

# Kapittel 17

## Wavelet analyse

[Copyright for kapittelet, tekst og figurer: Arnt Inge Vistnes.]

### 17.1 Innledning\*

Syng en “a” og hold tonen hele tiden mens vi digitaliserer lyden. Fourier-omvendtes signalet vil systematiske studier kunne påvise såkalte “formanter” som karakteriserer de ulike vokallydene. I tilfeller som dette, når signalet er “stasjonært” (endrer ikke nevneverdig karakter underveis) er Fourier transform en fantastisk metode for å hente ut verdifull informasjon.

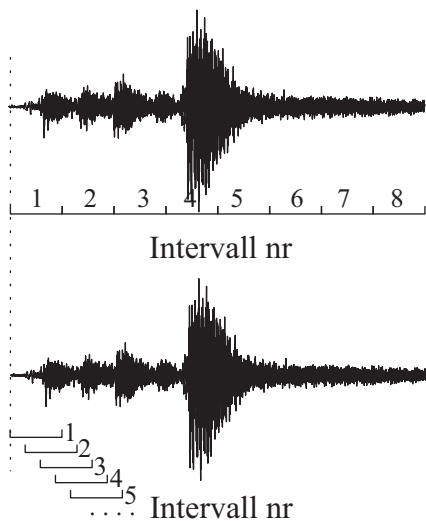
Da vi derimot foretok Fourier transform i siste del av oblig 1, så vi at Fourier transform egentlig er en metode som egner seg dårlig for signaler som endrer karakter med tiden. En av lydfilene vi arbeidet med bestod av skarpe trommelyder med nesten helt stille partier innimellom. I en Fourier transform forsvinner enhver tidsoppløsning, mens man har en god frekvensoppløsning. Men hva betyr egentlig frekvens i et signal som endrer karakter så mye underveis? Det spørsmålet bør man tenke litt over, slik at man ikke bruker en slagkraftig metode på et datamateriale der metoden egentlig ikke egner seg!

Vi kan bøte på dette problemet noe ved å stykke opp signalet slik at man har et noenlunde ensartet signal innenfor hver bit man analyserer. I så fall anvender vi en metode som kan kalles stykkevis Fourier transform (se figur 17.1). Wavelet-analyse egner seg imidlertid bedre. Vi skal i dette kapitlet se hva wavelet-analyse går ut på og hvilke fordeler den har bl.a. fremfor stykkevis Fourier transform.

Benyttes stykkevis FT ved at den totale datamengden deles opp i f.eks.  $m$  like store biter, skal det noe til at vi får ensartede signaler innenfor hver av bitene. Dersom vi i stedet velger å stykke opp i  $m$  biter med varierende lengde slik at det blir et noenlunde ensartet signal innen hver bit, vil det bli store forskjeller i frekvensspekteret fra bit til bit: Analyseresultatet vil bli kritisk avhengig av hvordan man stykker opp den totale datamengden.

En bedre metode kan kanskje være å velge ett “vindu” (f.eks. 256 datapunkter), og gjøre analysen først for de første 256 datapunktene i den lange datastrengen vi har, deretter flytte vinduet litt (f.eks. 16 punkter), og gjøre analysen på punkt 17 til og med punkt  $17+255=272$ , for så å flytte vinduet enda et hakk for neste analyse.

Ved en slik “glidende filter” metode unngår



Figur 17.1: For å få noe tidsinformasjon ved analyse av en lang tidsstreng med data, kan man stykke opp det totale tidsintervall- et og foreta Fourier transform for intervall etter intervall. Intervallene kan velges slik at de ikke overlapper hverandre (øverst) eller slik at de overlapper hverandre (nederst). Se teksten for vurdering av disse teknikkene.

man hopp i resultatene som skyldes tilfeldigheter i hvordan intervallene velges. Ulempen er at vi må gjennomføre til dels mange tilsynelatende unødvendige beregninger. Vi får med andre ord ganske mye overflødige data (engelsk: “redundancy”) i resultatene.

Resultatet kan fremstilles som et tredimensjonalt plot. Langs x-aksen har vi tiden for f.eks. midtpunktet i analysevinduet vi bruker, langs y-aksen har vi frekvensskalaen, og som den tredje komponenten har vi f.eks. intensiteten for bølgen (fourier-komponenten for denne frekvensen kvadrert). Denne informasjonen kan gis i form av “stack-plot”, kvotekurver, surface mesh, eller fargekoding for å nevne noe anskuleligformer.

Vi kommer til å benytte et slags “glidende filter” i vår prosjektoppgave i år, men altså

basert på wavelet-analyse i stedet for Fourier transform. Metoden vi bruker er en form for kontinuerlig wavelet transform (siden vi flytter “analysevinduet” stort sett en liten avstand relativt til vinduets bredde). Det kan nevnes at det finnes også en diskret wavelet transform hvor man foretar så få transformasjoner som overhodet mulig uten tap av informasjon. En slik transform er mye mer effektiv enn den kontinuerlige, og brukes i teknologiske sammenhenger der det er viktig at ting går fort. Ulempen med en diskret wavelet transform er at resultatet er langt vanskeligere å forstå enn et vanlig Fourier-spekter. Det er hovedgrunnen til at vi ikke går inn på den metoden i vårt kurs.

Wavelet-analyse er et omfattende fagfelt innen matematikk/informatikk, og det gis egne kurs om emnet her ved UiO. Vi kommer ikke til å gå i detalj om den strengt matematiske eller datatekniske siden av wavelets. Hensikten med å ta med wavelets i vårt kurs, er å gjøre studenten oppmerksom på at Fourier transform *ikke* egner seg for ikke-stasjonære signaler, og samtidig peke på en analysemetode som da ofte er langt å foretrekke. Dessuten kan arbeid med wavelets bidra til en dypere forståelse av tidsavgrensede fenomener og de tilsvarende frekvenser. Blant annet er Heisenbergs uskarphetsrelasjon dypt forankret i de sammenhengene wavelet-analyse bygger på.

En del av dere vil kunne ha bruk for wavelet-analyse i masteroppgaven eller i et evt. PhD-prosjekt (og senere jobber). Wavelets brukes bl.a. for å analysere solflekkaktivitet (og endringer i syklusen med tiden), El Niño sørlige oscillasjoner i Stillehavet, isbre-sykler, ruhet, kornstørrelseanalyser, analyse av f.eks. krefceller vs normale celler og mye mer.

Teknologisk er det en omfattende bruk av

wavelets bl.a. i jpeg-komprimering av bildefiler, og i mp3-komprimering av lyd.

## 17.2 Kort historikk

Den franske matematikeren Joseph Fourier (1768-1830) "oppdaget" Fourier transform for vel 200 år siden. (Fourier arbeidet forøvrig med varmestrømning, og var visstnok den første som oppdaget drivhuseffekten.)

Fourier transform benyttes i stor grad i analytisk matematikk. I tillegg fikk transformen en enorm utbredelse i dataverdenen etter at J.W.Cooley og J.W.Tukey i 1965 oppdaget den såkalte "Fast Fourier Transform" (FFT) hvor man kunne foreta en Fourier transform svært mye raskere enn tidligere. Ved FFT benytter man symmetriene i sinus og cosinus-funksjonene for å redusere antall multiplikasjoner ved utregningen, men for å få dette til, må antall datapunkter være en heltalls potens av 2, dvs  $N = 2^n$ .

Det hevdes at Cooley-Tukeys Fast Fourier Transform egentlig ble oppdaget av Carl Friedrich Gauss ca 1805, men glemt og delvis gjenoppfunnet flere ganger før 1965. Suksessen til Cooley og Tukeys gjenoppdagning skyltes nok at datamaskinen gjorde sitt inn-tog omtrent på denne tiden.

Wavelet-analyse er av langt yngre dato. Riktignok ble wavelets introdusert allerede ca 1909, men metoden ble for alvor først tatt i bruk fra ca 1980 av. Det er langt større spillerom for spesielle varianter av wavelet-analyse enn ved Fourier transform. Det er både en fordel og ulempe. Man kan langt på vei skreddersy en waveletanalyse slik at den passer optimalt for de dataene man vil analysere. Uelpen er at den store varia-

sjonsmuligheten medfører at man må bruke hodet litt mer ved wavelet-analyse enn ved Fourier transform.

## 17.3 Matematiske uttrykk\*

### 17.3.1 Fourier transform\*

Vi har gått gjennom Fourier transform i kapittel 2, men la oss repetere de matematiske uttrykkene også her.

La  $f(t)$  være en integrerbar funksjon av tid. Da kan vi beregne en ny funksjon  $F(\omega)$ , hvor  $\omega$  er vinkelfrekvens, på følgende måte:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (17.1)$$

Det morsomme med denne funksjonen er at vi kan ta en tilsvarende "omvendt" transform:

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} d\omega \quad (17.2)$$

og ende opp med eksakt den opprinnelige funksjonen igjen. Merk fortegnskiftet i den komplekse eksponentialfunksjonen. Det finnes tre ulike måter å fordele faktoren foran integraltegnet på, vi har valgt den varianten som gir mest symmetriske uttrykk.

Fra likningene 17.1 og 17.2 ser vi at selv om  $f(t)$  er reell, vil  $F(t)$  være kompleks. Dette er nødvendig for at  $F(t)$  både skal kunne angi hvor kraftig svinging vi har ved ulike frekvenser, og i tillegg angi innbyrdes fase til de ulike frekvenskomponentene. (En symmetri i  $F$  medfører at  $f$  etter den omvendte transformen blir reell, slik den var opprinnelig.)

Det bør forøvrig nevnes at  $f$  og  $F$  generelt sett ikke behøver å være funksjoner av tid og frekvens. Det finnes mange ulike typer funksjoner og sammenhenger hvor Fourier transform anvendes. I vårt kurs begrenser vi oss imidlertid (nesten utelukkende) til tid og frekvens.

Uttrykkene ovenfor er beregnet på analytiske beregninger. Når vi bruker datamaskin, kjenner vi ikke fullstendig til hvordan  $f(t)$  varierer i tid. Vi kjenner bare verdien av  $f$  i diskrete tidspunkt  $t_n$ . I disse tidspunktene har  $f$  verdiene  $f_n$  hvor  $n$  er en indeks som varierer fra 1 til  $N$ , dersom  $f$  er beskrevet i  $N$  tidspunkt. Vi antar at det er valgt ekvidistante tidspunkt slik at det er en fast tid mellom to nærliggende tidspunkt. Total tid  $f$  er beskrevet over er da  $T = (N - 1) * \delta t$  hvor  $\delta t$  er tiden mellom to tidspunkt i beskrivelsen.

Når Fourier transform gjennomføres på diskrete data, brukes en diskret transform. Denne kan angis slik:

$$F_k = \sum_{n=1}^N f_n e^{-i \frac{2\pi k}{N} n} \quad (17.3)$$

hvor  $k = 0, 1, 2, \dots, N-1$  angir de ulike frekvenskomponentene.

Den tilsvarende ”omvendte” transformen er da:

$$f_n = \frac{1}{N} \sum_{k=0}^{N-1} F_k e^{i \frac{2\pi n}{N} k} \quad (17.4)$$

for  $n = 1, 2, 3, \dots, N$ .

### 17.3.2 Wavelet transform\*

Wavelet transform kan angis tilsynelatende på nokså analog måte som en Fourier transform:

La  $f(t)$  være en integrerbar funksjon av tid. Da kan vi beregne en ny funksjon  $\gamma(s, \tau)$ , hvor  $s$  er en “scaling”-parameter som kan ha en nær sammenheng med frekvens, og  $\tau$  har store likheter med tid. Enkeltverdier for den nye wavelet transformerte funksjonen kan finnes på følgende måte:

$$\gamma(s, \tau) = \int_{-\infty}^{\infty} f(t) \Psi_{s, \tau}^*(t) dt \quad (17.5)$$

Her er  $\Psi_{s, \tau}$  selve waveleten. Asteriksen sier at det er den kompleks konjugerte av uttrykket for waveleten vi må bruke.

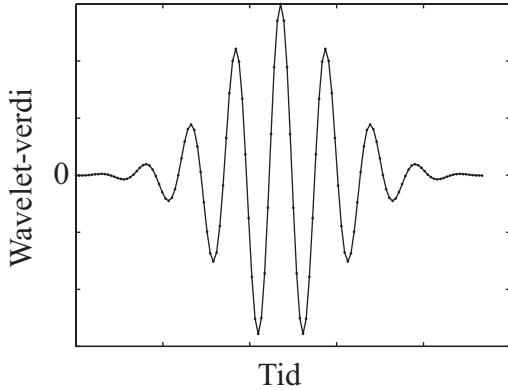
Det spesielle med wavelet-analyse er at man kan velge mellom nærmest uendelig mange forskjellige wavelets alt etter hva man ønsker å få fram i analysen. I vår sammenheng kommer vi bare til å bruke såkalt Morlet wavelets. Matematisk kan den uttrykkes som en sinusfunksjon konvolutert med en gaussisk funksjon (“gaussisk omhyllingskurve”). Waveleten kan beskrives som:

$$\Psi_{s, \tau} = \pi^{-\frac{1}{4}} e^{i \frac{\omega \tau}{s}} e^{-\frac{\tau^2}{2s^2}} \quad (17.6)$$

Figur 17.2 viser et eksempel på en Morlet wavelet. Vi ser at den bærer navnet med rette: “wavelet” kan nemlig oversettes med “Liten bølge”.

Dersom vi sammenligner ligning 17.1 med ligning 17.5, ser vi at uttrykkene ligner mye på hverandre. Vi integrerer opp produktet mellom en funksjon  $f$  med en bølge. Begge er derved knyttet til et “indreprodukt” innen matematikken, men som sagt, vi skal bare touche matematikken med en harelabb i vårt kurs.

Det er imidlertid flere forskjeller enn man først skulle tro. En vesentlig forskjell ligger i at wavelet-transformasjonen fører til en



Figur 17.2: Eksempel på en Morlet wavelet.

tredimensjonal beskrivelse, mens en Fourier transform beskrivelse bare er todimensjonal.

Også for en wavelet-transform er det mulig å foreta en ”omvendt” transform:

$$f(t) = \int \int_{-\infty}^{\infty} \gamma(s, \tau) \Psi_{s, \tau}(t) ds d\tau \quad (17.7)$$

Wavelet transform brukes nesten utelukkende i diskret form, siden beregningene er så omfattende at de nesten er umulige å gjennomføre analytisk (unntatt i svært enkle modell-beskrivelser).

La oss se på de diskrete uttrykkene: Vi starter som før med et signal som vi vil analysere. Signalet kan i det diskrete tilfellet med ekvidistante punkter beskrives med en tallrekke  $f_n$ .

En wavelet av Morlet type kan da på diskret form skrives som:

$$\Psi_{(s, \tau), (n' - n)} = \pi^{-\frac{1}{4}} \sqrt{\frac{dt}{s}} e^{i\omega \frac{(n' - n)dt}{s}} e^{-\frac{((n' - n)dt)^2}{2s}} \quad (17.8)$$

Her er  $n$  en indeks som henviser til tallrekken  $f_n$ , mens  $n'$  viser til at vi vil velge å plassere midtpunktet i waveleten på ulike posisjoner langs tallrekken  $f_n$ . Størrelsen  $dt$  er tiden

mellom to ”målinger”. Skaleringsparameteren  $s$  kommer vi tilbake til.

Selve wavelet-transformen vil da kunne beskrives slik:

$$\gamma(s, n') = \sum_{n=1}^N f_n \Psi_{(s, \tau), (n' - n)}^* \quad (17.9)$$

Det bør allerede her nevnes at selv om man i prinsippet summerer over *alle*  $n$ , dvs hele tidsstrengen, er dette unødvendig for de tilfeller at waveleten har en utstrekning som er liten i forhold til hele tidsstrengen. Vi kommer tilbake til dette om litt.

### 17.3.3 Noen matematiske krav

Det finnes ulike matematiske krav til wavelets, men vi tar bare tak i to av dem. Waveleten er en funksjon av tid, hvilket betyr at vi kan foreta en vanlig Fourier transform av waveleten. Da kreves det at Fouriertransformen for  $\Psi(\omega)$  i annen potens skal være integrerbar:

$$\int \frac{|\Psi(\omega)|^2}{\omega} d\omega < +\infty$$

Videre kreves det at

$$|\Psi(\omega)|^2 = 0 \text{ for } \omega = 0$$

Det siste kravet er ekvivalent med at middelverdien av  $\Psi(t)$  er lik null.

Begge disse kravene tilsier at wavelet-transformen har en ”bånd-pass-spekter”-karakter.

Morlet-waveleten tilfredsstiller ikke det siste kravet generelt sett. For enkelte verdier av  $\omega$  i ligning 17.6 er kravet meget nær tilfredsstilt. En av disse verdiene er  $\omega = 6$ . Vi kommer tilbake til dette temaet i prosjektoppgaven i år.

### 17.3.4 Faseinformasjon

I vanlig Fouriertransform foretar vi i prinsippet to transformeringer samtidig, nemlig en av typen

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \cos(-i\omega t) dt$$

og en av typen

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \sin(-i\omega t) dt$$

Grunnen er at vi må kunne fange opp f.eks. et sinussignal i  $f$  uansett hvilken fase det har.

I vanlig Fourier transform har vi *ett* startpunkt for analysen. Det betyr at det er lett å finne hvilken relativ fase de ulike frekvenskomponentene har.

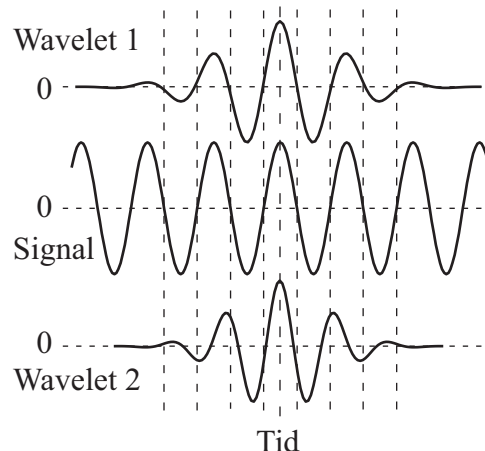
I waveletanalyse har vi ulike startpunkt og lengder på analysevinduet underveis i beregningene. Det gjør det langt vanskeligere å holde orden på faser. Dette er nok en av grunnene til at vi nesten utelukkende bare beregner kvadratet av verdiene, noe som vanligvis betegnes som *energien*, dvs

$$E(s, n') = |\gamma(s, n')|^2$$

### 17.3.5 Litt bildeliggjøring

La oss bildeliggjøre prosessen litt for at vi skal bedre forstå hva den går ut på. I figur 17.3 viser vi et utsnitt av en tidsstreng sammen med to ulike valg av wavelets. Wavelettransform består i å multiplisere signalet med waveleten, tidspunkt etter tidspunkt, og danne en sum av produktene.

For wavelet 1 ser vi at signalet skifter fortegn omtrent på samme sted som waveleten skifter fortegn. Det vil si at produktet i ethvert



Figur 17.3: Et sinusformet signal (i midten) sammen med en wavelet med samme periodetid (øverst) og en wavelet med noe kortere periodetid (nederst).

punkt blir positivt, og summen av produkter blir derfor ganske stor. For wavelet 2 skifter signalet fortegn på andre steder enn for waveleten. Noen av produktene blir derved positive og noen negative. Summen av produkter blir betydelig lavere enn for det første tilfellet.

Vi har derved forsøkt å anskueliggjøre at wavelettransformen av en regulær sinusbølge vil ha et maksimum når “bølgelengden” til waveleten svarer til “bølgelengden” til signalet i det tidsintervallet hvor vi foretar analysen.

Det betyr i neste omgang at dersom vi skal kunne plukke ut ulike bølgelengder i signalet  $f_n$ , må vi bruke wavelets med ulike bølgelengder.

Ser vi på det matematiske uttrykket for en wavelet, oppdager vi at vi kan i praksis få variert bølgelengden innen en wavelet ved hjelp av scale-parameteren  $s$ . Det ligger jo i navnet “scale”. Waveletanalyse kan gjennomføres med nokså forskjellige form



på waveleten, også med former der det er langt vanskeligere å peke på en “bølgelengde” enn for Morlet-waveleten vi bruker. Av den grunn bruker man ikke uttrykket “frekvens” i wavelet-analyse, men “scaling”. For Morlet-waveleten er det likevel et meget nært slektskap mellom frekvens og scale.

## 17.4 Frekvensoppløsning vs tidsoppløsning

Tar vi utgangspunkt i figur 17.3 skjønner man umiddelbart at dersom bølgelengden i signalet  $f$  er eksakt lik bølgelengden for waveleten, vil wavelet-transformen gi maksimal verdi (proporsjonal med amplituden i signalet  $f$ ). Endrer vi litt på bølgelengden i waveleten, vil transformen gi en lavere verdi.

Vi kan studere dette systematisk og sette opp resultatet i form av en figur som f.eks. i figur 17.4. I figuren er den generelle formen på waveleten gitt sammen med frekvensresponsen. Bølgelengden, eller rettere sagt den underliggende frekvensen da man genererte waveleten ifølge ligning 17.6, er variert for å framstille frekvensresponsen. Frekvensen til selve signalet  $f$  er uendret.

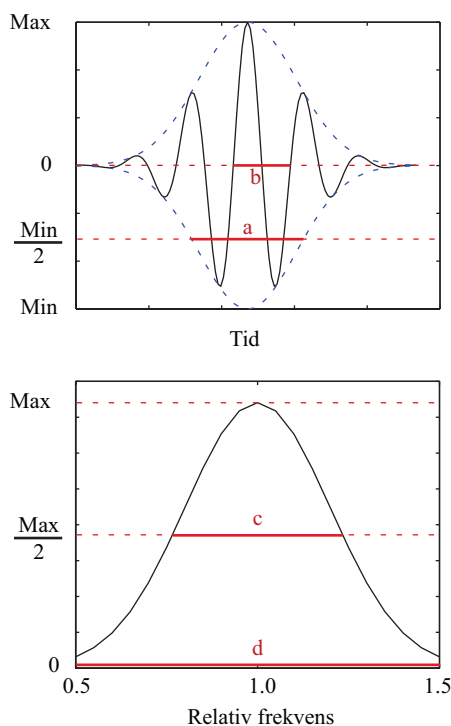
Frekvensresponsen er gitt med “relativ frekvens” langs x-aksen. Med det mener vi at frekvensen hvor bølgelengden for signalet er identisk med bølgelengden for waveleten er satt lik 1.0.

Vi ser at frekvensresponsen ser ut som en “klokkefunksjon” av en eller annen form. Det kan vises at den faktisk har en gaussisk form.

Halvverdbredden for tidsutstrekningen til waveleten tar utgangspunkt i omhyllingskurven, og er markert i figuren. Halvverdbred-

den kan angis i antall ganger periodetiden  $T$ , og figurteksten viser hvordan denne “relative” halvverdbredden kan beregnes.

Halvverdbredden for utstrekningen av frekvensresponsen er markert i nederste del av figuren. Figurteksten viser hvordan halvverdbredden kan beregnes som en relativ verdi i forhold til den sentrale frekvensen  $f = 1/T$ .



Figur 17.4: En wavelet har en avgrenset utstrekning i tiden. Man kan angi dette f.eks. som halvverdbredden, gjerne som et relativt mål (målt i antall periodetider  $T$ , dvs forholdet mellom lengdene  $a$  og  $b$  angitt i figuren). Med denne waveleten vil et helt frekvensintervall gi omtrent samme respons. Bredden i frekvensresponsen kan angis i relativt mål (målt relativt til senterfrekvensen, dvs  $1/T$ , dvs forholdet mellom lengdene  $c$  og  $d$  i figuren).

Vi gjennomfører så samme type beregninger for litt ulike bredder av Morlet-waveleten målt i antall bølgelengder. Dette svarer til at vi systematisk varierer  $\omega$  i ligning 17.6 og ellers beholder alt ellers likt.

I figur 17.5 er det vist fem waveleter med ulik relativ halvverdibredde (tall angitt) sammen med de tilsvarende frekvensresponsene. Også relative halvverdibredder i frekvensresponsene er gitt. Vi ser et generelt trekk, nemlig at når vi bruker en wavelet med mange toppe innenfor omhyllingskurven (stor relativ halvverdibredde i tid), blir frekvensresponsen skarpere (mindre relativ halvverdibredde i frekvens) enn når waveleten inneholder færre toppe.

Multipliserer vi disse halvverdibreddene med hverandre, finner vi at produktet ligger temmelig nær 1.0 for alle fem linjene i figuren.

Vi kan sette dette opp matematisk slik:

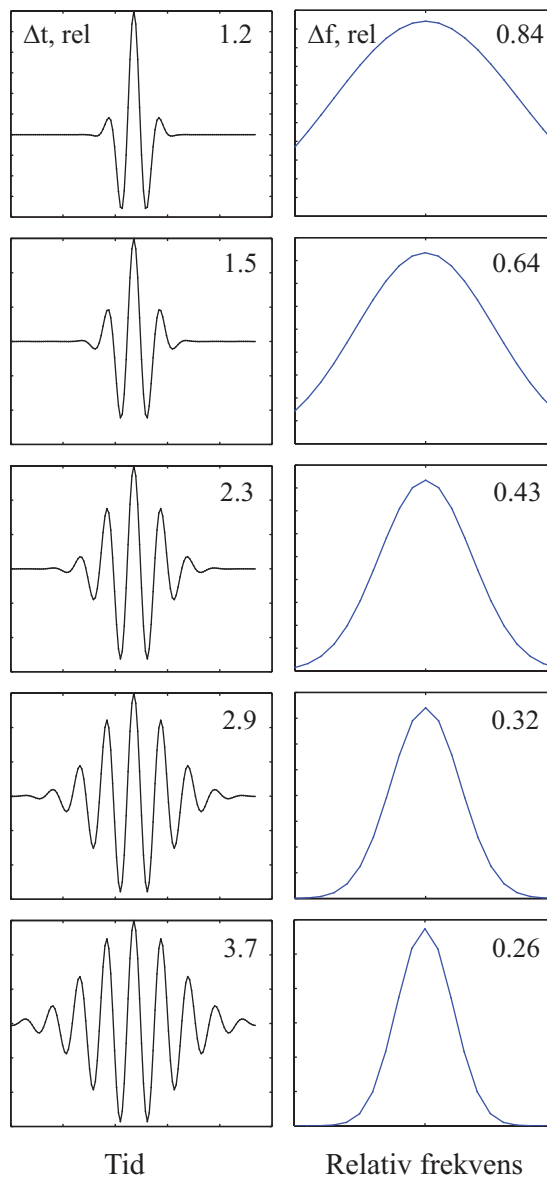
$$\Delta t \cdot \Delta f = 1$$

Her er det på sin plass med en liten, men viktig, digresjon. Multipliserer vi relasjonen ovenfor med Plancks konstant  $h$ , får vi:

$$\Delta t \Delta E = h \quad (17.10)$$

Her har vi allerede tatt oss friheten til å erstatte  $\Delta f \cdot h = \Delta(hf) = \Delta E$ . Vi har da funnet en relasjon som er meget nær det vi kjenner fra Heisenbergs uskarphetsrelasjon. Hadde vi brukt andre former for omhyllingsfunksjonen, ville produktet  $\Delta t \cdot \Delta f$  ikke vært eksakt 1, men som oftest litt høyere.

Ellers er det viktig å merke seg at vår definisjon av “usikkerhet” som en halvverdibredde i tids- og frekvensrespons er noe vilkårlig.



Figur 17.5: Her er det gitt fem ulike lengder av Morlet wavelets og de frekvensresponsene disse gir. Se forøvrig teksten for detaljer.



Med litt andre valg vil vi faktisk ende opp med den velkjente Heisenbergs uskarphetsfunksjon akkurat slik vi kjenner den fra kvantefysikken.

Vi har altså kommet fram til Heisenbergs uskarphetsrelasjon for tid og energi. Ved å ta utgangspunkt i en  $f$  som er en funksjon av posisjon, vil vi få en bølgetallsrespons i stedet for en frekvensrespons. Bruker vi så deBroglie-relasjonen, ender vi opp med Heisenbergs uskarphetsfunksjon for posisjon og bevegelsesmengde. Etter manges mening er de relasjonene vi har tatt utgangspunkt i, den dypeste kjernen i Heisenbergs uskarphetsrelasjoner. Og kjernen er *bølgenaturen* til såvel bølger som det vi ellers tenker oss som partikler.

I vår fremstilling ovenfor har vi tatt utgangspunkt i en analyse av et datamateriale. Vi foretar en “måling”. Vi har et signal  $f$  og forsøker å bestemme nøyaktig når en spesiell detalj i signalet befinner seg. Presisjonen i vår angivelse av tidspunkt er likevel begrenset av bredden av waveleten. En bred wavelet gir mindre mulighet for å oppdage raske tidsvariasjoner enn en smal wavelet. *Målenøyaktigheten ved tidsfastsettingen er altså begrenset av bredden til waveleten.*

På lignende måte er det med frekvensbestemmelsen. Har vi en bred wavelet, klarer vi å bestemme (“måle”) frekvensen nøyaktigere enn dersom vi “måler” ved å bruke en smal wavelet.

Med andre ord: Med en og samme måling kan vi ikke bestemme både en meget nøyaktig frekvens og et meget nøyaktig tidspunkt for når signalet befant seg.

En digresjon til:

Man kan undres over om relasjonen mellom tidsnøyaktighet og frekvensnøyaktighet *bare* er knytt-

et til *målingene*, men det er ikke slik! I wavelettransformen inngår et produkt av signal og wavelet. I vår analyse som ligger bak figur 17.5 valgte vi et signal med mye lengre varighet enn waveleten. Da var det formen til waveleten som satte begrensinger som ga oss frekvensresponsene som vist.

Dersom det var selve signalet  $f$  som hadde en langt kortere varighet enn waveleten, ville det svare til at vi byttet om wavelet og signal i våre matematiske uttrykk. Frekvensresponsen ville da bestemmes av selve signalet, og ikke av waveleten. Med andre ord, *det er selve signalet selv som setter de ultimate grensene for hva vi kan måle!*

Dette prinsippet slår begge veier: Dersom signalet varer svært lenge, kan vi i prinsippet velge en bredere og bredere wavelet og få en mer og mer nøyaktig frekvensbestemmelse (og dårligere og dårligere tidsbestemmelse), men bare til et visst punkt som er begrenset av selve signalets varighet.

Heisenbergs uskarphetsrelasjon har derfor *ikke* sin dybde i måleprosessen, men den forteller noe om det fysiske systemet i seg selv. Bredden til en spektrallinje kan vi måle på en god eller dårlig måte, og usikkerheten i måleresultatene varierer med hvordan vi foretar målingen. Men systemet *har* en bredde i seg selv, uavhengig av hvordan vi gjør målingene eller ikke, og det beste vi kan gjøre er å karakterisere den naturlige linjebredden så godt som mulig.

Hva vil det i neste omgang si at vi ikke klarer å bestemme frekvensen til lys nøyaktig? Med en partikkelmodell for lys ser man helt bort fra de betraktningene vi har gjort i dette kapitlet. Men kan det være at linjebredden for spektrallinjer er koblet opp med hvor lang tid det tar for en bølge av denne typen å passere oss? I så fall ville i alle fall vår forklaring av linjebredden få en forklaring som har klare analogier med mange andre sider av fysikk (mens dagens forklaring om “levetidsforbredning” henger i løse luften i forhold til fysikkfenomener forøvrig).

### 17.4.1 Sammenligning wavelets vs stykkevis FT

Skulle vi bruke stykkevis FT, ville et fast tidsintervall medføre at man har svært få (eller ingen hele) periodetider innenfor intervallet for lave frekvenser, men ganske mange periodetider for høye frekvenser. Det betyr at vi ville få en elendig frekvensoppløsning for de laveste frekvensene (målt som relativ frekvens), men en langt bedre frekvensoppløsning for de høyere frekvensene. Det betyr at vi ville ende opp med en analyse som ikke ville være optimal.

Skaleringsprosedyren som brukes i wavelet-analyse gir en optimal tidsoppløsning for *alle* frekvenser. Men man *kan* likevel velge å vektlegge tidsoppløsning *noe* på bekostning av frekvensoppløsning og omvendt alt etter hva man ønsker å studere. Det gjør at metoden blir et meget slagkraftig hjelpemiddel i mange sammenhenger.

Kunne vi valgt en stykkevis FT der vi faktisk brukte skaleringsprinsippet med å stykke opp tidsstrengen i mindre biter når vi analyserte høye frekvenser enn ved lave? Det ville kreve en enorm mengde Fourier transformasjoner, men ville da gi omtrent samme resultat som en waveletanalyse. Resultatet ville likevel ikke bli like godt. Vi ville nemlig med skalerte FT-intervaller i prinsippet hatt en ren wavelet-analyse, men med en wavlet som har en annen form enn Morlet. Det ville være en firkantpuls som da var omhyllingskurven. Frekvensresponsen ville da ha en form

$$\left(\frac{\sin(x)}{x}\right)^2$$

i stedet for en gaussisk omhyllingskurve. Resultatet ville bli en hale utenfor den sentrale toppen som gjør at vi får en dårligere frekvensbestemmelse enn ved Morlet wavelets.

Enda en digresjon:

I denne sammenheng kan vi trekke analogier til diffraksjon. Dersom vi sender lys inn mot en smal spalt og har *samme* intensiteten over hele spalten, vil diffraksjonsintensiteten nettopp være gitt som  $(\frac{\sin(\theta)}{\theta})^2$ . Dersom vi derimot sendte en lysstråle med *gaussisk* intensitetsprofil gjennom spalten, ville vi fått en gaussisk intensitetsprofil på skjermen uten ekstra striper ved siden av den sentrale toppen.

### 17.4.2 Logaritmiske akser

Da vi drøftet resultatene i figur 17.5 ovenfor, viste vi at det er en sammenheng mellom tidsoppløsning og frekvensoppløsning. Bruker vi f.eks wavletformen gitt midt i figuren, ser vi at frekvensresponsen viser at vi får nesten samme resultat selv om frekvensen på waveleten er 10 % høyere eller lavere enn den optimale. Frekvensresponsen er nemlig temmelig flat nær toppen. Det betyr at det er liten vits i å gjøre frekvensanalyser veldig nær den optimale, siden vi jo kjenner resultatet relativt nøyaktig dersom vi har resultatet for den optimale frekvensen.

Det betyr at når vi skal gjennomføre en wavelet transform, er det liten vits i å gjøre en analyse ved 1050 Hz dersom vi allerede har gjort det ved 1000 Hz. Tilsvarende vil det være lite vits i å gjøre en analyse ved 52.5 Hz dersom vi har gjort den ved 50 Hz. Vi merker oss at det er den *relative* frekvensen vi bør ta utgangspunkt i når vi skal velge hvilke frekvenser vi vil undersøke. Vi bør med andre ord velge en logaritmisk frekvensakse for å være mest mulig effektive ved beregningene. Til sammenligning kan vi minne om at Fourier transform gir en lineær frekvensakse.

For tidsoppløsningen er det annerledes. Ved analyse av lave frekvenser ville det vært gunstig å foreta store hopp i tid mellom hver

beregning, siden tidsoppløsningen da uansett er dårlig (sammenlignet med samplingsfrekvensen). Ved høye frekvenser måtte vi derimot bruke mye mindre tidssteg, siden tidsoppløsningen her er bedre (siden waveleten da er skalert ned slik at den har liten utstrekning i tid).

Ved diskret wavelet transform av den mest effektive typen, velger man ulike tidssteg alt etter frekvens. Vi vil likevel velge å bruke samme tidssteg uansett frekvens. Det gir mye unødvendig beregning, og waveletdiagrammet vi kommer fram til vil bære preg av at tidsoppløsningen i de lave frekvensene er elendig (og kan heller ikke bli bedre pga Heisenbergs uskarphetsrelasjon). Men det gir i det minste et resultat som er rimelig forståelig.

Vi ender da opp med et lin-log diagram: Lineært mhp tid og logaritmisk mhp frekvens.

Det er opp til oss som brukere av wavelettransformen å velge høyeste og laveste frekvens vi vil analysere. Den teoretisk høyeste frekvensen vi kan analysere svarer til en periodetid på to tidssteg ( $2/f_{\text{sampling}}$ ). Den laveste frekvensen man kan analysere er den som svarer til at hele dataopptaket utgjør én periodetid. Når vi kan velge grensene som vi vil, behøver vi ikke gå helt til disse ytterpunktene. Det kommer an på hvor mye vi vet om signalet på forhånd og hva vi er interessert i.

Så snart vi har valgt den høyeste frekvensen vi vil analysere, ligger det i kortene hvilket tidssteg vi bør bruke når vi arbeider oss gjennom den totale datastrengen  $f_n$ .

Ut fra bredden på frekvensresponsen til den waveleten vi faktisk bruker, må vi velge stegene i frekvens (scalingsfaktor) for analysen slik at ikke for mye informasjon går tapt og

heller ikke at vi gjør svært mye unødvendige beregninger.

Litt mer konkret:

Vi ga i ligning 17.8 følgende uttrykk for en Morlet wavelet:

$$\Psi_{(s,\tau),(n'-n)} = \pi^{-\frac{1}{4}} \sqrt{\frac{dt}{s}} e^{i\omega \frac{(n'-n)dt}{s}} e^{-\frac{((n'-n)dt)^2}{2s}}$$

Vi velger da f.eks. en minste  $s = s_0$  og multipliserer den skalaen  $s$  nettopp har brukt med en faktor for å finne skalaen vi skal bruke ved neste beregning. Når dette inngår i en løkke og vi har valgt å ta med  $M$  frekvenssteg, kan vi da evt sette:

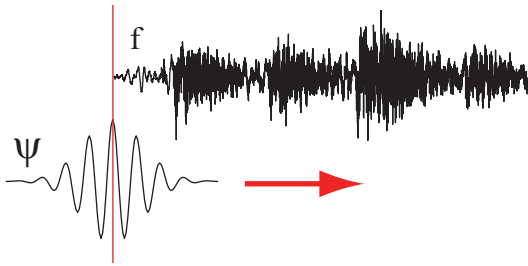
$$s_m = s_0 \cdot 2^{(m*dds)}$$

hvor  $m = 0, \dots, M-1$ . Dersom  $dds = 0.2$  betyr det at er nær 15 % økning i skala fra gang til gang (dvs ca 15 % reduksjon i midtfrekvensen i analysen vår). Skalaen dobles da for hvert femte beregning. Dersom  $dds = 0.1$  øker skala bare med 7.2 % og man må gå gjennom ti beregningsrunder før man har endret skala med en faktor 2 (og derved frekvensen med en faktor 0.5).

### 17.4.3 Randproblem

Når man skal foreta en wavelet transform, multipliserer man i prinsippet et signal med en wavelet og summerer alle produktene. Man flytter så waveleten og gjør det samme på ny. Dette foretar man seg fra at waveletens midtpunkt ligger i ene enden av signalet til waveletens midtpunkt ligger i andre enden av signalet.

Man endrer så skala for waveleten og gjør det samme på ny for å gjøre analysen ved en annen "effektiv frekvens" (skala).



Figur 17.6: *Det er ikke mulig å få et korrekt wavelet-resultat for tider og frekvenser som fører med seg at ikke hele waveleten kommer innenfor dataområdet under beregningene.*

Her oppstår det imidlertid et problem. Så lenge waveleten ikke er fullstendig innenfor dataområdet, vil man måtte forvente et annet resultat enn om hele waveleten ble brukt i beregningene. Dette er anskueliggjort i figur 17.6. For den posisjonen waveleten har i forhold til dataene i denne figuren, vil bare om lag halvparten av waveleten benyttes i praksis. Det betyr at summen av produktene må forventes å bli langt lavere (i størrelsesorden halvparten) av hva summen ville blitt dersom vi hadde fullt overlapp.

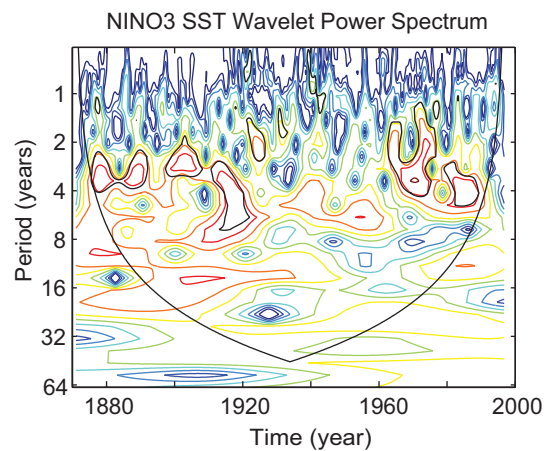
Av den grunn kan vi ikke stole noe særlig på datapunktene i en endelig wavelet transform hvor waveleten som er brukt ikke lå fullstendig innenfor datastrengen vi analyserer.

I figur 17.7 er det vist et eksempel på et waveletdiagram (scalogram) etter analyse av temperaturoscillasjoner i det sydlige Stillehav. Det er brukt kvoter sammen med farger for å markere “energien” i ulike former for svingninger (periodisitet) etter som de har utviklet seg de siste vel hundre år.

Det interessante for oss i denne sammenheng er den krumme V-formede linjen som starter langt nede og midt i skalogrammet med symmetriske buede linjer som går opp og ut mot kanten. Disse linjene markerer området hvor

waveletene er fullstendig innenfor datastrengen: Alt over denne buede V-en er gyldige data. Alt under er data man må ta med en klype salt.

Det kan være svært nyttig å tegne inn slike kurver i de skalogrammene man selv lager for å minne seg selv og andre om å ikke legge mye vekt på resultatene under den buede V-en.



Figur 17.7: *Eksempel på et wavelet-diagram (scalogram). Figuren er produsert med data og programvare tilgjengelig fra web (referanse 1 nevnt sist i kapittelet).*

## 17.5 Effektive algoritmer

Fourier transform har flere nydelige egenskaper, og en av dem kan vi benytte oss av for å effektivisere wavelet-analysen ut over det vi har angitt i de matematiske uttrykkene hittil.

Den Fourier transformerte av en tidsfunksjon  $f$  er (som vist ovenfor):

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt$$

Anta vi har en annen funksjon  $g$  beskrevet i samme tidsintervall som  $f$  og vi multipliserer verdiene av  $f$  og  $g$  i ethvert tidspunkt. Vi får da en ny funksjon  $h$ :

$$h(t) = (f * g)(t)$$

Da kan det vises at Fourier transformen for  $h$  kan finnes ved rett og slett multiplisere Fourier-spektrene til  $f$  og  $g$ . Med andre ord:

$$H(\omega) = F(\omega) \cdot G(\omega)$$

Hvordan kan vi bruke denne egenskapen i wavelet transform? Jo, en Fourier transform av et tidssignal gir oss et frekvensspekter (spektraltetthet vs frekvens) beregnet ut fra alle tidspunkt som var med i analysen. Dersom vi gjør en omvendt transform, kommer vi tilbake til tidbildet igjen.

La oss tenke oss at vi foretar en filtrering av det Fouriertransformerte signalet før vi transformerer tilbake, f.eks. ved å bare plukke ut frekvenser mellom 200 og 400 Hz og sette alle andre frekvensbidrag lik null. Etter tilbaketransformasjonen vil vi da få et tidssignal som forteller oss *når* det opprinnelige tidssignalet hadde mye eller lite innhold av frekvenser *omtrent* i intervallet 200 og 400 Hz.

Det er på sett og vis denne type informasjon vi ønsker å få ut ved waveletanalyse! Men prosedyren ovenfor med en skarp filtrering i intervall akkurat mellom 200 og 400 Hz ville ikke fungert ordentlig i praksis. Vi ville få en litt uryddig blanding av tidsinformasjon ved tilbaketransformasjonen.

Derimot er det slik at dersom  $f$  er det tidssignalet vi ønsker å analysere, og  $g$  er en wavelet, vil tilbaketransformeringen faktisk bli wavelet-transformen for akkurat den waveleten vi benyttet. Forutsetningen for at metoden skal fungere er at vi kjenner et analytisk

uttrykk for den Fourier-omvendte waveleten, dvs  $G(\omega)$ . Hver gang vi endrer scale i waveleten, vil  $G(\omega)$  endre seg, og prosedyren ovenfor må gjentas på ny.

På denne måten kan man bygge opp horisontal linje etter horisontal linje i waveletdiagrammet (skalogrammet) inntil vi har dekket så mange skalaer (frekvenser) som vi ønsker. Man kan undre seg på om det er noe effektivisering å bruke en slik algoritme sammenlignet med en rett fram metode som beskrevet først i dette kapitlet. Siden Fast Fourier Transform er så effektiv som den er, blir gevinsten med å gå via FFT betydelig.

## 17.6 Wavelet-ressurser på nett

1. <http://paos.colorado.edu/research/wavelets/>, A practical guide to wavelet analysis (Torrence og Compo) (se punkt 1, 4 og evt 6)
2. <http://pagesperso-orange.fr/polyvalens/clemens/wavelets/wavelets.html>, A really friendly guide to wavelets (C. Valens)
3. <http://tftb.nongnu.org/>, Time-frequency toolbox
4. <http://www-stat.stanford.edu/wavelab/> Wavelab850, Stanford og Berkley
5. <http://www-dsp.rice.edu/software/rice-wavelet-toolbox>, Rice Wavelet Toolbox
6. <http://www.cosy.sbg.ac.at/uhl/wav.html>, Mengde wavelet-lenker (men ikke oppdatert!)