

Domain	Type	Buffer strategy	Method	Buffer style	Total	Per operation			Per sample
					Latency (L)	# Mem cells (R_i)	# COPs	# MACs	# MACs
Time	incremental	-	I-side	-	1	$2N+2$	1	N	N
	incremental	-	O-side	-	1	$2N+2$	1	N	N
Time	frame-based	single	OA	I	$2M$	$K+N$	M	MN	N
				O	$2M$	$K+N$	0	MN	N
			OS	I	$2M$	$K+N$	0	MN	N
				O	$2M$	$K+N$	M	MN	N
		double	OA	I	$(1+V)M$	$2K+1$	M	MN	N
				O	$(1+V)M$	$2K+1$	0	MN	N
			OS	I	$(1+V)M$	$2K+1$	0	MN	N
				O	$(1+V)M$	$2K+1$	M	MN	N
Frequency	frame-based	single	OA	I	$2M$	$3K-M$	K	$2K(1+2\log_2 K)+N-1$	$\leftarrow /M$
				O	$2M$	$3K-M$	K	$2K(1+2\log_2 K)+N-1$	$\leftarrow /M$
			OS	I	$2M$	$3K-M$	$K+N-1$	$2K(1+2\log_2 K)$	$\leftarrow /M$
				O	$2M$	$3K-M$	$K+N-1$	$2K(1+2\log_2 K)$	$\leftarrow /M$
		double	OA	I	$(1+V)M$	$3K$	$K-(N-1)$	$2K(1+2\log_2 K)+N-1$	$\leftarrow /M$
				O	$(1+V)M$	$3K$	$K-(N-1)$	$2K(1+2\log_2 K)+N-1$	$\leftarrow /M$
			OS	I	$(1+V)M$	$3K$	K	$2K(1+2\log_2 K)$	$\leftarrow /M$
				O	$(1+V)M$	$3K$	K	$2K(1+2\log_2 K)$	$\leftarrow /M$

with: $K = M + N - 1$

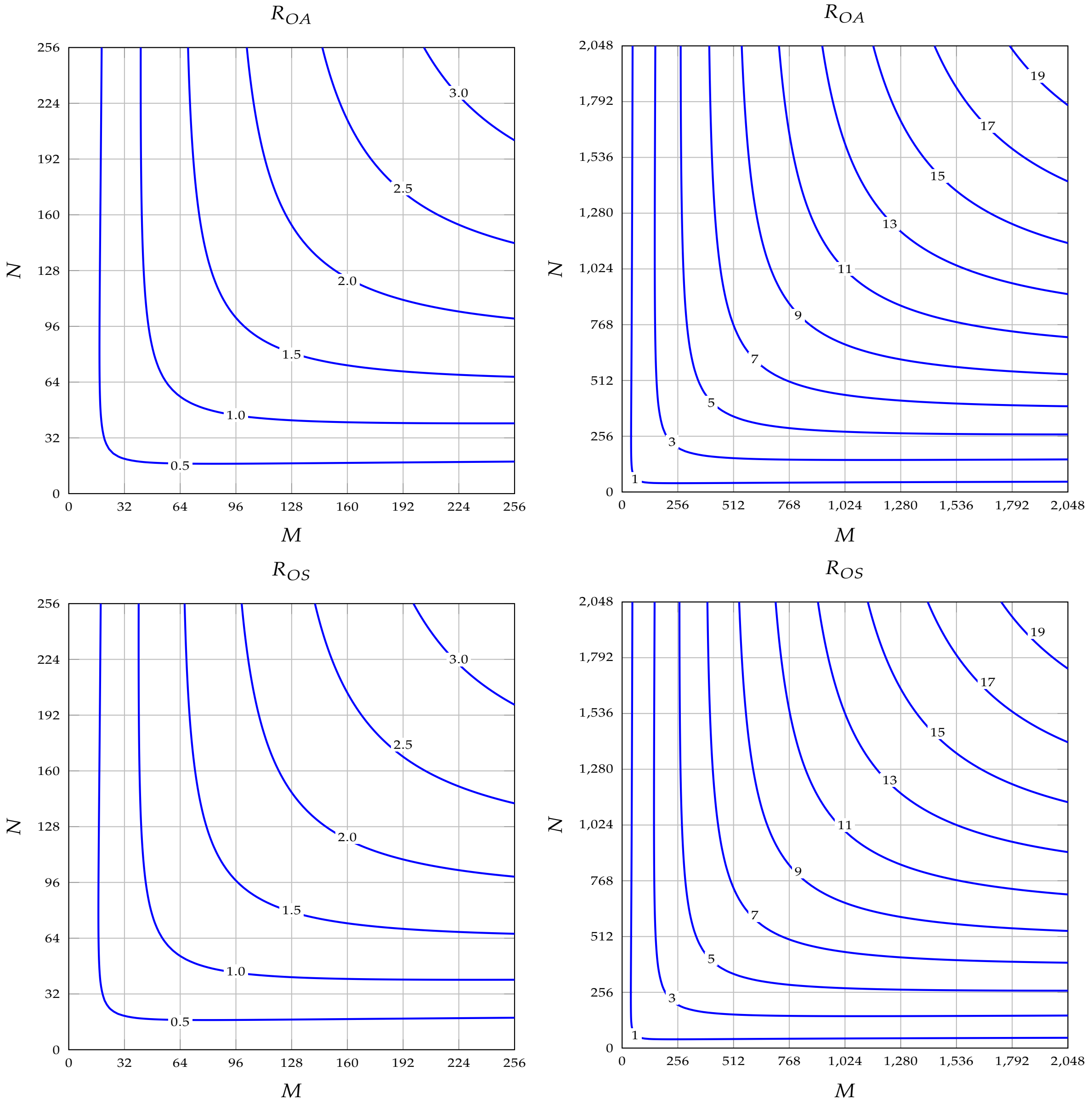
Legend

Symbol	Meaning	Abbrev.	Meaning	Abbrev.	Meaning
M	the frame buffer size	OA	overlap add	MAC	multiply-accumulate operation
N	convolution kernel length	OS	overlap–save	COP	copy operation
V	total number of operations	I	input		
		O	output		

Table 2.1: Computational comparison of the different convolution implementations

Overlap–add

$$R_{OA} = \frac{T_{i,t-dom,OA,sample}}{T_{i,f-dom,OA,sample}} = \frac{MN}{2K(1+2\log_2 K)+N-1}$$



Overlap–save

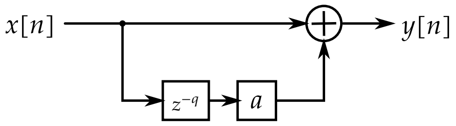
$$R_{OS} = \frac{T_{i,t-dom,OS,sample}}{T_{i,f-dom,OS,sample}} = \frac{MN}{2K(1+2\log_2 K)}$$

with: $K = M + N - 1$

Collection 3

FIR Filter Design

3.1 Zero-placement - Feedforward comb filters



- $-1 < a < 1$:

$$a = \frac{R-1}{R+1}$$

- $|a| > 1$:

$$a = \frac{R+1}{R-1}$$

3.2 Impulse response invariance method

Approximate $h(t)$ by:

$$\tilde{h}[n] = h(nT_s) \cdot w[n]$$

with $w[n]$ an appropriate window function.

3.3 Frequency sampling design method

1. Consider the spectrum from $\omega = 0$ to $\omega = \omega_s$. Keep in mind that the spectrum is periodic.
2. sample the (frequency-domain) magnitude spectrum using $N+1$ equidistant points, putting the extreme points at 0 and $+\omega_s$; assume the phase spectrum to be zero
3. calculate the corresponding time-domain impulse response using the iFFT
4. move the samples at $n \geq N/2$ to $n - N$ to make the impulse response center-symmetric
5. shift the impulse response to make it causal
6. check the result using the FFT
7. finally check the result using a zero-padded FFT

8. apply a window function to the resulting impulse response
9. check the result using the FFT
10. finally check the result using a zero-padded FFT

3.4 Optimal linear-phase filter design

	FIR symmetry	
FIR Length	Symmetrical	Antisymmetrical
Odd	I	III
Even	II	IV

3.4.1 Least-squares method

Minimize

$$\epsilon_2 = \|W(\tilde{\omega})\,(|H(\tilde{\omega})| - G(\tilde{\omega}))\|_2$$

W is a center-symmetrical weighting function defined on $[-\pi, +\pi]$.

3.4.1.1 Design Method 1: grid approach

Assuming

$$H(\tilde{\omega}) = \sum_{n=0}^{\lfloor K \rfloor} q_n f\left((K-n)\,|\tilde{o}mega|\right)$$

with f respresenting cos for filter types I and II, or sin for filter types III and IV.
Pick a frequency grid $[\tilde{\omega}_1, \tilde{\omega}_2, \ldots, \tilde{\omega}_L]$ (with $\tilde{\omega}_i \geq 0, \forall i$) and solve:

$$\boldsymbol{A}X = \boldsymbol{B}$$

with for type I

$$\boldsymbol{A} = \begin{bmatrix} f(K\tilde{\omega}_1) & f((K-1)\tilde{\omega}_1) & f((K-2)\tilde{\omega}_1) & \cdots & f(\tilde{\omega}_1) & f(0) \\ f(K\tilde{\omega}_2) & f((K-1)\tilde{\omega}_2) & f((K-2)\tilde{\omega}_2) & \vdots & f(\tilde{\omega}_2) & f(0) \\ f(K\tilde{\omega}_3) & f((K-1)\tilde{\omega}_3) & f((K-2)\tilde{\omega}_3) & \cdots & f(\tilde{\omega}_3) & f(0) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f(K\tilde{\omega}_L) & f((K-1)\tilde{\omega}_L) & f((K-2)\tilde{\omega}_L) & \cdots & f(\tilde{\omega}_L) & f(0) \end{bmatrix}$$

and

$$\vec{B} = \begin{bmatrix} G(\tilde{\omega}_1) \\ G(\tilde{\omega}_2) \\ G(\tilde{\omega}_3) \\ \vdots \\ G(\tilde{\omega}_L) \end{bmatrix} \qquad \vec{X} \qquad = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_K \end{bmatrix}$$

for type III

$$\boldsymbol{A} = \begin{bmatrix} f(K\tilde{\omega}_1) & f((K-1)\tilde{\omega}_1) & f((K-2)\tilde{\omega}_1) & \cdots & f(\tilde{\omega}_1) \\ f(K\tilde{\omega}_2) & f((K-1)\tilde{\omega}_2) & f((K-2)\tilde{\omega}_2) & \vdots & f(\tilde{\omega}_2) \\ f(K\tilde{\omega}_3) & f((K-1)\tilde{\omega}_3) & f((K-2)\tilde{\omega}_3) & \cdots & f(\tilde{\omega}_3) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ f(K\tilde{\omega}_L) & f((K-1)\tilde{\omega}_L) & f((K-2)\tilde{\omega}_L) & \cdots & f(\tilde{\omega}_L) \end{bmatrix}$$

and

$$\vec{B} = \begin{bmatrix} G(\tilde{\omega}_1) \\ G(\tilde{\omega}_2) \\ G(\tilde{\omega}_3) \\ \vdots \\ G(\tilde{\omega}_L) \end{bmatrix} \qquad \vec{X} \qquad = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_{K-1} \end{bmatrix}$$

and $q_K = 0$. for type II and IV

$$A = \begin{bmatrix} f(K\tilde{\omega}_1) & f((K-1)\tilde{\omega}_1) & f((K-2)\tilde{\omega}_1) & \cdots & f(\tilde{\omega}_1/2) \\ f(K\tilde{\omega}_2) & f((K-1)\tilde{\omega}_2) & f((K-2)\tilde{\omega}_2) & \vdots & f(\tilde{\omega}_2/2) \\ f(K\tilde{\omega}_3) & f((K-1)\tilde{\omega}_3) & f((K-2)\tilde{\omega}_3) & \cdots & f(\tilde{\omega}_3/2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ f(K\tilde{\omega}_L) & f((K-1)\tilde{\omega}_L) & f((K-2)\tilde{\omega}_L) & \cdots & f(\tilde{\omega}_L/2) \end{bmatrix}$$

and

$$\vec{B} = \begin{bmatrix} G(\tilde{\omega}_1) \\ G(\tilde{\omega}_2) \\ G(\tilde{\omega}_3) \\ \vdots \\ G(\tilde{\omega}_L) \end{bmatrix} \qquad \vec{X} \qquad = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_{K-1/2} \end{bmatrix}$$

3.4.1.2 Design Method 2: integral approach

$$A\vec{Q} = \vec{B}$$

with

$$\vec{B} = \begin{bmatrix} b(0) \\ b(1) \\ b(2) \\ \vdots \\ b(\lfloor K \rfloor) \end{bmatrix} \qquad \vec{Q} \qquad = \begin{bmatrix} q(0) \\ q(1) \\ q(2) \\ \vdots \\ q(\lfloor K \rfloor) \end{bmatrix}$$

$$a(m) = \frac{1}{2\pi} \int_{-\pi}^{+\pi} W(\tilde{\omega}) \cos(m\tilde{\omega}) \, \mathrm{d}\tilde{\omega}$$

$$A = \frac{A_1 \pm A_2}{2}$$

with:

$$A_1 = \begin{bmatrix} a(0) & a(1) & a(2) & \cdots & a(\lfloor K \rfloor) \\ a(1) & a(0) & a(1) & \cdots & a(\lfloor K \rfloor - 1) \\ a(2) & a(1) & a(0) & \cdots & a(\lfloor K \rfloor - 2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a(\lfloor K \rfloor) & a(\lfloor K \rfloor - 1) & a(\lfloor K \rfloor - 2) & \cdots & a(0) \end{bmatrix}$$

$$A_2 = \begin{bmatrix} a(2K-0) & a(2K-1) & a(2K-2) & \cdots & a(2K - \lfloor K \rfloor) \\ a(2K-1) & a(2K-2) & a(2K-3) & \cdots & a(2K - \lfloor K \rfloor + 1) \\ a(2K-2) & a(2K-3) & a(2K-4) & \cdots & a(2K - \lfloor K \rfloor + 2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a(2K - \lfloor K \rfloor) & a(2K - \lfloor K \rfloor + 1) & a(2K - \lfloor K \rfloor + 2) & \cdots & a(2K - 2\lfloor K \rfloor) \end{bmatrix}$$

Special property of $a(m)$

$$a(m) \xrightarrow{\text{DtFT}} W(\tilde{\omega})$$

Special property of $b(k)$

$$b(k) = \frac{1}{2\pi}\int_{-\pi}^{+\pi} W(\tilde{\omega})G(\tilde{\omega})f\left((K-k)\,|\tilde{\omega}|\right)\mathrm{d}\tilde{\omega}$$

For type I and II filters:

$$b(k) \xrightarrow{\text{DtFT}} W(\tilde{\omega})G(\tilde{\omega})$$

For type III and IV filters, use the generic formula above.

3.4.2 Parks-McClellan / Remez-exchange algorithm

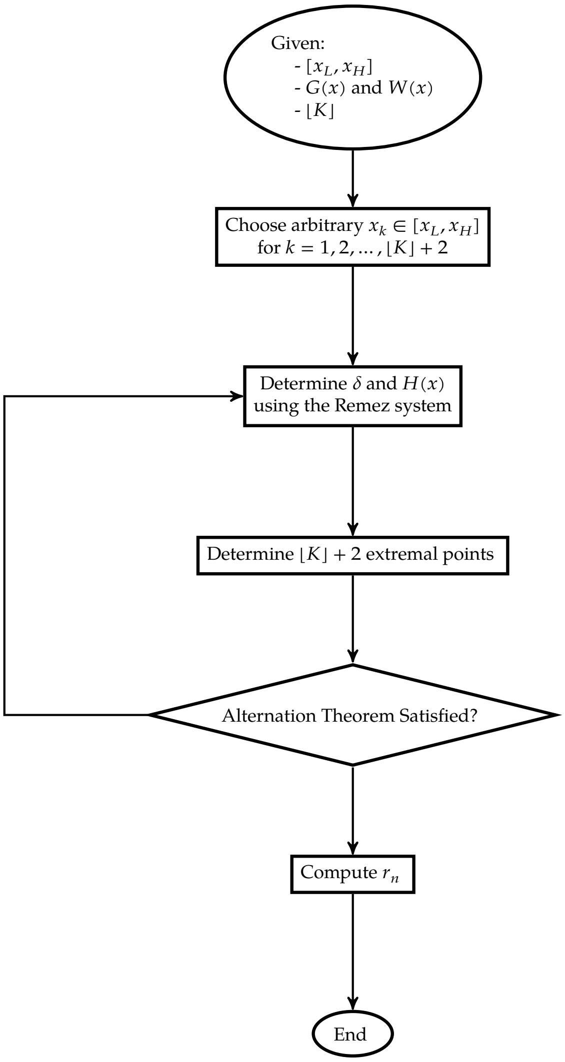
Filter objective Minimize

$$\epsilon_\infty = \left\| W(\tilde{\omega})\left(|H(\tilde{\omega})| - G(\tilde{\omega})\right)\right\|_\infty$$

The Remez-exchange algorithm

$$|H(x)| = \sum_{n=0}^{\lfloor K\rfloor} r_n x^n$$

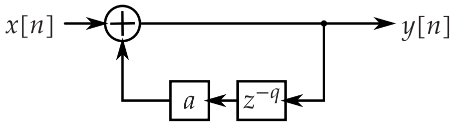
$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{\lfloor K\rfloor} & (-1)^1/W(x_1) \\ 1 & x_2 & x_2^2 & \cdots & x_2^{\lfloor K\rfloor} & (-1)^2/W(x_2) \\ 1 & x_3 & x_3^2 & \cdots & x_3^{\lfloor K\rfloor} & (-1)^3/W(x_3) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_{\lfloor K\rfloor+2} & x_{\lfloor K\rfloor+2}^2 & \cdots & x_{\lfloor K\rfloor+2}^{\lfloor K\rfloor} & (-1)^{\lfloor K\rfloor+2}/W(x_{\lfloor K\rfloor+2}) \end{bmatrix} \begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ \vdots \\ r_{\lfloor K\rfloor} \\ \delta \end{bmatrix} = \begin{bmatrix} |G(x_1)| \\ |G(x_2)| \\ |G(x_3)| \\ \vdots \\ |G(x_{\lfloor K\rfloor+2})| \end{bmatrix}$$


Given:
- $[x_L, x_H]$
- $G(x)$ and $W(x)$
- $\lfloor K \rfloor$
Choose arbitrary $x_k \in [x_L, x_H]$ for $k = 1, 2, \ldots, \lfloor K \rfloor + 2$
Determine δ and $H(x)$ using the Remez system
Determine $\lfloor K \rfloor + 2$ extremal points
Alternation Theorem Satisfied?
Compute r_n
End

Collection 4

IIR Filter Design

4.1 Zero-placement - Feedback comb filters



- $-1 < a < 1$:

$$a = \frac{R-1}{R+1}$$

- $|a| > 1$: does not lead to stable filter designs.

4.2 Impulse invariance method

1. determine the analog filter's transfer function $H(s)$
2. decompose $H(s)$ using partial fraction decomposition
3. apply the inverse Laplace transform to obtain the continuous-time impulse $h(t)$
4. sample $h(t)$ to obtain a discrete-time impulse response $h[n]$
5. convert $h[n]$ to $H(z)$ using the Z-transform.
6. gather the complex-conjugate and real poles and zeros poles in biquad sections
7. make a parallel connection of all the biquad sections (using an extra summation node at the output) to obtain the desired filter

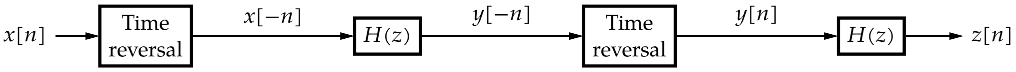
4.3 Bilinear transformation

$$H_d(z) = H_c\left(s \mapsto \frac{2}{T_s}\frac{z-1}{z+1}\right)$$

$$\omega_c = \frac{2}{T_s}\tan\left(\frac{\omega_d T_s}{2}\right)$$

$$\omega_d = \frac{2}{T_s}\arctan\left(\frac{\omega_c T_s}{2}\right)$$

4.4 Reverse filtering



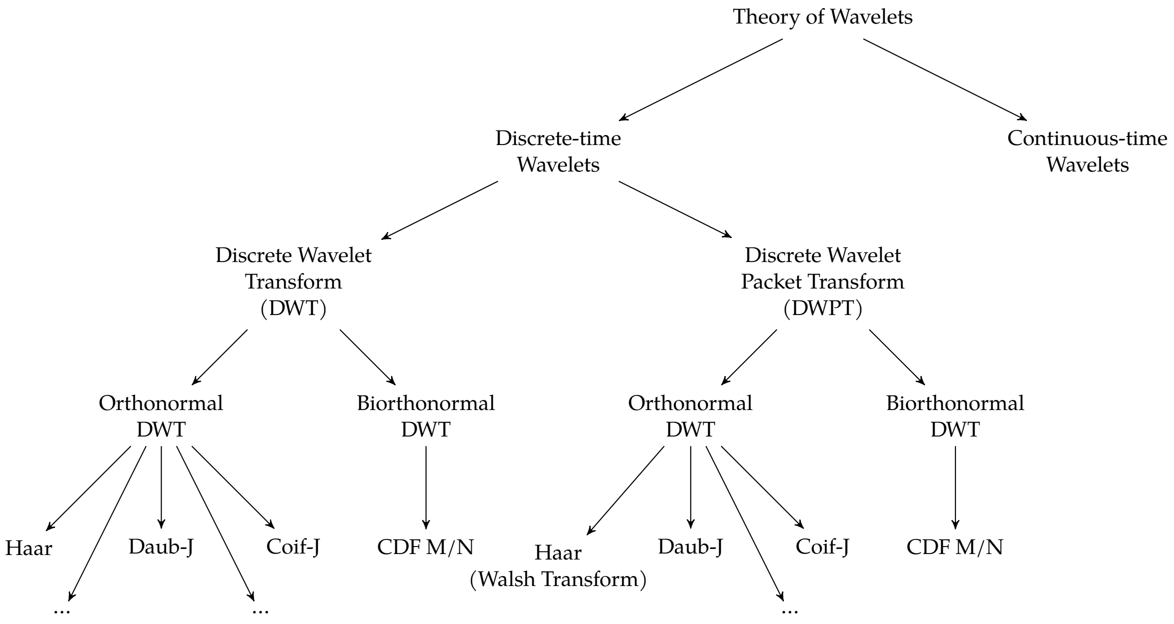
results in:

$$X_p(\omega_1)\left|H(\mathrm{e}^{\mathrm{j}\omega_1})\right|^2$$

Collection 5

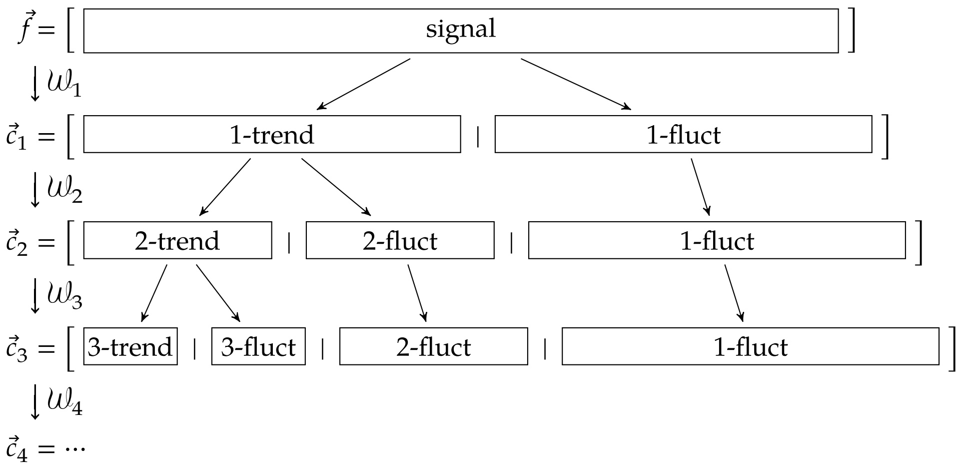
Signal Transforms — Wavelets

5.1 The ball park



5.2 Wavelet transforms

$$\begin{tikzcd}
\vec{f} \arrow[r, "\mathcal{W}_1"] \arrow[r, bend right, "{}_1\mathcal{W}"'] \arrow[rr, bend right, "{}_2\mathcal{W}"'] \arrow[rrr, bend right, "{}_3\mathcal{W}"'] \arrow[rrrrr, bend right, "{}_n\mathcal{W}"'] & \vec{c}_1 \arrow[r, "\mathcal{W}_2"] & \vec{c}_2 \arrow[r, "\mathcal{W}_3"] & \vec{c}_3 \arrow[r] & \cdots \arrow[r, "\mathcal{W}_n"] & \vec{c}_n
\end{tikzcd}$$



5.3 Wavelet construction

The scaling signals on level 1 are generically defined by:

$$\begin{aligned}
\vec{s}_{1,0} &= \begin{bmatrix} \alpha_0, & \alpha_1, & \alpha_2, & \alpha_3, & 0, & 0, & 0, & 0, & \dots, & 0, & 0, & \alpha_{\text{-}2}, & \alpha_{\text{-}1} \end{bmatrix}\\
\vec{s}_{1,1} &= \begin{bmatrix} \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1, & \alpha_2, & \alpha_3, & 0, & 0, & \dots, & 0, & 0, & 0, & 0 \end{bmatrix}\\
\vec{s}_{1,2} &= \begin{bmatrix} 0, & 0, & \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1, & \alpha_2, & \alpha_3, & \dots, & 0, & 0, & 0, & 0 \end{bmatrix}\\
&\vdots\\
\vec{s}_{1,M_1-2} &= \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & \dots, & \alpha_0, & \alpha_1, & \alpha_2, & \alpha_3 \end{bmatrix}\\
\vec{s}_{1,M_1-1} &= \begin{bmatrix} \alpha_2, & \alpha_3, & 0, & 0, & 0, & 0, & 0, & 0, & \dots, & \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1 \end{bmatrix}
\end{aligned}$$

Note that the scaling numbers of the first scaling signal are centered on position column zero, such that there's one more scaling number to the right than there are scaling numbers to left. This specific balancing makes sure that higher-order scaling base vectors do not drift to the right.

Similarly, the wavelets on level 1 are defined by:

$$\begin{aligned}
\vec{w}_{M_1} &= \begin{bmatrix} \beta_0, & \beta_1, & \beta_2, & \beta_3, & 0, & 0, & 0, & 0, & \dots, & 0, & 0, & \beta_{\text{-}2}, & \beta_{\text{-}1} \end{bmatrix}\\
\vec{w}_{M_1+1} &= \begin{bmatrix} \beta_{\text{-}2}, & \beta_{\text{-}1}, & \beta_0, & \beta_1, & \beta_2, & \beta_3, & 0, & 0, & \dots, & 0, & 0, & 0, & 0 \end{bmatrix}\\
\vec{w}_{M_1+2} &= \begin{bmatrix} 0, & 0, & \beta_{\text{-}2}, & \beta_{\text{-}1}, & \beta_0, & \beta_1, & \beta_2, & \beta_3, & \dots, & 0, & 0, & 0, & 0 \end{bmatrix}\\
&\vdots\\
\vec{w}_{N-2} &= \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & \dots, & \beta_0, & \beta_1, & \beta_2, & \beta_3 \end{bmatrix}\\
\vec{w}_{N-1} &= \begin{bmatrix} \beta_2, & \beta_3, & 0, & 0, & 0, & 0, & 0, & 0, & \dots, & \beta_{\text{-}2}, & \beta_{\text{-}1}, & \beta_0, & \beta_1 \end{bmatrix}
\end{aligned}$$

Recursion:

$$\begin{aligned}
\vec{s}_{n,i} &= \sum_k \alpha_k \vec{s}_{n-1,2i+k}\\
&= \alpha_{\text{-}2}\vec{s}_{n-1,2i-2} + \alpha_{\text{-}1}\vec{s}_{n-1,2i-1} + \alpha_0\vec{s}_{n-1,2i} + \alpha_1\vec{s}_{n-1,2i+1} + \alpha_2\vec{s}_{n-1,2i+2} + \alpha_3\vec{s}_{n-1,2i+3}\\
\vec{w}_{M_n+i} &= \sum_k \beta_k \vec{s}_{n-1,2i+k}\\
&= \beta_{\text{-}2}\vec{s}_{n-1,2i-2} + \beta_{\text{-}1}\vec{s}_{n-1,2i-1} + \beta_0\vec{s}_{n-1,2i} + \beta_1\vec{s}_{n-1,2i+1} + \beta_2\vec{s}_{n-1,2i+2} + \beta_3\vec{s}_{n-1,2i+3}
\end{aligned}$$

with $i = 0, 1, \dots, M_n - 1$ and

$$s_{0,i}[n] = \delta[n-i]$$

Daubechies-4 wavelets (Daub-4)

$$\begin{aligned}
\beta_{\text{-}1} &= \frac{1-\sqrt{3}}{4\sqrt{2}} & \qquad \alpha_{\text{-}1} &= \frac{1+\sqrt{3}}{4\sqrt{2}} \\
\beta_{0} &= -\frac{3-\sqrt{3}}{4\sqrt{2}} & \qquad \alpha_{0} &= \frac{3+\sqrt{3}}{4\sqrt{2}} \\
\beta_{1} &= \frac{3+\sqrt{3}}{4\sqrt{2}} & \qquad \alpha_{1} &= \frac{3-\sqrt{3}}{4\sqrt{2}} \\
\beta_{2} &= -\frac{1+\sqrt{3}}{4\sqrt{2}} & \qquad \alpha_{2} &= \frac{1-\sqrt{3}}{4\sqrt{2}}
\end{aligned}$$

Daubechies-6 wavelets (Daub-6)

$$\begin{aligned}
\beta_{\text{-}2} &= 0.0352262918857095 & \qquad \alpha_{\text{-}2} &= 0.332670552950083 \\
\beta_{\text{-}1} &= 0.0854412738820267 & \qquad \alpha_{\text{-}1} &= 0.806891509311092 \\
\beta_{0} &= -0.135011020010255 & \qquad \alpha_{0} &= 0.459877502118491 \\
\beta_{1} &= -0.459877502118491 & \qquad \alpha_{1} &= -0.135011020010255 \\
\beta_{2} &= 0.806891509311092 & \qquad \alpha_{2} &= -0.0854412738820267 \\
\beta_{3} &= -0.332670552950083 & \qquad \alpha_{3} &= 0.0352262918857095
\end{aligned}$$

Coifman-6 wavelets (Coif-6)

$$\begin{aligned}
\beta_{\text{-}2} &= -\frac{-3+\sqrt{7}}{16\sqrt{2}} & \qquad \alpha_{\text{-}2} &= \frac{1-\sqrt{7}}{16\sqrt{2}} \\
\beta_{\text{-}1} &= \frac{1-\sqrt{7}}{16\sqrt{2}} & \qquad \alpha_{\text{-}1} &= \frac{5+\sqrt{7}}{16\sqrt{2}} \\
\beta_{0} &= -\frac{14-2\sqrt{7}}{16\sqrt{2}} & \qquad \alpha_{0} &= \frac{14+2\sqrt{7}}{16\sqrt{2}} \\
\beta_{1} &= \frac{14+2\sqrt{7}}{16\sqrt{2}} & \qquad \alpha_{1} &= \frac{14-2\sqrt{7}}{16\sqrt{2}} \\
\beta_{2} &= -\frac{5+\sqrt{7}}{16\sqrt{2}} & \qquad \alpha_{2} &= \frac{1-\sqrt{7}}{16\sqrt{2}} \\
\beta_{3} &= \frac{1-\sqrt{7}}{16\sqrt{2}} & \qquad \alpha_{3} &= \frac{-3+\sqrt{7}}{16\sqrt{2}}
\end{aligned}$$

Cohen-Daubechies-Feauveau 5/3 (CDF 5/3)

$$\begin{aligned}
\vec{s}_{A1,0} &= \begin{bmatrix} \alpha_0, & \alpha_1, & \alpha_2, & 0, & 0, & 0, & \dots, & 0, & 0, & 0, & 0, & \alpha_{\text{-}2}, & \alpha_{\text{-}1} \end{bmatrix} \\
\vec{s}_{A1,1} &= \begin{bmatrix} \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1, & \alpha_2, & 0, & \dots, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix} \\
\vec{s}_{A1,2} &= \begin{bmatrix} 0, & 0, & \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1, & \dots, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix} \\
&\vdots \\
\vec{s}_{A1,M_1-2} &= \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & \dots, & \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1, & \alpha_2, & 0 \end{bmatrix} \\
\vec{s}_{A1,M_1-1} &= \begin{bmatrix} \alpha_2, & 0, & 0, & 0, & 0, & 0, & \dots, & 0, & 0, & \alpha_{\text{-}2}, & \alpha_{\text{-}1}, & \alpha_0, & \alpha_1 \end{bmatrix}
\end{aligned}$$

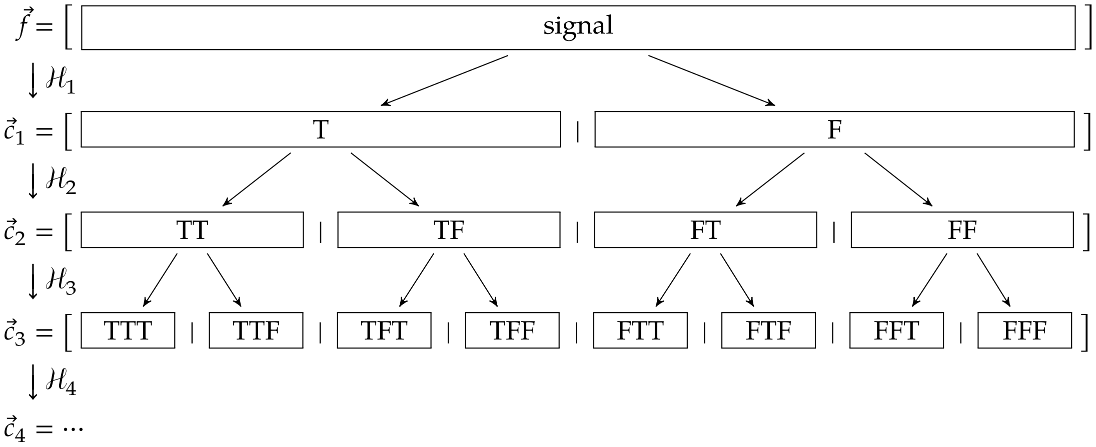
$$\begin{aligned}
\vec{w}_{M_1} &= \begin{bmatrix} \beta_0, & \beta_1, & \beta_2, & 0, & 0, & 0, & \dots, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix} \\
\vec{w}_{AM_1+1} &= \begin{bmatrix} 0, & 0, & \beta_0, & \beta_1, & \beta_2, & 0, & \dots, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix} \\
\vec{w}_{AM_1+2} &= \begin{bmatrix} 0, & 0, & 0, & 0, & \beta_0, & \beta_1, & \dots, & 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix} \\
&\vdots \\
\vec{w}_{AN-2} &= \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & \dots, & 0, & 0, & \beta_0, & \beta_1, & \beta_2, & 0 \end{bmatrix} \\
\vec{w}_{AN-1} &= \begin{bmatrix} \beta_2, & 0, & 0, & 0, & 0, & 0, & \dots, & 0, & 0, & 0, & 0, & \beta_0, & \beta_1 \end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
\alpha_{-2} &= -\sqrt{2}\frac{1}{8} \\
\alpha_{-1} &= \sqrt{2}\frac{1}{4} & \beta_0 &= \sqrt{2}\frac{1}{4} \\
\alpha_0 &= \sqrt{2}\frac{3}{4} & \beta_1 &= -\sqrt{2}\frac{1}{2} \\
\alpha_1 &= \sqrt{2}\frac{1}{4} & \beta_2 &= \sqrt{2}\frac{1}{4} \\
\alpha_2 &= -\sqrt{2}\frac{1}{8}
\end{aligned}$$

Cohen-Daubechies-Feauveau 9/7 (CDF 9/7)

$$\begin{aligned}
\alpha_{-4} &= 0.037828456 \\
\alpha_{-3} &= -0.023849465 & \beta_{-2} &= -0.064538887 \\
\alpha_{-2} &= -0.110624404 & \beta_{-1} &= 0.040689418 \\
\alpha_{-1} &= 0.377402856 & \beta_0 &= 0.418092273 \\
\alpha_0 &= 0.852698679 & \beta_1 &= -0.788485616 \\
\alpha_1 &= 0.377402856 & \beta_2 &= 0.418092273 \\
\alpha_2 &= -0.110624404 & \beta_3 &= 0.040689418 \\
\alpha_3 &= -0.023849465 & \beta_4 &= -0.064538883 \\
\alpha_4 &= 0.037828456
\end{aligned}$$

5.4 Wavelet packet transforms



5.5 Two-dimensional wavelet transforms

$$\vec{f} = \left[\begin{array}{|c|}\hline \quad\text{signal}\quad \\ \hline\end{array}\right] \xrightarrow{\omega_1} \vec{c}_1 = \left[\begin{array}{|c|c|}\hline \text{1-t} & \text{1-h} \\ \hline \text{1-v} & \text{1-d} \\ \hline\end{array}\right]$$

5.6 Signal compression

Shannon's source coding theorem for finite-length discrete-time signals
A signal $\vec{f}$ consisting of N samples, quantized into levels x_i requires at least $N \cdot H(\vec{f})$ bits to be encoded without losing information.

Definition: signal entropy
The entropy H of a discrete-time signal $\vec{f}$, quantized into levels x_i, with relative occurrence frequencies denoted by $p(x_i)$ equals:

$$H(\vec{f}) = \sum_i p(x_i) \log_2 \left(\frac{1}{p(x_i)} \right)$$

Scalar product If $f[n] = [\ldots, f_i, \ldots]$, and $g[n] = [\ldots, g_i, \ldots]$, then

$$\langle \vec{f}, \vec{g} \rangle = \sum_{i=-\infty}^{+\infty} f_i \overline{g_i}$$

Orthogonality

$$\vec{f} \perp \vec{g} \quad \Leftrightarrow \quad \langle \vec{f}, \vec{g} \rangle = 0$$

Norm

$$\|\vec{f}\| = \sqrt{\langle \vec{f}, \vec{f} \rangle}$$

Distance

$$d(\vec{f}, \vec{g}) = \|\vec{f} - \vec{g}\|$$

5.6.1 Decomposition of vectors in terms of the base vectors

$$\vec{f} = c_1 \vec{e}_1 + c_2 \vec{e}_2 + c_3 \vec{e}_3 + \ldots$$

with

$$c_i = \frac{1}{\|\vec{e}_i\|^2} \langle \vec{f}, \vec{e}_i \rangle = \frac{\langle \vec{f}, \vec{e}_i \rangle}{\langle \vec{e}_i, \vec{e}_i \rangle}$$

5.6.2 Parseval's identity

$$\|\vec{f}\|^2 = |c_1|^2 \|\vec{e}_1\|^2 + |c_2|^2 \|\vec{e}_2\|^2 + |c_3|^2 \|\vec{e}_3\|^2 + \cdots$$

Scalar product If $f[n] = [\dots, f_i, \dots]$, and $g[n] = [\dots, g_i, \dots]$, then

$$\left\langle \vec{f}, \vec{g} \right\rangle = \sum_{i=-\infty}^{+\infty} f_i \overline{g_i}$$

Orthogonality

$$\vec{f} \perp \vec{g} \quad \Leftrightarrow \quad \left\langle \vec{f}, \vec{g} \right\rangle = 0$$

Norm

$$\left\| \vec{f} \right\| = \sqrt{\left\langle \vec{f}, \vec{f} \right\rangle}$$

Distance

$$d(\vec{f}, \vec{g}) = \left\| \vec{f} - \vec{g} \right\|$$

5.6.3 Decomposition of vectors in terms of the base vectors

$$\vec{f} = c_1 \vec{e}_1 + c_2 \vec{e}_2 + c_3 \vec{e}_3 + \dots$$

with

$$c_i = \frac{1}{\left\| \vec{e}_i \right\|^2} \left\langle \vec{f}, \vec{e}_i \right\rangle = \frac{\left\langle \vec{f}, \vec{e}_i \right\rangle}{\left\langle \vec{e}_i, \vec{e}_i \right\rangle}$$

5.6.4 Parseval's identity

$$\left\| \vec{f} \right\|^2 = |c_1|^2 \left\| \vec{e}_1 \right\|^2 + |c_2|^2 \left\| \vec{e}_2 \right\|^2 + |c_3|^2 \left\| \vec{e}_3 \right\|^2 + \cdots$$