



UiO • **University of Oslo**

Uke 10: Diskret Fourier Transform, II

Jo Inge Buskenes

Institutt for informatikk, Universitetet i Oslo

INF3470/4470, høst 2012



Dagens temaer

Spektral glatting pga endelig lengde data

Bruk av DFT'en

Frekvensestimering

FFT

Tema

Spektral glatting pga endelig lengde data

Spektral glatting

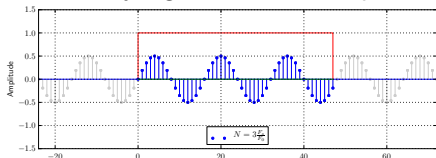
Diskret tid vinduer

Endelig lengde data

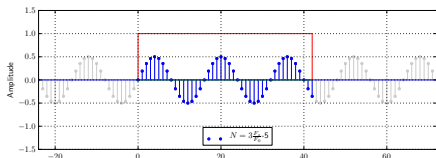
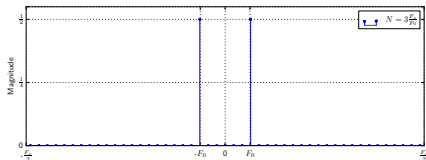
- ▶ Hvis vi definerer et vindu $w[n] = \begin{cases} 1, & n \in 0, 1, \dots, N-1 \\ 0 & \text{ellers} \end{cases}$
- ▶ og et periodisk signal $x_p[n], \quad n \in -\infty \dots \infty$
- ▶ så vil produktet av disse $x[n] = x_p[n] w[n]$ beskrive en tidsbegrenset versjon av $x_p[n]$.
- ▶ Tar vi DFT'en av $x[n]$ vil vi kunne få problemer med:
 - ▶ Spektral lekkasje (når $N \neq kT_s, k \in \mathbb{Z}_{\geq 0}$)
 - ▶ Glatting

Endelig lengde data

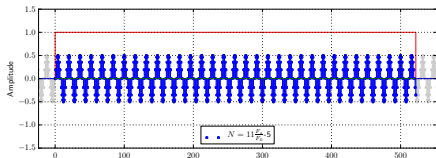
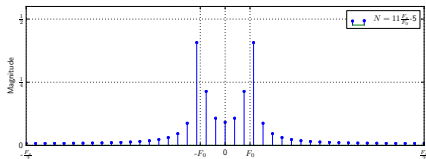
Forskjellige utsnitt fra det periodiske signalet $x_p[n] = 0.5 \sin(2\pi F_0 n)$:



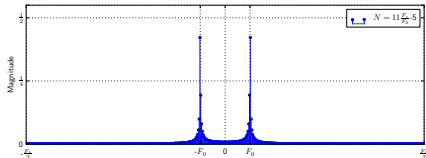
DFT
⇒



DFT
⇒



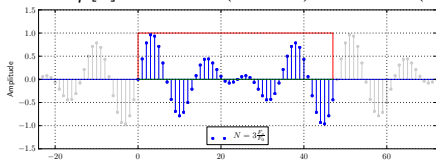
DFT
⇒



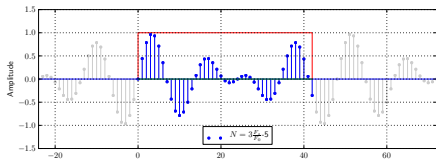
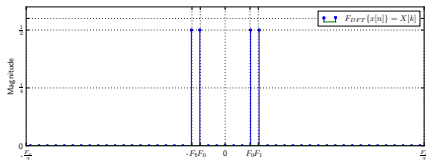
Endelig lengde data

Forskjellige utsnitt fra det periodiske signalet

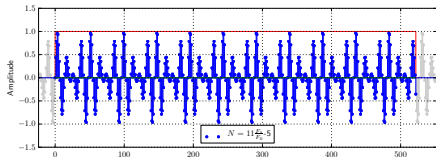
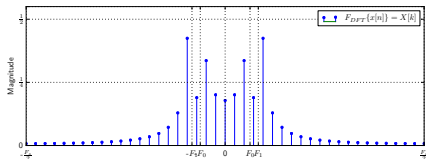
$$x_p[n] = 0.5 \sin(2\pi F_0 n) + 0.5 \sin(2\pi F_1 n):$$



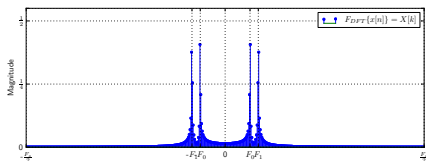
DFT
⇒



DFT
⇒



DFT
⇒

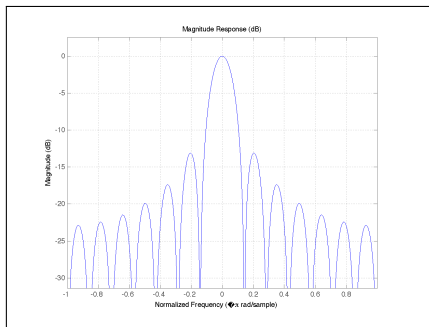


Spekter til en tidsbegrenset sinusfunksjon

- ▶ Et vindu $w[n] = \begin{cases} 1, & n \in 0, 1, \dots, N-1 \\ 0 & \text{ellers} \end{cases}$
- ▶ har DTFT'en $W(\Omega) = \frac{\sin N\Omega/2}{\sin \Omega/2} e^{-j(N-1)\Omega/2}$

```

1 > N = 14;
2 > omega = linspace(-pi,pi,512);
3 > W = (sin(N*omega/2) ./ sin(omega/2)) ...
4     .* exp(-j*(N-1)*omega/2);
5 > plot(omega,real(W));
6
7 > h = fvtool(ones(N,1)/N,1,...
8     'frequencyrange','[-pi, pi)');
9
10 > print -dpng WindowedSin
  
```



Spektral glatting pga endelig lengde data

Spektral glatting

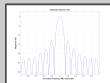
Spekter til en tidsbegrenset sinusfunksjon

- Et vindu $w[n] = \begin{cases} 1, & n \in 0, 1, \dots, N-1 \\ 0 & \text{ellers} \end{cases}$
- har DTFT'en $W(\Omega) = \frac{\sin(N\Omega/2)}{\sin(\Omega/2)} e^{-j(N-1)\Omega/2}$

```

1 N = 10;
2 w = ones(1, N);
3 f = linspace(0, 1, N);
4 W = zeros(1, N);
5 for n = 0:N-1
6     W(n+1) = exp(-j*2*pi*f(n)*N);
7 end
8 plot(w, 'b');
9 title('Vinduet w[n]');
10
11 N = 10;
12 w = ones(1, N);
13 f = linspace(0, 1, N);
14 W = zeros(1, N);
15 for n = 0:N-1
16     W(n+1) = exp(-j*2*pi*f(n)*N);
17 end
18 plot(W, 'b');
19 title('Spekter W(\Omega)');
20
21 N = 10;
22 w = ones(1, N);
23 f = linspace(0, 1, N);
24 W = zeros(1, N);
25 for n = 0:N-1
26     W(n+1) = exp(-j*2*pi*f(n)*N);
27 end
28 plot(W, 'b');
29 title('Spekter W(\Omega)');
30
31 N = 10;
32 w = ones(1, N);
33 f = linspace(0, 1, N);
34 W = zeros(1, N);
35 for n = 0:N-1
36     W(n+1) = exp(-j*2*pi*f(n)*N);
37 end
38 plot(W, 'b');
39 title('Spekter W(\Omega)');
40

```



Fourier transformen til et diskret aperiodisk vindu

- Gitt et diskret aperiodisk vindu $w[n] = \begin{cases} 1, & n \in 0, 1, \dots, N-1 \\ 0, & \text{ellers} \end{cases}$
- Da finner vi frekvensresponsen ved DTFT:

$$\begin{aligned}
 W(\Omega) &= \sum_{k=-\infty}^{\infty} w[n] e^{-j\Omega n} = \sum_{k=0}^{N-1} e^{-j\Omega n} \\
 &= \frac{1 - e^{-j\Omega N}}{1 - e^{-j\Omega}} \\
 &= \frac{(e^{j\frac{\Omega N}{2}} - e^{-j\frac{\Omega N}{2}}) e^{-j\frac{\Omega N}{2}}}{(e^{j\frac{\Omega}{2}} - e^{-j\frac{\Omega}{2}}) e^{-j\frac{\Omega}{2}}} \\
 &= \frac{2j \sin \frac{\Omega N}{2}}{2j \sin \frac{\Omega}{2}} \frac{e^{-j\frac{\Omega N}{2}}}{e^{-j\frac{\Omega}{2}}} \\
 &= \frac{\sin \frac{\Omega N}{2}}{\sin \frac{\Omega}{2}} e^{-j\frac{\Omega(N-1)}{2}}
 \end{aligned}$$

Her gjenkjenner vi amplitude- og fasedelen lett:

$$W(\Omega) = |W(\Omega)| e^{j\Theta_W(\Omega)}$$

$$|W(\Omega)| = \frac{\sin \frac{\Omega N}{2}}{\sin \frac{\Omega}{2}}$$

$$\Theta_W(\Omega) = -\frac{\Omega(N-1)}{2}$$

Spektral glatting pga endelig lengde data

Spektral glatting

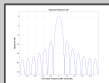
Spekter til en tidsbegrenset sinusfunksjon

- Et vindu $w[n] = \begin{cases} 1, & n \in 0, 1, \dots, N-1 \\ 0 & \text{ellers} \end{cases}$
- har DTFT'en $W(\Omega) = \frac{\sin(N\Omega/2)}{\sin(\Omega/2)} e^{-j(N-1)\Omega/2}$

```

1 N = 10;
2 w = linspace(0,pi,pi, N);
3 N = ceil(N/2);
4 w = linspace(0,pi,pi, N);
5 N = ceil(N/2);
6 N = 10;
7 w = linspace(0,pi,pi, N);
8 N = ceil(N/2);
9 N = 10;
10 w = linspace(0,pi,pi, N);

```

Analyse av $|W(\Omega)|$ for det rektangulære vinduet

- Toppverdien i $\Omega = 0$: $|W(0)| = ?$

$$|W(0)| = \lim_{\Omega \rightarrow 0} \frac{\sin \frac{\Omega N}{2}}{\sin \frac{\Omega}{2}} = \lim_{\Omega \rightarrow 0} \frac{\frac{\Omega N}{2}}{\frac{\Omega}{2}} = N$$

- Avstanden fra $\Omega = 0$ til nullpunktene:

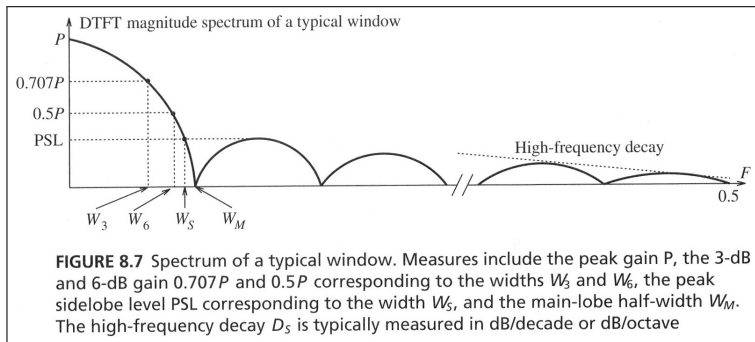
$$|W(\Omega)| = \frac{\sin \frac{\Omega N}{2}}{\sin \frac{\Omega}{2}} = 0 \quad \text{når} \quad \sin \frac{\Omega N}{2} = 0$$

$$\text{dvs.} \quad \frac{\Omega N}{2} = \frac{2\pi FN}{2} = k\pi \quad k \in \mathbb{Z}_{\geq 1}$$

Første nullpunkt ved $\Omega = \frac{1}{N}$. Dette tilsvarer halve hovedlobebredden. Gang med 2 for å få hele.

- Høyden på første sidelobe: Vanskelig å finne en eksplisitt løsning for dette. Bruker en numerisk løsning.

Respos for typisk vindu; kvalitetsmål



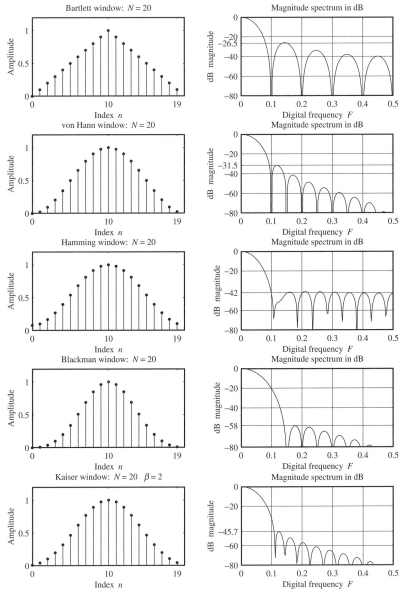
► Kvalitetsmål, vinduer;

- Maks verdi; Peak value (P)
- Maks sidelobe verdi, Peak Sidelobe Level (PSL)
- -3dB bredde (W_3), $\sim \sqrt{2}/2 P$
- -6dB bredde (W_6), $\sim 0.5P$
- Bredde 1. nullpkt (W_M)
- Bredde for å nå max-SL nivå (W_5)

Noen typiske vinduer

- ▶ De forskjellige vinduene gir oss forskjellig kompromiss mellom oppløsning og lekkasje

FIGURE 8.8
Commonly used DFT windows and their spectral characteristics



Noen typiske vinduer ...

Entry	Window	Expression for $w[n]$	$W_M = \frac{K}{N}$	Normalized Peak Sidelobe
1	Boxcar	1	$2/N$	$0.2172 \approx -13.3$ dB
2	Bartlett	$1 - \frac{2 k }{N}$	$4/N$	$0.0472 \approx -26.5$ dB
3	von Hann	$0.5 + 0.5 \cos\left(\frac{2\pi k}{N}\right)$	$4/N$	$0.0267 \approx -31.5$ dB
4	Hamming	$0.54 + 0.46 \cos\left(\frac{2\pi k}{N}\right)$	$4/N$	$0.0073 \approx -42.7$ dB
5	Blackman	$0.42 + 0.5 \cos\left(\frac{2\pi k}{N}\right) + 0.08 \cos\left(\frac{4\pi k}{N}\right)$	$6/N$	$0.0012 \approx -58.1$ dB
6	Kaiser	$\frac{I_0(\pi\beta\sqrt{1 - (2k/N)^2})}{I_0(\pi\beta)}$	$\frac{2\sqrt{1 + \beta^2}}{N}$	$\frac{0.22\pi\beta}{\sinh(\pi\beta)} \approx -45.7$ dB (for $\beta = 2$)

NOTES: $k = 0.5N - n$, where $n = 0, 1, \dots, N - 1$. W_M is the main-lobe width.

For the Kaiser window, $I_0(\cdot)$ is the modified Bessel function of order zero.

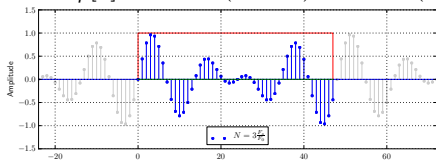
For the Kaiser window, the parameter β controls the peak sidelobe level.

The von Hann window is also known as the *Hanning* window.

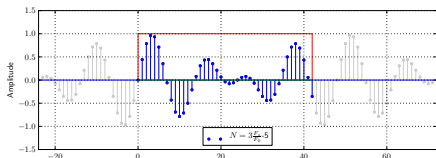
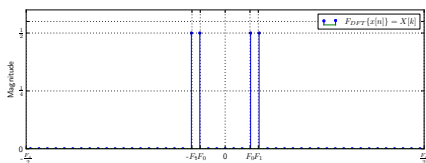
Endelig lengde data - rektangulært vindu

Forskjellige utsnitt fra det periodiske signalet

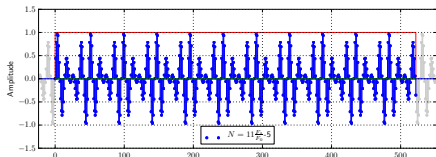
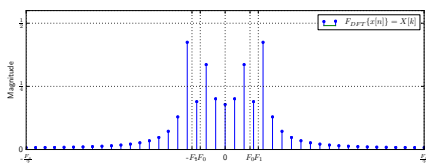
$$x_p[n] = 0.5 \sin(2\pi F_0 n) + 0.5 \sin(2\pi F_1 n):$$



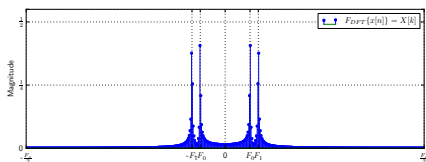
DFT
⇒



DFT
⇒



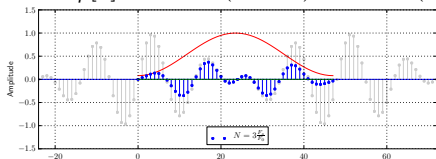
DFT
⇒



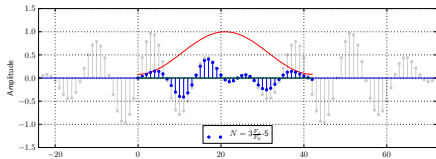
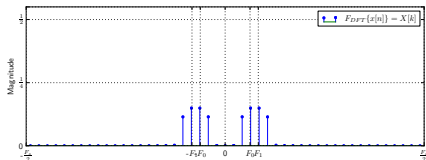
Endelig lengde data - Hamming vindu

Forskjellige utsnitt fra det periodiske signalet

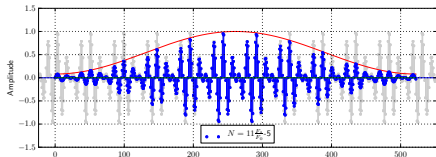
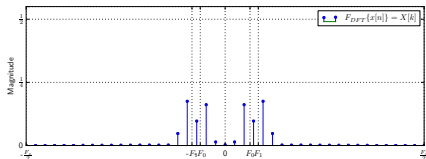
$$x_p[n] = 0.5 \sin(2\pi F_0 n) + 0.5 \sin(2\pi F_1 n):$$



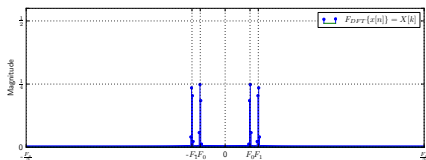
DFT
⇒



DFT
⇒



DFT
⇒



Oppløsning til vinduer

- ▶ Hvor nære kan to frekvenser være uten at vi får problemer med å skille dem?
- ▶ Frekvensoppløsning: $\Delta f = W_M F_s$, der W_M er avhengig av vinduet og angir avstand til første nullpunkt i magnitudespekteret.

Eksempel 8.13:

- ▶ Gitt:

$$x(t) = A_1 \cos 2\pi f_0 t + A_2 \cos 2\pi(f_0 + \Delta f)t$$
$$A_1 = A_2 = 1 \qquad f_0 = 30 \text{ Hz} \qquad F_s = 128 \text{ Hz}$$

- (a) Hva er minste Δf som kan bli oppløst ved bruk av Boxcar og Hanning når:

- i $N = 256$, $N_{\text{FFT}} = 2048$
- ii $N = 512$, $N_{\text{FFT}} = 2048$
- iii $N = 256$, $N_{\text{FFT}} = 4096$

Oppløsning til vinduer

► Svar:

(a) Hva er minste Δf som kan bli oppløst ved bruk av Boxcar og Hanning når:

i $N = 256, N_{\text{FFT}} = 2048$

$$\Rightarrow \text{Rect: } \Delta F = \frac{2}{N} F_s = \frac{2}{256} 128 \text{ Hz} = 1 \text{ Hz}$$

$$\Rightarrow \text{Hann: } \Delta F = \frac{4}{N} F_s = \frac{4}{256} 128 \text{ Hz} = 2 \text{ Hz}$$

ii $N = 512, N_{\text{FFT}} = 2048$

$$\Rightarrow \text{Rect: } \Delta F = \frac{2}{N} F_s = \frac{2}{512} 128 \text{ Hz} = 0.5 \text{ Hz}$$

$$\Rightarrow \text{Hann: } \Delta F = \frac{4}{N} F_s = \frac{4}{512} 128 \text{ Hz} = 1 \text{ Hz}$$

iii $N = 256, N_{\text{FFT}} = 4096$

$$\Rightarrow \text{Rect: } \Delta F = \frac{2}{N} F_s = \frac{2}{256} 128 \text{ Hz} = 1 \text{ Hz}$$

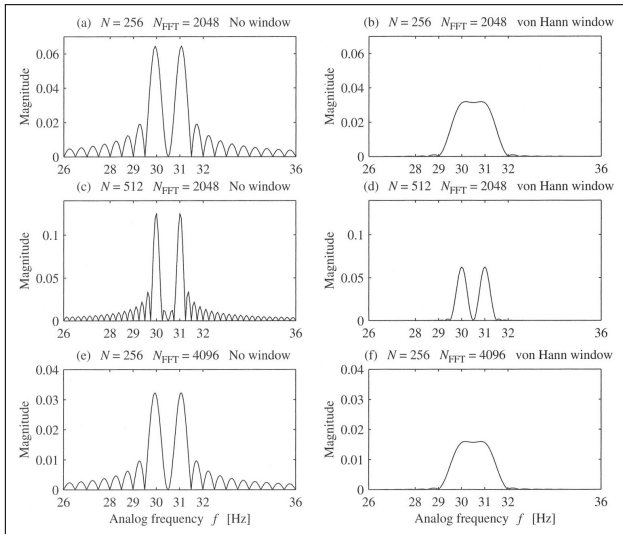
$$\Rightarrow \text{Hann: } \Delta F = \frac{4}{N} F_s = \frac{4}{256} 128 \text{ Hz} = 2 \text{ Hz}$$

- Rektangulært vindu gir best oppløsning, men har størst sidelober / mest lekkasje.
- Bedre oppløsning ved å øke N .
- Null-utvidelse har ingen effekt.
- Hanning gir halvparten av oppløsningen, men mindre lekkasje.

Oppløsning til vinduer

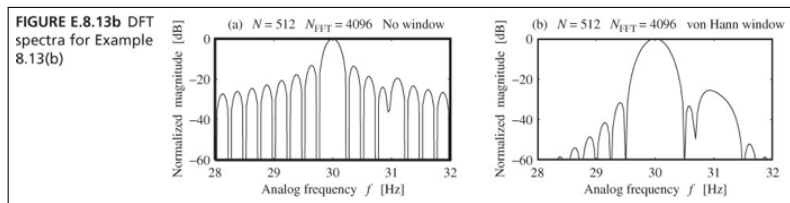
Frekvens oppløsning:

- Kriterie for oppløsning av to nabofrekvenser



Oppløsning til vinduer ...

- ▶ “Dynamisk område” oppløsning



Tema

Bruk av DFT'en

Lineær konvolusjon med lange sekvenser

Bånd-begrenset interpolasjon

Konvolusjon av lange sekvenser

- ▶ Gitt $h[n]$ med lengde N_1 , og $x[n]$ med lengde N_2 .
 1. Nullutvider $h[n]$ og $x[n]$ til lengde $N \geq N_1 + N_2 - 1$.
 2. Finner N -punkts DFT av $h[n]$ og $x[n]$.
 3. Multipliserer sammen $Y[k] = H[k] X[k]$.
 4. Finner invers av $Y[k] \rightarrow y[n] = h[n] * x[n]$.
- ▶ **Beregningsbyrde:**
 - ▶ En $2N$ samples konvolusjon krever N^2 komplekse multiplikasjoner.
 - ▶ Algoritmen over krever $3N \log_2(2N) + 2N$ komplekse multiplikasjoner v/FFT.
 - ▶ Gitt $N = 2^m$:

$$3N \log_2(2N) + 2N = 3 \cdot 2^m \log_2(2 \cdot 2^m) + 2 \cdot 2^m = 2^m(3m + 5)$$
 - ▶ $m = 5$: $2^m(3m + 5) = 640$
 - ▶ $m = 10$: $2^m(3m + 5) = 35\,840$
 - ▶ Lange sekvenser blir fort veldig beregningstunge!
- ▶ Løsning: Blokk-konvolusjon. To typer:
 - ▶ Overlap-add
 - ▶ Overlap-save

Overlap-add

1. Splitt $x[n]$ i m sekvenser $x_i[n]$ med lengde M .
2. Nullutvid $x_i[n]$ til 2-potens
3. Beregn

$$X_i[k] = F_{\text{DFT}}\{x_i[n]\}$$
 ved å bruke FFT
4. Legg sammen alle $X_i[k]$

$$X[k] = \sum_{i=0}^{M-1} X_i[k]$$

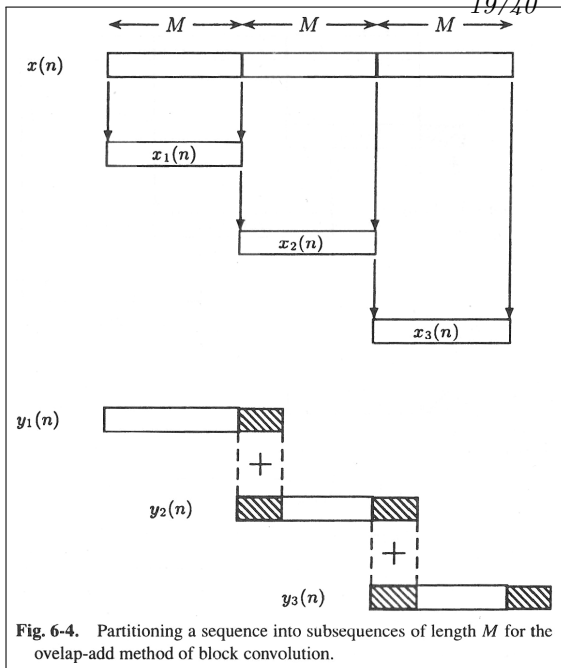
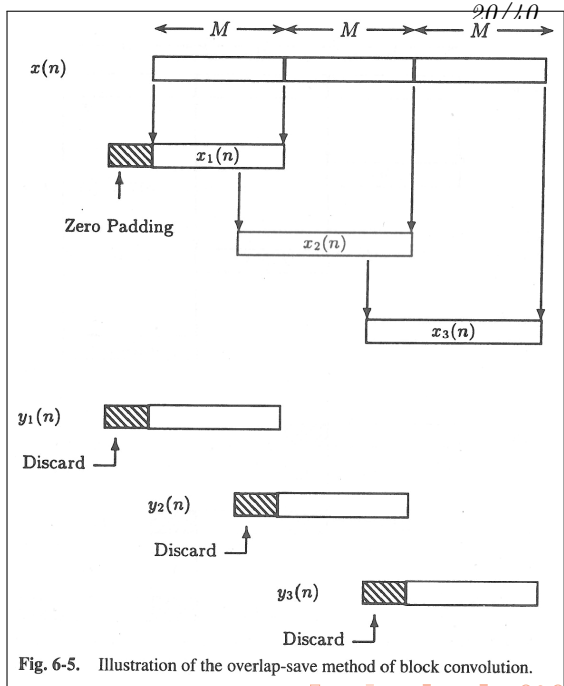


Fig. 6-4. Partitioning a sequence into subsequences of length M for the overlap-add method of block convolution.

Overlap-save

Hvis $x[n]$ er den lange sekvensen:

1. Nullutvid $x[n]$ med $N - 1$ nuller i front.
2. Del i k overlappende segmenter ($N - 1$ overlapp) av med lengde M (typisk $M = 2N$)
3. Nullutvid $h[n]$ til lengde M .
4. Beregn periodisk konvolusjon for hver del og kutt overlapp.



Interpolasjon i frekvens

Ønsker å interpolere opp et signal med lengde N til lengde NM :

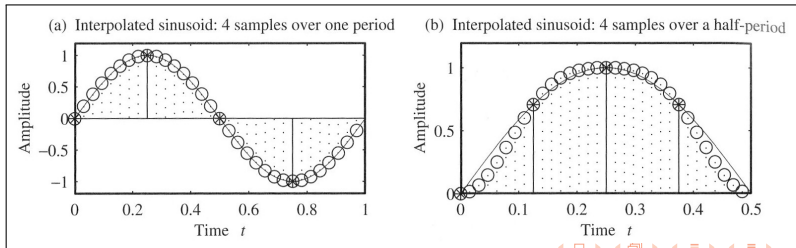
- $x[n] \xrightarrow{\text{DFT}} X[k]$

- Nullutvid $X[k]$ til $X_i[k]$ på følgende måte:

$$X_i[k] = \begin{cases} \{X[0], \dots, X[N/2 - 1], \dots, (M-1)N \text{ 0'er} \dots X[N/2], \dots, X[N-1]\} & \text{hvis } N \text{ odde} \\ \{X[0], \dots, 0.5X[N/2], \dots, (M-1)N - 1 \text{ 0'er} \dots 0.5X[N/2], \dots, X[N-1]\} & \text{hvis } N \text{ like} \end{cases}$$

- $x_i[n] \xleftarrow{\text{IDFT}} X_i[k]$

- Det hadde vært identisk å nullutvide $x[n]$ med nuller mellom samplene, ta DFT'en, legge på et LP-filter, og inverstransformere.



Tema

Frekvensestimering

Periodogram og dets avarter

Spektrogram

Periodogrammet

- ▶ **Effekt tetthets spekteret** (Power Spectral Density, PSD) er definert ved

$$R_{xx}(f) = \mathcal{F}\{r_{xx}(t)\}$$

det $r_{xx}(t)$ er autokorrelasjonen av $x(t)$,

$$r_{xx}(t) = \int_{-\infty}^{\infty} x(t') x(t + t') dt'$$

- ▶ Problem: Vi har ikke $x(t)$ men en samlet endelig tidssekvens $x[n]$.
- ▶ Vi må *estimere* $R_{xx}(f)$. En mulighet er *Periodogrammet*:

$$P[k] = \frac{1}{N} |X_{DFT}[k]|^2$$

- ▶ Godt estimat for deterministiske, båndbegrensede power signaler samlet over Nyquist-raten.
- ▶ Dårlig estimat for støyfullt signal.
- ▶ Estimert ikke forbedret med økende N .

Forbedrede estimer

- ▶ Bartletts metode
 - ▶ Del dataene opp i L ikke-overlappende deler.
 - ▶ Beregn P for hver del og P_B som et midlet estimat.
- ▶ Welchs metode
 - ▶ Del dataene opp i K overlappende deler.
 - ▶ Beregn P fra vektete utgaver av hver del og P_W som et midlet estimat.
- ▶ Blackman-Tukeys metode
 - ▶ Beregn estimatet for $r_{xx}(t)$ ved $r_{xx}[n]$.
 - ▶ Beregn P_{B-T} som $DFT\{w \cdot r_{xx}\}$, der $w[n]$ er et vindu (som har positivt FT).
 - ▶ Kalles også “smoothed periodogram”.

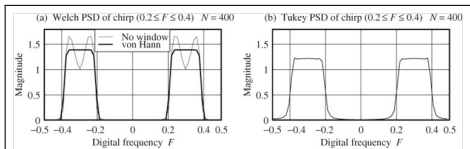
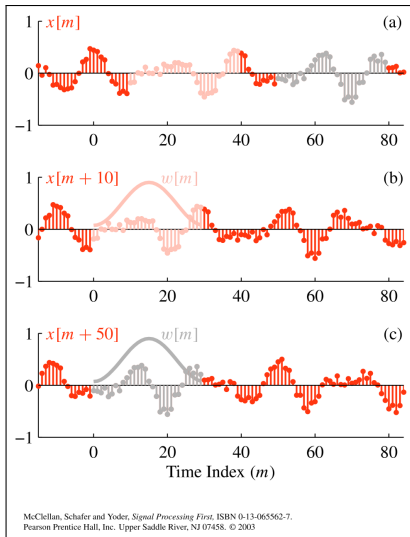


FIGURE 8.10 Welch and Tukey PSD of a chirp signal. A chirp signal is characterized by a spectrum that is constant over the frequency range present in the chirp. The unwindowed spectrum of a chirp signal is shown (in light) in the first plot. The

Tid-frekvens plott

- Spektrogrammet
definert ved

$$X[k, n] = \sum_{m=0}^{L-1} w[m] x[n+m] \times e^{-j(2\pi k/N)m}$$



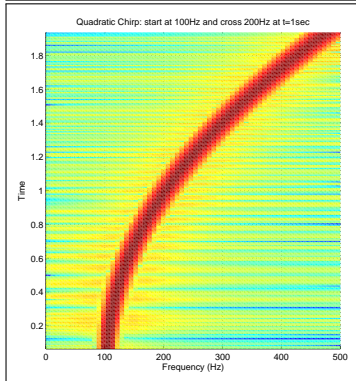
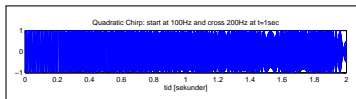
Tid-frekvens plott ...

I matlab:

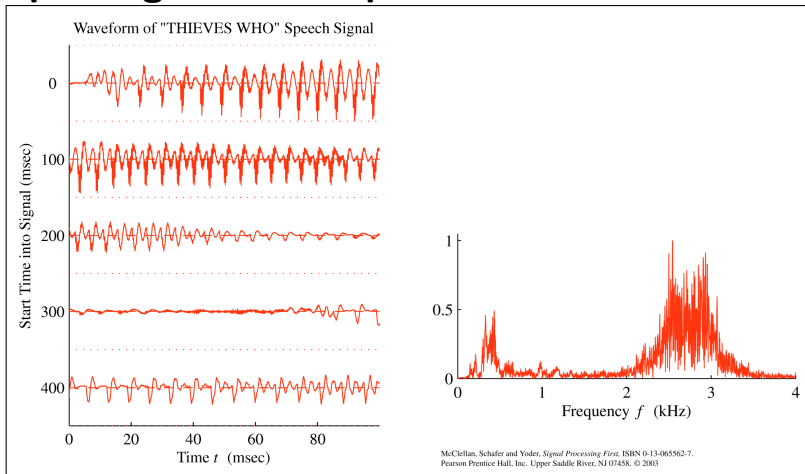
```

1 [s,f,t] = spectrogram(x>window,
2     noverlap,NFFT,Fs)
3
4 t=0:0.001:2;
5 % 2 secs @ 1kHz sample rate
6 y=chirp(t,100,1,200,'q');
7 % Start @ 100Hz, cross 200Hz at t=1s
8
9 plot(t,y);
10 xlabel('tid [sekunder]');
11 title(['Quadratic Chirp: start at
12     100Hz and cross 200Hz at t=1s'
13     ]);
14
15 spectrogram(y,128,120,128,1E3);
16 % Display the spectrogram
17 title(['Quadratic Chirp: start at
18     100Hz and cross 200Hz at t=1s'
19     ]);

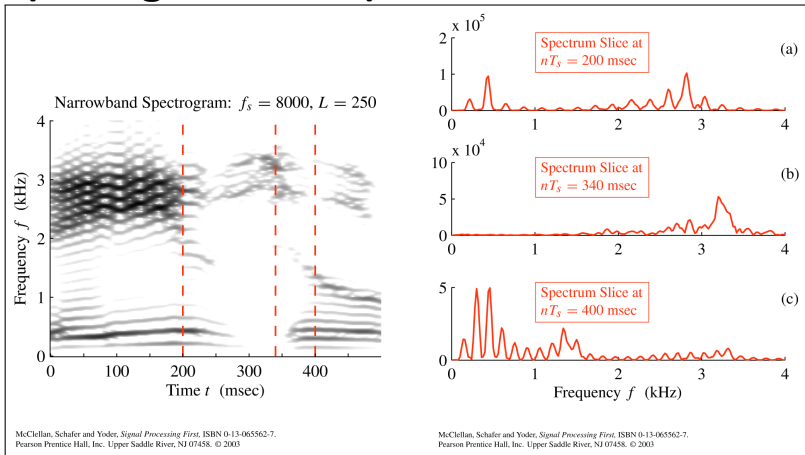
```



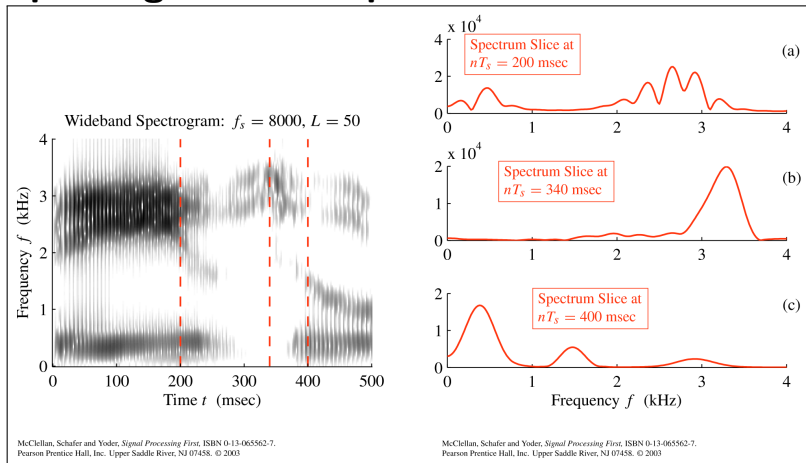
Spektrogram, eksempel



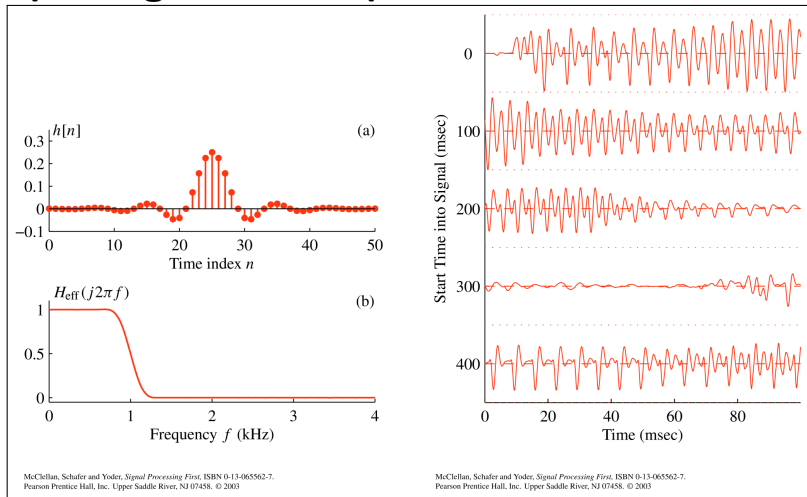
Spektrogram, eksempel ...



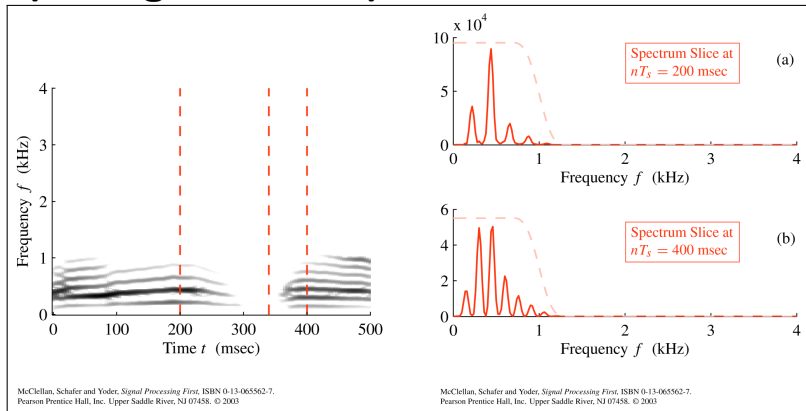
Spektrogram, eksempel ...



Spektrogram, eksempel ...



Spektrogram, eksempel ...



Tema

FFT

DFT på matriseform

Radix-2

DFT på matriseform

8.13 Matrix Formulation of the DFT and IDFT

If we let $W_N = e^{-j2\pi/N}$, the defining relations for the DFT and IDFT may be written as

$$X_{\text{DFT}}[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}, \quad k = 0, 1, \dots, N-1 \quad (8.46)$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X_{\text{DFT}}[k] [W_N^{nk}]^*, \quad n = 0, 1, \dots, N-1 \quad (8.47)$$

The first set of N DFT equations in N unknowns may be expressed in matrix form as

$$\mathbf{X} = \mathbf{W}_N \mathbf{x} \quad (8.48)$$

Here, \mathbf{X} and \mathbf{x} are $(N \times 1)$ matrices, and \mathbf{W}_N is an $(N \times N)$ square matrix called the **DFT matrix**. The full-matrix form is described by

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix} = \begin{bmatrix} W_N^0 & W_N^0 & W_N^0 & \dots & W_N^0 \\ W_N^0 & W_N^1 & W_N^2 & \dots & W_N^{N-1} \\ W_N^0 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ W_N^0 & W_N^{N-1} & W_N^{2(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix} \quad (8.49)$$

The exponents t in the elements W_N^t of \mathbf{W}_N are called **twiddle factors**.

DFT på matriseform

8.13 Matrix Formulation of the DFT and IDFT

If we let $\mathbf{W}_N = e^{-j2\pi/N}$, the defining relations for the DFT and IDFT may be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) \mathbf{W}_N^{kn}, \quad k = 0, 1, \dots, N-1 \quad (8.46)$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \mathbf{W}_N^{-kn}, \quad n = 0, 1, \dots, N-1 \quad (8.47)$$

The first set of N DFT equations in N unknowns may be expressed in matrix form as

$$\mathbf{X} = \mathbf{W}_N \mathbf{x} \quad (8.48)$$

Here, \mathbf{X} and \mathbf{x} are $(N \times 1)$ column vectors, and \mathbf{W}_N is an $(N \times N)$ square matrix called the DFT matrix. The elements here is described by

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ \vdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} \mathbf{W}_0^0 & \mathbf{W}_0^1 & \mathbf{W}_0^2 & \dots & \mathbf{W}_0^{N-1} \\ \mathbf{W}_1^0 & \mathbf{W}_1^1 & \mathbf{W}_1^2 & \dots & \mathbf{W}_1^{N-1} \\ \mathbf{W}_2^0 & \mathbf{W}_2^1 & \mathbf{W}_2^2 & \dots & \mathbf{W}_2^{N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{W}_{N-1}^0 & \mathbf{W}_{N-1}^1 & \mathbf{W}_{N-1}^2 & \dots & \mathbf{W}_{N-1}^{N-1} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{bmatrix} \quad (8.49)$$

The elements in the columns \mathbf{W}_k of \mathbf{W}_N are called twiddle factors.

Fra forrige foil så ser vi at DFT'en og IDFT'en er gitt som

$$\mathbf{X} = \mathbf{W}_N \mathbf{x} \quad \text{og} \quad \mathbf{x} = \mathbf{W}_N^{-1} \mathbf{X}.$$

Å regne den inverse til en matrise er i et generelt tilfelle ofte beregningstungt, men vi vet jo at forskjellen på DFT og IDFT er:

1. n og k bytter plass.
2. Fortegnet til fasen til den komplekse eksponentialen snus.
3. Den inverse transformen har en normalisering.

(1) oppnår vi ved å transponere \mathbf{W} , (2) ved så å konjugere, og (3) ved til slutt å normalisere:

$$\mathbf{W}_N^{-1} = \frac{1}{N} (\mathbf{W}_N^*)^T = \frac{1}{N} \mathbf{W}_N^H$$

Beregningsbyrden her er veldig lav.

FFT

- ▶ *Fast Fourier Transform* (FFT) er en effektiv måte å regne DFT'en av et signal på. Beregningsbyrden reduseres *mye* fordi:
 - ▶ Den *utnytter symmetri og periodiske egenskaper* for å unngå unødvendige beregninger.
 - ▶ Den *bryter signalet opp i mindre segmenter med lengde N* (typisk $N = 2$, kalles da *radix-2*) og tar DFT'en av disse i stedet.
 - ▶ Like- og oddeindekser behandles hver for seg, noe som er beregningsmessig gunstig.

FFT - Radix-2

- ▶ Vi prøver å ta DFT'en av like- og odde-indeksler hver for seg:

$$\begin{aligned}
 X_{\text{DFT}}[k] &= \sum_{n=0}^{N-1} x[n] W_N^{nk} \\
 &= \sum_{n=0}^{N/2-1} x[2n] W_N^{2nk} + \sum_{n=0}^{N/2-1} x[2n+1] W_N^{(2n+1)k} \\
 &= \sum_{n=0}^{N/2-1} x[2n] W_N^{2nk} + W_N^k \sum_{n=0}^{N/2-1} x[2n+1] W_N^{2nk} \\
 &= \sum_{n=0}^{N/2-1} x[2n] W_{N/2}^{nk} + W_N^k \sum_{n=0}^{N/2-1} x[2n+1] W_{N/2}^{nk} \\
 &= X^e[k] + W_N^k X^o[k]
 \end{aligned}$$

- ▶ $X^e[k]$ og $X^o[k]$ kan nå igjen splittes opp i like- og odde-indexer, og til slutt står man igjen DFT'en av veldig små sekvenser.

FFT - Radix-2

- ▶ DFT'en av veldig små sekvenser:

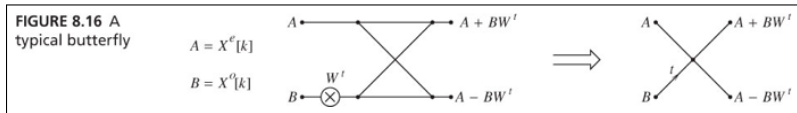
- ▶ 1-punkt:

$$\mathcal{F}_{\text{DFT}}\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{k}{N}n} = x[0] e^0 = x[0]$$

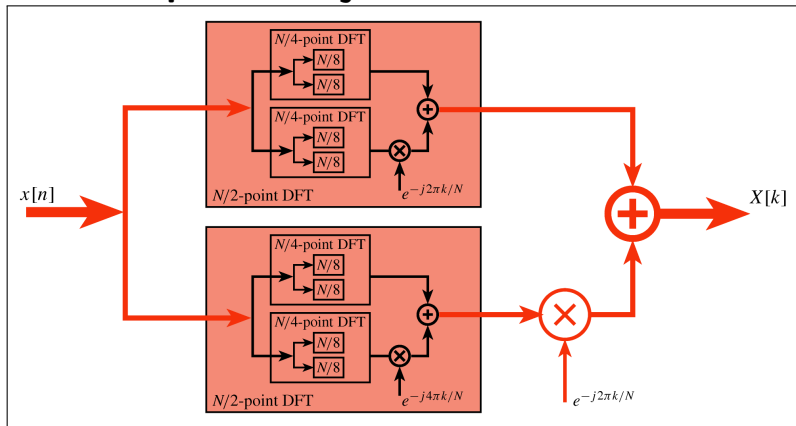
- ▶ 2-punkt:

$$\begin{aligned} \mathcal{F}_{\text{DFT}}\{x[n]\} &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{k}{N}n} \\ &= \begin{cases} x[0] e^0 + x[1] e^0 & = x[0] + x[1] & k = 0 \\ x[0] e^0 + x[1] e^{-j2\pi \frac{1}{2}} & = x[0] - x[1] & k = 1 \end{cases} \end{aligned}$$

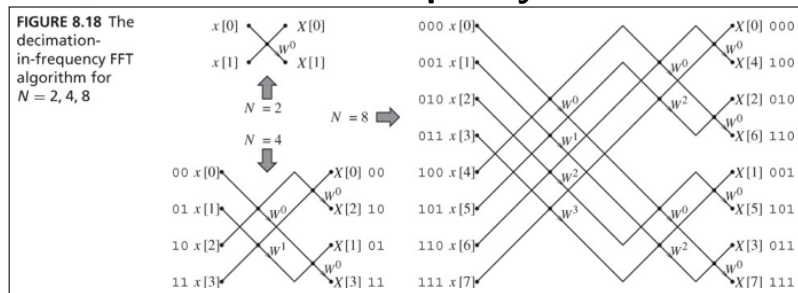
- ▶ DFT av små sekvenser \Rightarrow veldig enkelt.
- ▶ Det er dette FFT'en utnytter.



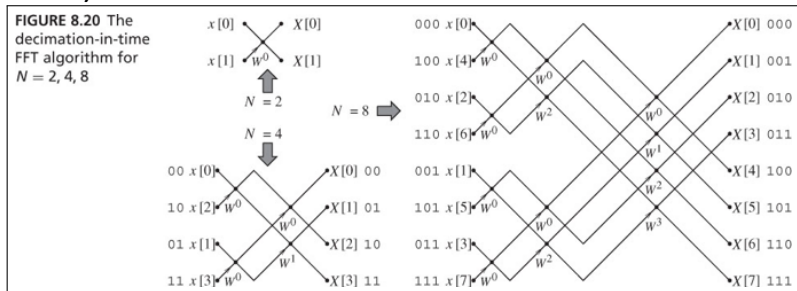
FFT - implementasjon



FFT, Decimation in frequency



FFT, Decimation in time



FFT, Beregningsbyrde

TABLE 8.7 ▶
Computational Cost
of the DFT and FFT

Feature	N -Point DFT	N -Point FFT
Algorithm	Solution of N equations in N unknowns	$0.5N$ butterflies/stage, m stages Total butterflies = $0.5mN$
Multiplications	N per equation	1 per butterfly
Additions	$N - 1$ per equation	2 per butterfly
Total multiplications	N^2	$0.5mN = 0.5N \log_2 N$
Total additions	$N(N - 1)$	$mN = N \log_2 N$