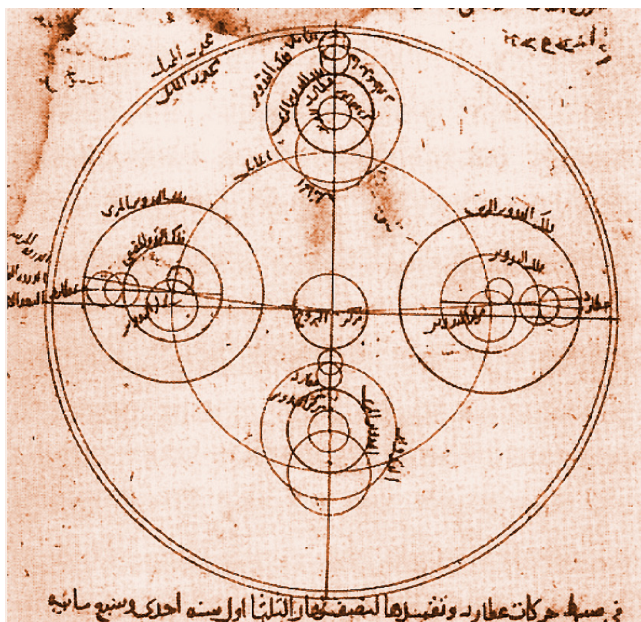


Kapittel 4

Fourieranalyse



Fouriertransformasjon og fourieranalyse har klare likhetstrekk med middelalderens bruk av episykler for å beregne hvordan planeter og sola beveget seg i forhold til hverandre. Figuren er nedlastet fra s1.hubimg.com/u/7219468_f520.jpg 30.01.2013 og lett bearbeidet.

Senere i boka vil vi omtale lyd fra ulike musikkinstrumenter, og til og med hvordan vi kan lage syntetisk lyd som ligner på den virkelige. Her er fouriertransformasjon et meget nyttig verktøy. Men ethvert verktøy har sitt bruksområde der det egner seg best. En fouriertransformasjon egner seg slett ikke for analyse av alle typer signaler. Vi trenger å være på vakt og ikke bruke fouriertransformasjon når metoden egentlig ikke passer.

I dette kapitlet skal vi ta for oss en meget anvendelig metode for å studere periodisitet i en funksjon eller et signal. Vi kommer nesten utelukkende til å foreta fouriertransformasjon av et signal som varierer i tid. Fouriertransformasjonen gir oss da et frekvensspekter, og ut fra frekvensspekteret kan vi foreta en fourieranalyse eller spektralanalyse.

Signalet vi starter ut med kaller vi ”tidsbilde”. En fouriertransformasjon gir oss da et ”frekvensbilde” av det samme signalet.

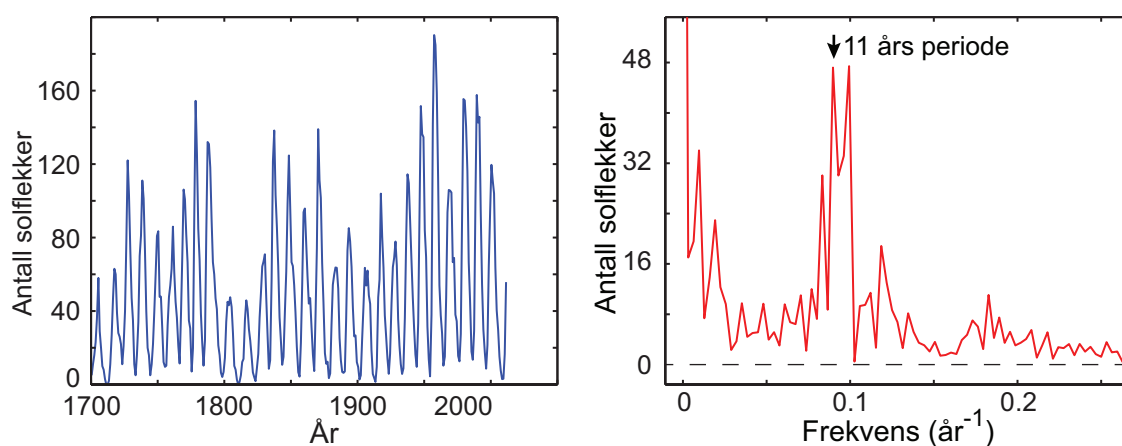
Det er mange detaljer og nye ord knyttet til fouriertransformasjon, så som grunnfrekvens og harmoniske, folding, samplings-teoremet, reell og imaginer del av fourierkoeffisienter, fullstendig sett funksjoner og mere til. Jeg er redd du rett og slett må gjennomføre noen fouriertransformasjoner på egen hånd for å forstå alt sammen.

¹Copyright 2013 for tekst og figurer: Arnt Inge Vistnes.

Versjon 30012013.

4.1 Innledning

Mengden av solflekker varierer med en periodetid på 11 år, hører vi rett som det er. Hva er grunnlaget for en slik påstand? Vi kan plote antall solflekker per år i løpet av en del år. Vi får da en kurve som i venstre del av figur 4.1. Vi kan se at det er omtrent 11 år mellom de ulike toppene, men vi skulle gjerne hatt en metode for å kvantifisere periodetiden og dessuten se om det også synes å være andre lengre periodetider som solflekkeaktiviteten følger. I slike sammenhenger bruker vi ofte fouriertransformasjon i analysen, og i høyre del av figur 4.1 er det vist et utdrag av resultatene etter en fouriertransformasjon av dataene i venstre del



Figur 4.1: Venstre del viser solflekker som dukket opp hvert år gjennom de siste tre hundre år. Høyre del viser et utdrag fra den tilsvarende fouriertransformerte funksjonene. Dataene er hentet 30.1.2012 fra <http://sidc.be/DATA/yearssn.dat>

I dette kapitlet skal vi ta for oss fouriertransformasjon. Vi starter med den klassiske matematiske beskrivelsen, går så raskt innom såkalte fourierrekker, men bruker mesteparten av tiden på diskret fouriertransformasjon. Den diskrete transformasjonen brukes svært mye i fysikk i dag. Den er godt anvendelig både ved analyse av eksperimentelle data og ved numeriske beregninger og simuleringer basert på teori.

Det er et mål for oss at kapitlet skal bidra til at du skal lære å beherske diskret fouriertransformasjon slik at du kan hente maksimalt med informasjon ut av de diagrammene fouriertransformasjonene gir. Samtidig ønsker vi at du skal unngå noen av de til dels alvorlige feilslutningene som dessverre også forekommer ved bruk av metoden.

4.2 Fouriertransformasjon (FT)

Det var den franske matematikeren og fysikeren Joseph Fourier (1768-1830) som introduserte fouriertransformasjon i analytisk matematikk.² Fouriertransformasjonen er siden videreutviklet til også å omfavne funksjoner gitt på digital form (som en endelig streng med tall). Vi kommer tilbake til den såkalte diskrete fouriertransformasjonen om litt.

Du har muligens møtt fouriertransformasjon i matematikken tidligere. I matematikken er transformasjonen gjerne knyttet opp til indreprodukt mellom to funksjoner, og vi definerer en basis av sinus og cosinusfunksjoner og anvender Gram-Schmidt på en funksjon for å finne dens Fouriertransformerte. Vi velger en mer praktisk tilnærming i vår sammenheng.

La oss se på den opprinnelige formalismen:

La $f(t)$ være en integrerbar funksjon med t (gjærne tid) som parameter. I fysikk er $f(t)$ ofte en reell funksjon, men matematisk sett kan den gjerne være kompleks.

Med basis i $f(t)$ kan det beregnes en ny funksjon $F(\omega)$, på følgende måte:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (4.1)$$

Parameteren ω er vinkelfrekvens dersom t representerer tid.

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

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} d\omega \quad (4.2)$$

og ende opp med eksakt den opprinnelige funksjonen igjen. Merk fortegnskiftet i den komplekse eksponensialfunksjonen.

I ulike bøker angis faktorene foran integraltegnene gjerne på ulikt vis. Vi har valgt den varianten som gir mest symmetriske uttrykk. Produktet av disse faktorene er $1/2\pi$ i denne formen for fouriertransformasjon.

Fra ligningene (4.1) og (4.2) ser vi at selv om $f(t)$ er reell, vil $F(\omega)$ være kompleks. Det er viktig å forstå årsaken til dette, og vi vil komme tilbake til denne problemstillingen mange ganger i dette kapitlet.

La oss aller først forsøke å gjennomskue hva $F(\omega)$ står for i et enkelt eksempel.

²Fourier er forøvrig kjent for å ha påvist/forklart drivhuseffekten for global oppvarming i 1824.

4.3 Hva sier $F(\omega)$ oss?

Erfaringsmessig er det mange som har vanskelig for å se for seg hva som skjer i en fouriertransformasjon. Vi vil derfor starte med en matematisk argumentasjon basert på enkle prinsipper i håp om at det skal lette innføringen i metoden.

Første utgangspunkt er å merke seg at fouriertransformasjonen inneholder et komplekst eksponentialledd. Vi minner derfor om Eulers formel:

$$e^{ix} = \cos x + i \sin x$$

Brukes Eulers formel på ligning (4.1), får vi:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \{f(t) \cos(\omega t) - if(t) \sin(\omega t)\} dt \quad (4.3)$$

Andre utgangspunkt er følgende kjente formler fra Rottmann:

$$\sin(x \pm y) = \sin x \cos y \pm \cos x \sin y$$

$$\cos(x \pm y) = \cos x \cos y \mp \sin x \sin y$$

Fra disse formlene kan vi utlede følgende relasjoner:

$$\sin x \cos y = \frac{1}{2} (\sin(x + y) + \sin(x - y)) \quad (4.4)$$

$$\cos x \cos y = \frac{1}{2} (\cos(x + y) + \cos(x - y)) \quad (4.5)$$

$$\sin x \sin y = \frac{1}{2} (\cos(x - y) - \cos(x + y)) \quad (4.6)$$

Tredje utgangspunkt er at integralet av en sinus eller cosinus over en periode er null. Det samme gjelder for "gjennomsnittsnivået" over mange perioder. En bestemt skrivemåte fører til et spesialtilfelle av interesse for oss:

$$\lim_{T \rightarrow \infty} \left\{ \frac{1}{2T} \int_{-T}^T \sin(at) dt \right\} = 0 \quad \text{for alle reelle } a \quad (4.7)$$

$$\lim_{T \rightarrow \infty} \left\{ \frac{1}{2T} \int_{-T}^T \cos(at) dt \right\} = 0 \quad \text{for alle reelle } a \neq 0 \quad (4.8)$$

$$\lim_{T \rightarrow \infty} \left\{ \frac{1}{2T} \int_{-T}^T \cos(at) dt \right\} = 1 \quad \text{for } a = 0 \quad (4.9)$$

I denne sammenheng kan det også være nyttig å minne om at:

$$\lim_{T \rightarrow \infty} \left\{ \frac{1}{2T} \int_{-T}^T \sin^2(t) dt \right\} = \lim_{T \rightarrow \infty} \left\{ \frac{1}{2T} \int_{-T}^T \cos^2(t) dt \right\} = 1/2$$

4.3.1 Fouriertransformasjon av en ren sinusfunksjon

Vi velger nå å finne den fouriertransformerte av funksjonen

$$f(t) = \sin(\omega_0 t)$$

Fouriertransformen av denne funksjonen blir:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \{ \sin(\omega_0 t) \cos(\omega t) - i \sin(\omega_0 t) \sin(\omega t) \} dt$$

Vi bruker ligningene (4.4) og (4.6), og får:

$$\begin{aligned} F(\omega) &= \frac{1}{\sqrt{8\pi}} \int_{-\infty}^{\infty} (\sin\{(\omega_0 + \omega)t\} + \sin\{(\omega_0 - \omega)t\}) dt \\ &\quad + \frac{1}{\sqrt{8\pi}} \int_{-\infty}^{\infty} i (\cos\{(\omega_0 + \omega)t\} - \cos\{(\omega_0 - \omega)t\}) dt \end{aligned} \quad (4.10)$$

Vi har nå et problem, nemlig at integralet av sinus eller cosinus egentlig ikke er veldefinert og endelig når integrasjonsgrensene er uendelig. For den videre argumentasjonen er det derfor en fordel å heller tenke seg et integral av typen gitt i ligning (4.9). Det er slik fouriertransformasjonen skjer i praksis i de situasjonene vi faktisk vil møte i dette kurset.

Det essensielle er å innse at det resulterende integralet *bare* får bidrag dersom

$$\omega = \omega_0 \quad \text{eller} \quad \omega = -\omega_0$$

Videre ser vi at det bare er imaginærdelen av uttrykket som overlever. Nærmere bestemt får vi (når vi ikke bruker faktoren $\sqrt{2\pi}$ i definisjonen av en fouriertransformasjon pga problemet med integrerbarheten):

$$F(\omega = \omega_0) = - \lim_{T \rightarrow \infty} \left\{ \frac{1}{4T} \int_{-T}^T i \cos(\omega_0 - \omega_0) dt \right\} = -i/2$$

Og tilsvarende:

$$F(\omega = -\omega_0) = i/2$$

og

$$F(\omega) = 0 \quad \text{for alle andre } \omega.$$

Ikke la deg irritere eller forvirre for mye av tilsynelatende inkonsistenser mhp konstantfaktorene som er brukt i ulike varianter av fouriertransformasjon i dette kapitlet. Det er ulike måter å gjøre dette på, og det spiller liten rolle hvilken variant vi velger så lenge vi vet hva vi gjør. I vårt kurs vektlegger vi numerisk fouriertransformasjon, og da er problemet mindre enn for analytiske uttrykk. Vær oppmerksom på at ulike dataprogrammer velger ulike konstantfaktorer. Dette spiller ingen rolle dersom vi bare betrakter relative verdier i den transformerte funksjonen. Imidlertid, dersom absoluttverdier er viktig, må vi sjekke med en kjent funksjon for å se hvilket valg av konstantledd som er brukt i det aktuelle programmet.

4.3.2 Fouriertransformasjon av en ren cosinusfunksjon

Dersom vi i stedet velger å finne den fouriertransformerte av funksjonen

$$f(t) = \cos(\omega_0 t)$$

blir fouriertransformen:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \{\cos(\omega_0 t) \cos(\omega t) - i \cos(\omega_0 t) \sin(\omega t)\} dt$$

Vi bruker i så fall ligningene (4.4) og (4.5), og får:

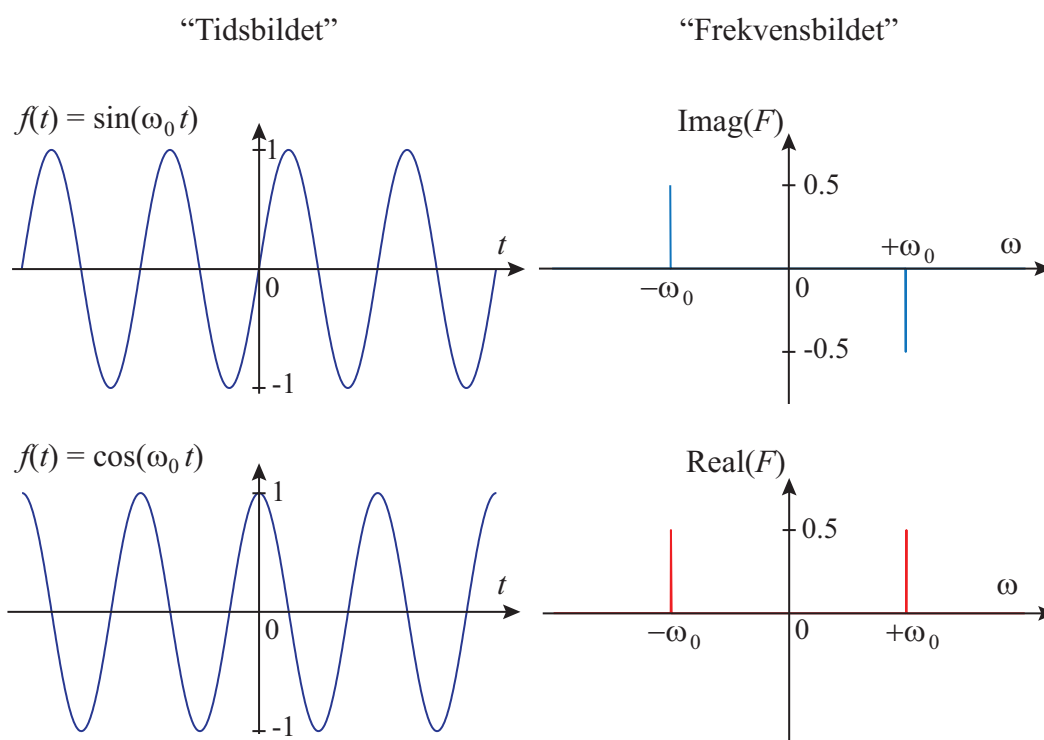
$$\begin{aligned} F(\omega) &= \frac{1}{\sqrt{8\pi}} \int_{-\infty}^{\infty} (\cos\{(\omega_0 + \omega)t\} + \cos\{(\omega_0 - \omega)t\}) dt \\ &\quad - \frac{1}{\sqrt{8\pi}} \int_{-\infty}^{\infty} i (\sin\{(\omega_0 + \omega)t\} + \sin\{(\omega_0 - \omega)t\}) dt \end{aligned} \quad (4.11)$$

Også i dette tilfellet får vi bare bidrag for $\omega = \omega_0$ og $\omega = -\omega_0$, men denne gangen er bidragene reelle og har samme fortegn:

$$F(\omega = \omega_0) = F(\omega = -\omega_0) = 1/2$$

$$F(\omega) = 0 \quad \text{for alle andre } \omega.$$

Figur 4.2 illustrerer det vi hittil har vist: Den fouriertransformerte av en ren sinusfunksjon og en ren cosinusfunksjon er null overalt unntatt ved vinkelfrekvensen til funksjonen vi starter ut med og den negative av denne vinkelfrekvensen. For sinusfunksjonen blir den fouriertransformerte funksjonen rent imaginær og $F(\omega)$ skifter fortegn når vi går fra ω_0 til $-\omega_0$. For cosinusfunksjonen er den fouriertransformerte rent reell og endrer ikke fortegn ved å gå fra ω_0 til $-\omega_0$.



Figur 4.2: Venstre: Tidssignalet til en harmonisk funksjon; $\sin(t)$ øverst og $\cos(t)$ nederst. Høyre: De tilsvarende fouriertransformerte funksjonene.

4.3.3 Fouriertransformasjon av en mer sammensatt funksjon

Dersom vi har funksjonen:

$$f(t) = A \cos(\omega_0 t + \phi) = a \cos(\omega_0 t) + b \sin(\omega_0 t)$$

sier resultatene ovenfor oss at den fouriertransformerte av $f(t)$ er:

$$F(\omega) = \begin{cases} a/2 - ib/2 \equiv c & \text{for } \omega = \omega_0, \\ a/2 + ib/2 \equiv c^* & \text{for } \omega = -\omega_0, \\ 0 & \text{for alle andre } \omega. \end{cases} \quad (4.12)$$

Vi innførte et komplekst tall c i ligningene ovenfor for å vise at $F(\omega_0)$ og $F(-\omega_0)$ er komplekskonjugert av hverandre.

Vi innser forhåpentligvis nå at fouriertransformasjon er en ganske nydelig metode for å analysere funksjoner. Dersom funksjonen er en ren harmonisk funksjon, vil vi kunne hente ut både amplitude og fase for funksjonen ved å bruke en fouriertransformasjon.

Dersom en funksjon er en sum av flere harmoniske funksjoner, vil vi kunne hente ut både amplitude og fase til hver eneste komponent i summen, siden analysen behandler funksjonen lineært.

Dersom vi har en sum av mange harmoniske funksjoner, vil sumfunksjonen kunne bli ganske uregelmessig og vanskelig å gjennomskue. I slike sammenhenger er iblant den fouriertransformerte av sumfunksjonen langt mer oversiktlig. Det er i slike sammenhenger at fouriertransformasjon er et nyttig hjelpemiddel i fysikk.

Merk at matematikken fører til at vi får bidrag både ved positiv og negativ vinkelfrekvens i analysen, selv om funksjonen vi analyserer bare kan sies å ha én frekvens (positivt reelt tall). Vi kommer tilbake til dette siden.

Det er viktig for utbyttet av resten av dette kapitlet at du raskest mulig gjennomskuer forskjellen og sammenhengen mellom $f(t)$ og $F(\omega)$. I fysikk omtaler vi gjerne $f(t)$ som *tidsbildet* av en funksjon, mens $F(\omega)$ angis som *frekvensbildet* eller *frekvensspekteret*.

4.4 Fourierrekker

Transformasjonene i ligning (4.1) og (4.2) forutsetter at vi kjenner funksjonen vi skal transformere i et uendelig langt tidsrom. Det er greit nok i idealiserte tilfeller (ren matematikk), men i praktiske fysiske eksperimentelle situasjoner er dette uaktuelt. Dersom vi imidlertid kjenner en funksjon over et endelig tidsrom T , kan vi lage en beskrivelse som strekker seg over uendelig tid ved å anta at funksjonen er periodisk med periodelengde lik T . Dette gir en interessant forenkling.

Dersom $f(t)$ er en periodisk funksjon med periode T , kan fouriertransformasjonen gjøres mer effektiv enn i den generelle transformasjonen i ligning (4.1). Transformasjonen kan angis ved en uendelig tallrekke, såkalte fourierkoeffisienter, $\{c_k\}$, der indeksen k er et naturlig tall mellom minus og pluss uendelig (!).

Fourierkoeffisientene beregnes slik:

$$c_k = \frac{1}{T} \int_{t_0}^{t_0+T} f(t) e^{-ik\omega_1 t} dt \quad (4.13)$$

hvor $\omega_1 = 2\pi\frac{1}{T}$, det vil si den vinkelfrekvensen som svarer til en funksjon som har nøyaktig én periode innenfor tiden T .

Siden $f(t)$ nå antas å være periodisk, kan startpunktet t_0 for integrasjonen i prinsippet velges fritt. Det antas at $f(t)$ er stykkevis glatt og kontinuerlig, og at $\int |f(t)|^2 dt < +\infty$ når integralet går over et intervall med lengde T .

Den inverse transformasjonen er da gitt ved:

$$f(t) = \sum_{k=-\infty}^{+\infty} c_k e^{ik\omega_1 t} \quad (4.14)$$

hvor igjen $\omega_1 \equiv 2\pi/T$ og svarer til frekvensen som har nøyaktig en sinusperiode innenfor intervallet T .

Dersom $f(t)$ er reell, kan det på grunn av symmetrien i ligning (4.12) enkelt vises at

$$f(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} \{a_k \cos(k\omega_1 t) + b_k \sin(k\omega_1 t)\} \quad (4.15)$$

hvor

$$a_k = c_k + c_{-k} = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \cos(k\omega_1 t) dt \quad (4.16)$$

$$b_k = i(c_k - c_{-k}) = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \sin(k\omega_1 t) dt \quad (4.17)$$

Ligning (4.15) sammen med uttrykkene (4.16) og (4.17) er gull verdt! De viser at ethvert periodisk signal med periode T kan skrives som en sum av harmoniske signal (sinussignaler med ulike amplituder og faser). I beskrivelsen inngår bare harmoniske signaler med eksakt et heltalls svingninger innenfor periodetiden T .

Harmoniske signaler med heltallige multiplum av en grunnfrekvens danner et ortogonalt basis-sett av funksjoner. Fourierrekken forteller da hvor mye vi har av hver av disse basis-funksjonene.

I prinsippet har vi uendelig mange basisfunksjoner med en frekvens som går fra $\omega_1 \equiv 2\pi/T$ til uendelig. Hva vinner vi da på å bruke en fourierrekke framfor en vanlig beskrivelse i tidsdomenet? Vel, i noen tilfeller er faktisk fourierrekken en mer komplisert beskrivelse enn en tidsbeskrivelse, men i mange fysikkrelaterede situasjoner er det motsatt. Dessuten gir fourierrekken et diagram som iblant avslører lovmessigheter i svingningene det ellers ville vært vanskelig å se.

Vi kommer tilbake til disse vurderingene senere i kapitlet, for det er svært viktig å være bevisst at vi ikke må blande rent matematiske analyser med fysiske tolkninger i hytt og vær, slik det dessverre gjøres en god del i dag!

4.5 Diskret fouriertransformasjon

En generell fourierrekke gitt i ligning (4.13) er basert på en kontinuerlig funksjon angitt i et intervall med lengde T . I det tilfellet endte vi opp med uendelig mange fourierkoeffisienter.

I vår moderne tid er eksperimentelle og computergenererte data bare kvasi-kontinuerlige. Vi samler en kontinuerlig funksjon, og ender opp med en tallrekke med et endelig antall datapunkter. Anta at dette er datapunkter tatt etter hverandre med en fast tidsdifferanse Δt . Vi kaller datapunktene x_n hvor $n = 0, \dots, N - 1$. Integralet i ligning (4.13) blir da erstattet med en sum.

Den diskrete fouriertransformasjonen er da gitt ved:

$$X_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n e^{-i\frac{2\pi}{N}kn} \quad (4.18)$$

for $k = 0, \dots, N - 1$. Dersom settet x_n består av verdier gitt i tidsdomenet, vil X_k være tilsvarende sett verdier i frekvensdomenet.

Den omvendte diskrete fouriertransformasjonen ser da naturlig nok slik ut:

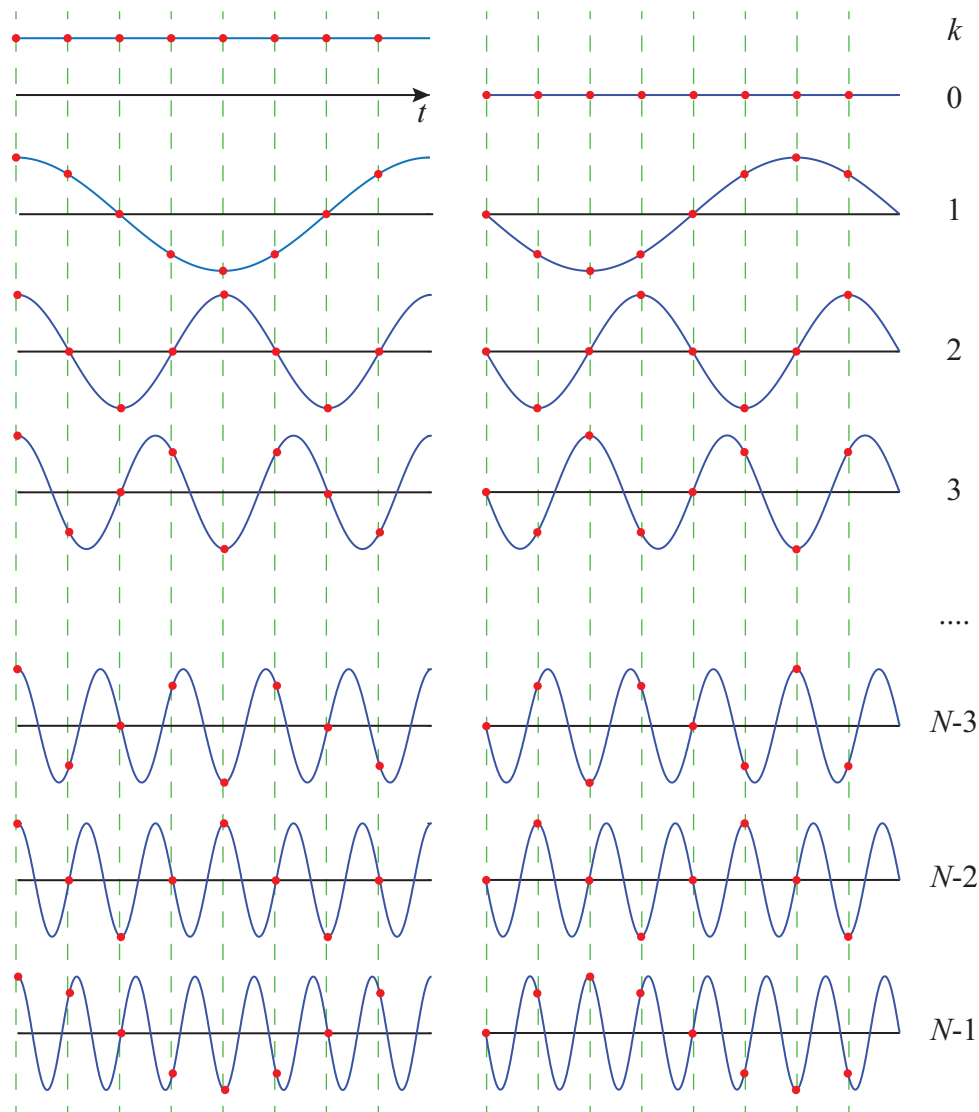
$$x_n = \sum_{k=0}^{N-1} X_k e^{i\frac{2\pi}{N}kn} \quad (4.19)$$

for $n = 0, \dots, N - 1$.

Vi så allerede i ligning (4.3) at fouriertransformasjon betyr at vi multipliserer funksjonen som skal transformeres med sinus- og cosinusfunksjoner med varierende frekvens. For den opprinnelige transformasjonen var funksjonen definert for alle tider t og vi multipliserte denne i prinsippet med sinuser og cosinuser med alle mulige frekvenser.

Når vi gjennomfører en diskret fouriertransformasjon, er funksjonen vi starter ut med bare definert i N punkter. Da kan vi bare multiplisere denne med en diskret representasjon av sinus- og cosinusfunksjoner, det vil si sinus- og cosinusfunksjoner som igjen bare er definert i N punkter.

Videre er funksjonen vår bare definert for en begrenset tid T . Da vet vi fra det vi har lært om fourierrekker, at det er tilstrekkelig å bruke sinus- og cosinusfunksjoner som har et helt antall perioder innenfor tiden T .



Figur 4.3: Funksjonene $e^{-i\frac{2\pi}{N}kn}$ som inngår i en fouriertransformasjon, cosinusfunksjonene til venstre og sinusfunksjonene til høyre. N ulike frekvenser blir brukt, og hver av frekvensene gir oss en fourierkoeffisient X_k .

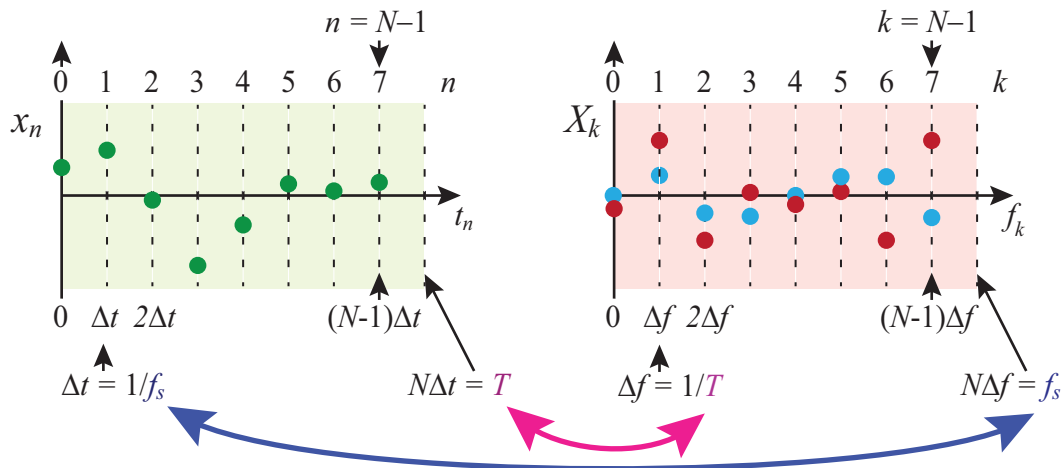
Alt dette er faktisk på plass i beskrivelsen av en diskret fouriertransformasjon slik den er gitt i ligning (4.18). Figur 4.3 viser hvilke funksjoner som leddet $e^{-i\frac{2\pi}{N}kn}$ representerer i et tilfelle der funksjonen som skal transformeres er definert i åtte punkter. Da blir sinus- og cosinusfunksjonene også bare definert i åtte punkter, og bare åtte ulike frekvenser er involvert (0, 1, 2, ..., 7 ganger den frekvensen som har nøyaktig én periode innenfor den tiden funksjonen er definert).

4.5.1 Diskret fouriertransform i mer fysiske termer

I ligning (4.18) og (4.19) er det matematisk sett bare snakk om et sett $\{x_n\}$ med N tall som kan transformeres til et nytt sett $\{X_k\}$ med N tall og tilbake igjen.

La oss nå koble matematikken til litt mer praktisk fysikk ved å se hva indeksene k , n og størrelsen N representerer.

Vi tenker at vi foretar N registreringer av en fysisk størrelse x_n over en begrenset tid T (se venstre del av figur 4.4). Dersom registreringene foretas med et mellomrom i tid lik Δt , sier vi at *samplefrekvensen* $f_s = 1/\Delta t$. Sammenhengen mellom størrelsene er som følger: $N = T f_s = T/\Delta t$.



Figur 4.4: En funksjon samlet i $N = 8$ tidspunkt (til venstre) sammen med fouriertransformasjonen til funksjonen (til høyre) som består av $N = 8$ komplekse tall. Realverdiene er gitt ved røde sirkelskiver og imaginærverdiene ved blå. Hvert punkt svarer til et lite tids- og frekvensintervall (hhv i venstre og høyre del). Merk sammenhengen mellom samplefrekvensen f_s og Δt og i særdeleshet sammenhengen mellom T og Δf . For å få en høy oppløsning i frekvensangivelsen i frekvensspekteret, må vi altså sample et signal i tilstrekkelig lang tid T .

Merk at hver sampling svarer til et helt lite tidsintervall Δt . I vår figur er det signalet i begynnelsen av hvert tidsintervall som registreres.

Fouriertransformasjonen i ligning (4.18) gir oss frekvensbildet (høyre del av figur 4.4). Frekvensbildet består av N komplekse tall, og disse er frekvenskomponentene fra frekvensen 0 (konstant-ledd), og frekvensene $k\Delta f$ hvor $\Delta f = 1/T$. (Denne detaljen bør du merke deg!).

Sammenhengen mellom ligning (4.18) og våre måletider $t_n = n\Delta t$ og de resulterende ana-

lysefrekvenser $f_k = k\Delta f$ kan illustreres slik (merk endringene i eksponensialfunksjonen):

$$X_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n e^{-i\frac{2\pi}{N}kn} = \frac{1}{N} \sum_{n=0}^{N-1} x_n e^{-i2\pi\frac{nT}{N}\frac{k}{T}} = \frac{1}{N} \sum_{n=0}^{N-1} x_n e^{-i2\pi(n\Delta t)(k\Delta f)}$$

$$X_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n e^{-i2\pi t_n f_k} = \frac{1}{N} \sum_{n=0}^{N-1} x_n \{\cos(\omega_k t_n) - i \sin(\omega_k t_n)\} \quad (4.20)$$

for $k = 0, \dots, N - 1$.

Fourierkoeffisienten X_k er altså på en måte integralet av produktet av signalet $\{x_n\}$ og en cosinus og en sinusfunksjon med frekvensen $\omega_k = k\Delta f$.

Gjør vi en lignende omskriving av den omvendte fouriertransformasjonen i ligning (4.19), får vi:

$$x_n = \sum_{k=0}^{N-1} (\Re(X_k) \cos(\omega_k t_n) - \Im(X_k) \sin(\omega_k t_n)) \quad (4.21)$$

for $n = 0, \dots, N - 1$. \Re og \Im står som før for realdelen og imaginærdelen.

Realdelen av X_k kan med andre ord betraktes som amplituden til hver enkelt cosinusfunksjon som inngår i summen, mens imaginærdelen av X_k er amplitudene til sinusfunksjonene. De komplekse kryssleddene i den opprinnelige ligning (4.19) er blitt borte på grunn av en symmetri som har rot i relasjonen vi viste i ligning (4.12), men som for diskret fouriertransform ser slik ut:

$$X_{k+1} = X_{N-(k+1)}^* \quad (4.22)$$

for $k = 0, \dots, N - 2$. Vi kommer tilbake til denne symmetrien når vi om litt skal diskutere "folding".

Siden tallrekken x_n er en vilkårlig valgt funksjon, samlet ved N tidspunkt, viser ligning (4.21) at funksjonen alltid kan beskrives som en lineær superposisjon av sinus- og cosinusfunksjoner som har eksakt et helt antall perioder innenfor den tiden T signalet er samlet. Frekvenskomponenten med lavest frekvens f_1 (ser bort fra konstantleddet f_0) har nøyaktig én periode i løpet av tiden T , mens den høyeste frekvensen som inngår i superposisjonen er lik samplingfrekvensen f_s (eller mer nøyaktig $(N - 1)/N \cdot f_s$). I tillegg til disse sinus- og cosinusfunksjonene kommer en konstantfunksjon (X_0) som er halve gjennomsnittsverdien av alle x_n .

Alternativt kan hver enkelt frekvenskomponent angis “på polar form”, som en amplitude og fase. Amplituden for hver enkelt frekvenskomponent er da gitt ved:

$$A_k = |X_k| = \sqrt{\Re(X_k)^2 + \Im(X_k)^2} \quad (4.23)$$

hvor $\Re(X_k)$ og $\Im(X_k)$ er hhv realdel og imaginærdel av X_k . Fasen for denne frekvenskomponenten er gitt ved:

$$\phi_k = \arctan(\Im(X_k)/\Re(X_k)) \quad (4.24)$$

Fase er i denne sammenhengen definert ut fra uttrykket $f(t) = A_k \cos(\omega_k t + \phi_k)$.

Husk at dersom du skal regne ut fasen på en datamaskin eller kalkulator, må du få med deg alle fortegnskombinasjoner for realdel og imaginærdel av X_k . Det betyr at du må bruke funksjonen $\text{atan2}(\Im(X_k), \Re(X_k))$.

♠ ⇒ Kommentar: Det finnes i dag en overordentlig effektiv algoritme for diskret fouriertransformasjon. Effektiviteten har medvirket sterkt til at fouriertransformasjon blir brukt mye i mange fag, ikke minst fysikk. Algoritmen ble visstnok oppdaget allerede i 1805 av Carl Friedrich Gauss, men ble glemte (den var ikke så interessant så lenge vi ikke hadde datamaskiner). Algoritmen ble i 1965 lansert av J.W.Cooley og J.Tukey som da arbeidet ved Princeton University. Deres fire siders artikkel: “An algorithm for the machine calculation of complex Fourier series.” i Math.Comput. 19 (1965) 297-301, hører til de “klassiske” artiklene som forandret fysikken.

I Matlab bruker vi Cooley og Tukey’s algoritme når vi anvender *fft* (“Fast Fourier Transform”) eller *ifft* (“Invers Fast Fourier Transform”). For å få full uttelling av metoden bør vi passe på at antall punkter N er eksakt et av tallene 2^n hvor n er et heltall. Det er først da vi får benyttet oss fullt ut av symmetrien i en sinus- og cosinusfunksjon.

⇐ ♠]

4.6 Et konkret eksempel

Vi vil nå gi et eksempel på fouriertransformasjon av en konkret funksjon og vil da i praksis se de lovmessighetene vi fant matematisk tidligere i kapitlet. Vi vil gå i stor detalj i den hensikt at kapitlet skal kunne brukes som en praktisk guide når vi ønsker å hente maksimalt og mest mulig presis informasjon ut av en fouriertransformasjon. Vi tror også at en meget detaljert gjennomgang av et eksempel vil kunne hjelpe på forståelsen av fouriertransformasjon, men forståelsen kommer ikke av seg selv! Vær nøye med detaljer!

Aller først vil vi skrive om ligning (4.18) og (4.19) på en slik måte at indeksene starter fra 1 og ikke fra 0 (for å klargjøre uttrykkene for Matlab-kjøring).

Uttrykket for en diskret fouriertransformasjon blir da som følger:

$$X_k = \frac{1}{N} \sum_{n=1}^N x_n e^{-i\frac{2\pi}{N}(k-1)(n-1)} \quad (4.25)$$

for $k = 1, \dots, N$. Dersom $\{x_n\}$ er en beskrivelse i tidsbildet, vil $\{X_k\}$ være en tilsvarende beskrivelse i frekvensbildet.

Den omvendte diskrete fouriertransformasjonen er gitt ved:

$$x_n = \sum_{k=1}^N X_k e^{i\frac{2\pi}{N}(k-1)(n-1)} \quad (4.26)$$

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

Innfører vi fysiske størrelser på lignende måte som ovenfor, vil vi med indekser fra 1 til N få følgende uttrykk (symbolene betyr det samme her som ovenfor, bl.a. i figur 4.4):

$$N = T f_s, \quad f_k = \frac{k-1}{T} = (k-1)\Delta f, \quad \text{og} \quad t_n = \frac{n-1}{N}T = (n-1)\Delta t$$

for $f_k = 0, \dots, f_s(N-1)/N$ og $t_n = 0, \dots, T(N-1)/N$.

Med disse symbolene blir uttrykket for den diskrete fouriertransformasjonen:

$$X_k = \frac{1}{N} \sum_{n=1}^N x_n e^{-i2\pi f_k t_n} \quad (4.27)$$

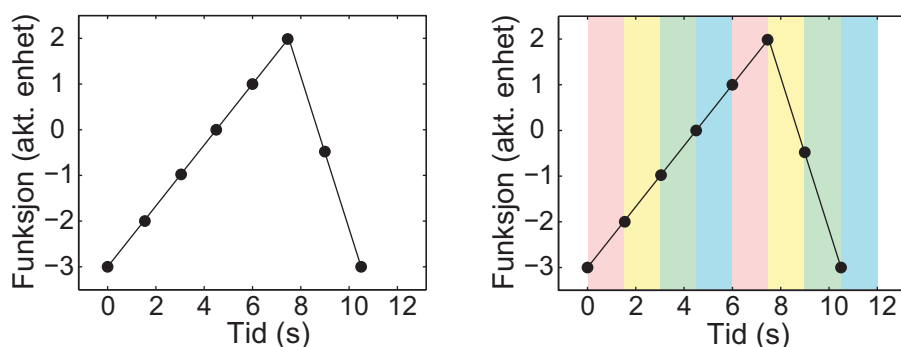
for $k = 1, \dots, N$.

og uttrykket for den inverse diskrete fouriertransformasjonen blir:

$$x_n = \sum_{k=1}^N X_k e^{i2\pi f_k t_n} \quad (4.28)$$

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

For å gjøre eksemplet så oversiktlig og enkelt som mulig, har vi valgt å beskrive en funksjon i bare $2^3 = 8$ punkter. Funksjonen ser i først omgang ut som en skjev sagtann, og venstre del av figur 4.5 viser funksjonen. Matlab-koden som ble brukt for å definere og plote funksjonen er vist nedenfor.



Figur 4.5: Venstre: Tidssignalet til en funksjon, definert i åtte punkter. Det er trukket en rett linje mellom hvert av punktene. Høyre: Hvert målepunkt svarer til hvert sitt lille tidsintervall. Funksjonen er derfor definert i en lengre tid enn vi ofte tenker over.

```

%*****
% Første del: Definerer funksjonen vi skal analysere
%*****
tmin = 0.0; % Tid ved begynnelse av første tidsintervall
tmax = 12.0; % Tid ved slutten av siste tidsintervall
N = 8; % Antall punkter funksjonsverdien skal angis
t = linspace(tmin, tmax*(N-1)/N, N); % Tidspunktene der funksjonen er definert
y = [-3.0 -2.0 -1.0 0.0 1.0 2.0 -0.5 -3.0]; % Vår valgte funksjon
%y = sin(2*pi*(1/tmax)*t); % Alternativt valg av funksjon
plot(t,y,'-k'); % Funksjonen plottes i sine N punkter
xlabel('Tid (s)');
ylabel('Funksjon (vilkarlig enhet)');
dt = 0.1*(tmax-tmin); % Detaljer for å få ønskede akser
ymin = min(y); % (uvesentlig i vår sammenheng)
ymax = max(y);
dy = 0.1*(ymax-ymin);
axis([tmin-dt tmax+dt ymin-dy ymax+dy]);

```

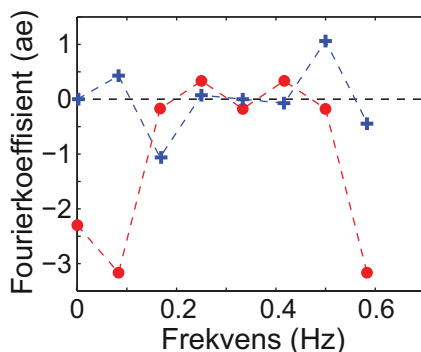
Merk at selve funksjonsverdiene er åtte reelle tall. De kan f.eks. representere en avstand målt i meter, spenning målt i volt eller hva som helst. Vi bruker bare betegnelsen “aktuell enhet” (ae), som kan stå for meter, volt eller hva det nå skulle være.

Funksjonens argument kan være så mangt. Vi har valgt at funksjonen som varierer med tiden (benevning sekunder). Måletall og enheter langs x- og y-aksen er isolert sett totalt uavhengig av hverandre i vårt oppkonstruerte eksempel. Virkelige enheter er bestemt av hva slags fysisk måling dataene $\{x_n\}$ representerer.

I høyre del av figur 4.5 markerer en viktig detalj. Dersom det er åtte målepunkter tatt med 1.5 s mellomrom, er det bare $(8-1)*1.5 \text{ s} = 10.5 \text{ s}$ mellom første og siste tidspunkt der målingen er angitt. Det er fort gjort å tro at våre åtte målepunkter svarer til en tidsperiode på 10.5 s. Det er feil. Hver måleverdi representerer et helt lite tidsintervall. Total måleperiode svarer til det halvåpne tidsintervallet $[0,12) \text{ s}$,

4.6.1 Fouriertransformasjonen

Fouriertransformeres funksjonen, får vi resultatet gitt i figur 4.6. Matlab-koden som er brukt er gitt nedenfor. Her er det flere detaljer å kommentere.



Figur 4.6: *Fouriertransformasjonen til funksjonen i forrige figur. Fouriertransformasjonen gir komplekse tall. Realverdiene er gitt ved røde sirkelskiver og imaginærverdiene ved blå kryss. Hvert punkt svarer til et lite frekvensintervall. Totalt frekvensområde strekker seg derfor lenger ut enn bare mellom punktene isolert sett.*

```
%*****  
% Andre del: Fouriertransformerer funksjonen og plotter denne  
%*****  
z = (1/sqrt(N))*fft(y);           % Selve fouriertransformasjonen  
figure;  
T = tmax - tmin;                 % Beregning av hvilke frekvenser de  
deltaF = 1.0/T;                 % forskjellige komponentene representerer  
for i = 1:N  
    f(i) = deltaF*(i-1);  
end;  
plot(f,imag(z),'.-b');          % Plotting av realdel og imaginærdel av  
hold on;                         % komponentene  
plot(f,real(z),'.-r');  
%plot(f,abs(z),'.-k');          % Plotter absoluttverdi også (iblant)  
xlabel('Frekvens (Hz)');  
ylabel('Fourierkoeffisient (vilkårlig enhet)');
```

For det første ser vi av programkoden at fouriertransformasjonen baserer seg *bare* på selve funksjonsverdiene. Benevninger er skrelt bort, og tidspunktene hvor funksjonen er definert er totalt irrelevant for selve transformasjonen. Resultatet fra transformasjonen er åtte komplekse tall, og realverdi og imaginærverdi for disse åtte tallene er angitt i figuren.

I figur 4.6 er det angitt verdier og benevning langs x-aksen. Dette er verdier vi selv må bestemme ut fra hva vi vet om verdiene langs x-aksen i tidsbildet (som i figur 4.5). Sammenhengen er som følger:

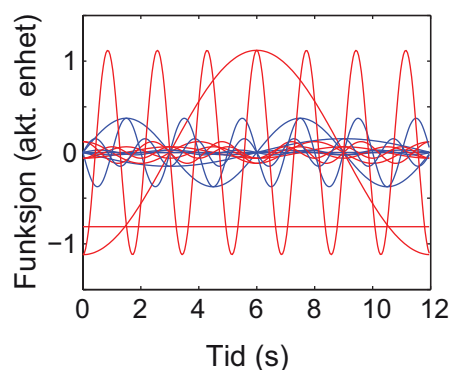
Dersom funksjonen i tidsbildet er definert i N punkter med tidsforskjell Δt , tilsammen over en total tid $T = N\Delta t$, vil funksjonen i frekvensbildet være definert i N punkter med en frekvensforskjell $\Delta f = 1/T$ og representere et totalt (halvåpent) frekvensintervall fra null til $N/T = 1/\Delta t = f_s$ der f_s er samplingsfrekvensen..

Tallene langs y-aksen i et frekvensbilde av en funksjon avhenger blant annet av hvilken faktor som er brukt i transformasjonen (vi har i ligning (4.18) valgt faktoren $1/N$). Ofte angis derfor funksjonen i frekvensbildet bare i relative verdier. Det er imidlertid ganske enkelt å angi fourierkoeffisientene som amplitudeverdier med samme enhet som den opprinnelige funksjonen $\{x_n\}$.

4.6.2 Tidspopløsningen, noen kommentarer

Fra ligning (4.21) vet vi at den omvendte fouriertransformasjonen svarer til summasjon av cosinus- og sinusfunksjoner med et heltall perioder. Amplitudene på cosinusfunksjonene er gitt fra realdelen av fourierkoeffisientene X_k , mens amplituden på sinusfunksjonene er gitt ved imaginærdelen av koeffisientene.

La oss teste dette ut i praksis. I figur 4.7 har vi plottet alle de harmoniske signalene vi får når realdeler og imaginærdel av alle koeffisientene X_k multipliseres med henholdsvis cosinus- og sinusfunksjoner i henhold til enkeltleddene i ligning (4.21). I vårt eksempel var realdelen av X_2 (og dermed også realdelen av X_8) størst slik at disse signalene dominerer litt over de øvrige. Vi ser at disse to svarer til henholdsvis én og syv hele perioder over tidsstrengen.



Figur 4.7: Alle harmoniske funksjoner hver for seg som inngår i summasjonen når funksjonen transformerer fra frekvensbildet til tidsbildet. Se tekst for detaljer.

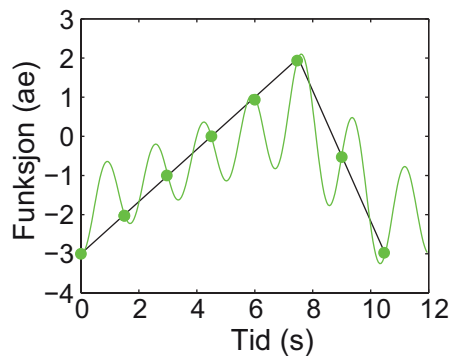
Programkoden for å skrive ut de harmoniske signalene hver for seg er gitt nedenfor.

```

%*****
% Tredje del: Plotter hver enkelt komponent i Fourierspekteret
%*****
nfin = 256; % Ønsker å plotte sin(t) og cos(t)
tfin = linspace(tmin, tmax*(nfin-1)/nfin, nfin); % mellom tidspunkter som brukes
ytotI = zeros(1,nfin);
ytotR = zeros(1,nfin);
figure;
faktor = (2.0*pi/(tmax-tmin));
% Beregner nå etter tur sinus og cosinussignaler som koeffisientene
% representerer, og plotter alle sammen hver for seg (i samme diagram)
for i = 1:N
    ks = - (1.0/sqrt(N))*imag(z(i))*sin(tfin*faktor*(i-1));
    kc = (1.0/sqrt(N))*real(z(i))*cos(tfin*faktor*(i-1));
    plot(tfin,ks,'-r'); % Sinussignalene plottes rødt
    hold on;
    plot(tfin,kc,'-b'); % Cosinussignalene plottes blått
    ytotI = ytotI + ks; % Summerer opp (for fjerde del)
    ytotR = ytotR + kc;
end;
xlabel('Tid (s)');
ylabel('Funksjon (vilkårlig enhet)');

```

Summerer vi opp alle sinus- og cosinussignaler i figur 4.7, får vi den grønne kurven gitt i figur 4.8. I samme figur har vi også tegnet inn den opprinnelige funksjonen, med eksakt samme skalering både i x- og y-retning som det grønne sumsignalet.



Figur 4.8: *Summen av alle harmoniske funksjoner som fourierspekteret svarer til når vi transformerer funksjonen tilbake til tidsbildet sammen med opprinnelig funksjon. I tidspunktene den opprinnelige funksjonen er definert, har summen av de harmoniske signalene identisk verdi med den opprinnelige funksjonen, men ikke mellom disse punktene. Se tekst for detaljer.*

Det ser ut for at det slett ikke er samsvar mellom den opprinnelige funksjonen $\{x_n\}$ og resultatet etter en vanlig og en invers fouriertransformasjon etter hverandre. Her er det imidlertid en viktig detalj å merke seg! Sinus og cosinusfunksjonene vi har plottet i figur

4.7 er plottet for “alle” tidspunkt, ikke bare de tidspunktene den opprinnelige funksjonen er angitt i. Dette er noe vi har lagt inn i vår egen presentasjon for å få fram et viktig poeng. Detaljer er gitt i koden ovenfor.

Vi ser at dersom vi begrenser oss til å angi sumsignalet i *bare* nøyaktig de tidspunktene som den opprinnelige funksjonen ble angitt i, får vi nøyaktig samme resultat som den opprinnelige funksjonen! Utenfor disse tidspunktene er det *ikke* samsvar mellom vår heltrukne kurve mellom de opprinnelige målepunktene og sumfunksjonen av harmoniske funksjoner. Det er imidlertid nokså meningsløst å sammenligne funksjonsverdier der funksjonen faktisk ikke er definert.

Programkoden for å plote summen av alle harmoniske funksjoner i den omvendte transformasjonen, er gitt nedenfor.

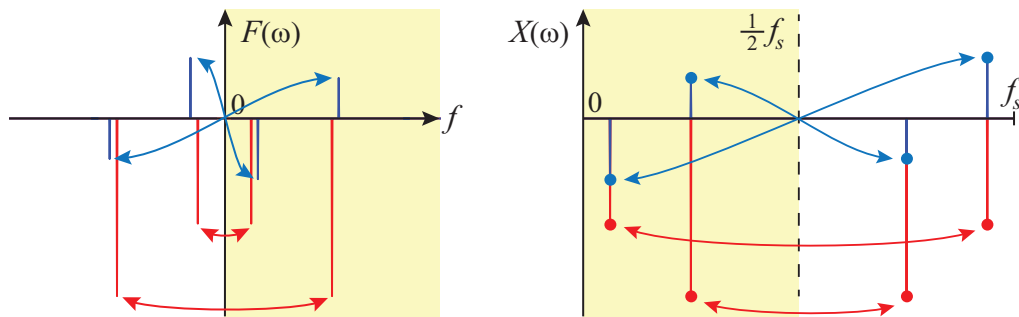
```
%*****
% Fjerde del: Plotter summen av alle komponentene i Fourierspekteret
%*****
figure;
plot(t,y,'.-k');          % Plotter opprinnelig funksjon (n1 punkt)
hold on;
ytot = ytotI+ytotR;       % Plotter sum av sinus og cosinussignaler
plot(tfin,ytot,'-g');
xlabel('Tid (s)');
ylabel('Funksjon (vilkårlig enhet)');
```

4.6.3 Speiling / folding

Vi viste tidligere (ligning (4.12)) i et spesialtilfelle at for en kontinuerlig fouriertransformasjon gjelder det at $F(\omega_0) = F^*(-\omega_0)$, det vil si at den fouriertransformerte ved en vinkelfrekvens er den komplekse konjugerte av den fouriertransformerte ved den negative vinkelfrekvensen. For diskret fouriertransformasjon har vi laget indeksene slik at vi slipper å arbeide med negative frekvenser. Likevel kan vi gjenfinne oppsplittingen i det fouriertransformerte signalet også ved diskret fouriertransformasjon. Figur 4.9 forsøker å skissere dette.

Årsaken til symmetrien i figur 4.9 for diskret fouriertransformasjon, har den oppmerksomme leser kanskje allerede oppdaget. I figur 4.3 ser vi at for cosinusdelen av $e^{-i\frac{2\pi}{N}kn}$ er punktene identiske for $k = 1$ som for $k = N - 1$, og tilsvarende identiske for $k = 2$ som for $k = N - 2$. Generelt er cosinusleddene de samme for $k = m$ som for $k = N - m$ for $m = 1, \dots, N/2$. (NB: $k = 0, 1, 2, \dots, N - 1$ i denne figuren.)

For sinusdelen av $e^{-i\frac{2\pi}{N}kn}$ er punktene identiske, men med motsatt fortegn, for $k = 1$ som for $k = N - 1$, og det samme gjelder generelt for $k = m$ sammenlignet med $k = N - m$ for $m = 1, \dots, N/2$. Det betyr nettopp at $e^{-i\frac{2\pi}{N}kn}$ og $e^{-i\frac{2\pi}{N}(N-k)n}$ er komplekst konjugerte av hverandre, og dermed ser vi ut fra ligning (4.18) at $X_k = X_{N-k}^*$ for $k = 1, 2, \dots, N/2$. Det



Figur 4.9: Til venstre: Et fourierspekter ved kontinuerlig fouriertransformasjon av et signal som varer ved i det uendelige, inneholder alle frekvenser mellom $-\infty$ og $+\infty$, men det er en “speiling/folding” og kompleks konjugering omkring frekvensen null. Reell del av den fouriertransformerte funksjonen er markert i rødt, den imaginære i blått. Ved diskret fouriertransformasjon av et signal (høyre del) som bare er definert i et endelig antall punkter i en begrenset tid, inngår bare frekvenser mellom 0 og (nesten) samplingsfrekvensen f_s . Her får vi også en speiling/folding og kompleks konjugering, men denne gang omkring halve samplingsfrekvensen $f_s/2$. (Vi har forskjøvet de reelle i forhold til de imaginære punktene noe i venstre del for at alt skulle bli lett synlig.) Den delen av diagrammene som har en lett bakgrunnsfarge, inneholder all informasjon i det fouriertransformerte signalet siden den øvrige halvparten bare er den kompleks konjugerte av den første.

betyr at vi har en klar symmetri omkring frekvensen $f_{N/2}$ (for $k = 0, 1, 2, \dots, N - 1$), det vil si halve samplingsfrekvensen.

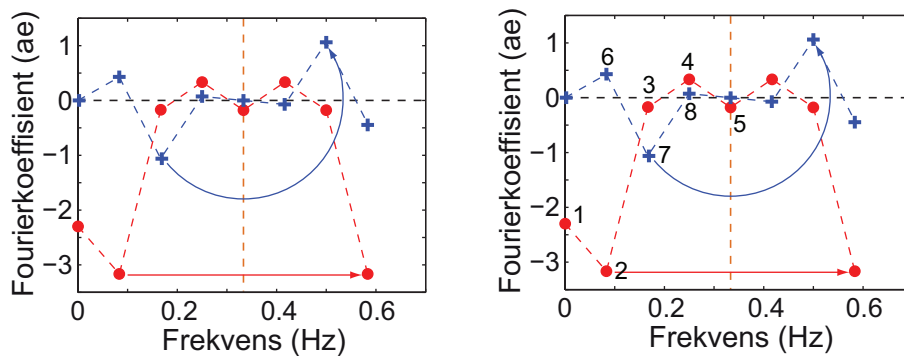
[♠ \Rightarrow Merk at i en diskret fouriertransformasjon får vi like mange tall ut som tall inn. Det vil si, det kan diskuteres! Dersom tidsbeskrivelsen $\{x_n\}$ er reell, er det nok med N reelle tall for å angi tallrekken $\{x_n\}$. Det må imidlertid $2N$ reelle tall til for å angi $\{X_k\}$ fordi $\{X_k\}$ er komplekse. Det betyr at ikke alle X_k kan være uavhengige av hverandre i slike tilfeller. Halvparten av alle X_k er egentlig overflødige når $\{x_n\}$ er reelle. Vi vil se nærmere på dette ved å gå tilbake til eksemplet vårt i figur 4.6.

Vi ser en symmetri i frekvensbildet: $X_k = X_{N+2-k}^*$ for $k = 2, 3, \dots, N/2 + 1$ (forutsatt av vi opererer med Matlab-indeks som starter på 1, dvs. $k = 1, 2, \dots, N$). Dette kommer fram i venstre del av figur 4.10. Realdelen av tallene har samme verdi etter refleksjon (speiling) omkring linjen $f_{N/2+1}$. Imaginærdelene av tallene har en rotasjon 180 grader rundt punktet $(f_{N/2+1}, 0)$. Denne formen for symmetri svarer nettopp til $f_k = f_{N+2-k}^*$.

Det er imidlertid et klart unntak fra symmetrieregelen. Første tall i frekvensbildet er noe for seg selv. Det svarer til en frekvens lik null, med andre ord til en konstantverdi. Konstantverdien er alltid reell når den opprinnelige funksjonen er reell.

I vårt tilfelle startet vi ut med åtte reelle tall og foretok en transformasjon som på en entydig måte skal inneholde all informasjon i den opprinnelige representasjonen. Da kan det ikke være mer enn åtte frihetsgrader i det endelige resultatet.

På grunn av “speilingen” omkring punktet ved frekvensen $f_{N/2+1}$, er det klart at alle punkter over denne frekvensen følger lovmessig av verdiene til punktene under denne frekvensen. Videre vet vi at ved frekvensen



Figur 4.10: Venstre: Fouriertransformasjonens komplekse tall viser en form for symmetri omkring frekvensen $f_{N/2+1}$. “ae” står for “aktuell enhet” (samme enhet som opprinnelige signal, men iblant kombinert med en skaleringsfaktor). Se teksten for detaljer. Høyre: Av de $2N$ punktene som fouriertransformasjonen fører til, er det bare N uavhengige verdier (markert med numrene 1 - 8). Dette er som forventet siden vi startet ut nettopp med N reelle verdier.

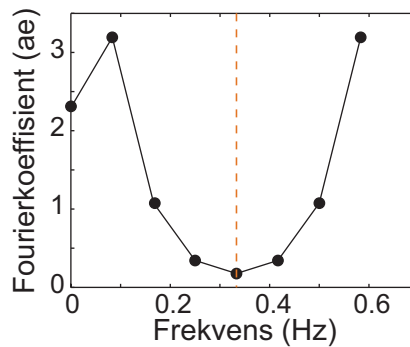
null er imaginærverdien alltid null. På grunn av symmetrien vil det samme gjelde for punktet ved $f_{N/2+1}$. Vi ser da av høyre del av figur 4.10 at vi står igjen med $N/2 + 1$ reelle verdier og $N/2 - 1$ imaginære verdier som er uavhengige av hverandre, tilsammen N uavhengige verdier. Transformasjonen utspenner da et funksjonsrom som er akkurat like stort som det vi startet ut med, og det er tilfredsstillende å konstatere. Dette gjelder også i de tilfellene der vi starter ut med en kompleks funksjon i tidsbildet, men detaljene blir da til dels ganske annerledes enn i vårt tilfelle. ← ♠]

Dersom vi bare betrakter absoluttverdien av fourierkoeffisientene X_k , bør de være symmetrisk rundt frekvensen $f_{N/2+1}$ (for $k = 1, 2, \dots, N$). Koeffisienten som representerer null frekvens, det vil si en konstant-funksjon, er ikke med i denne symmetrien. Dersom vi plottet bare absoluttverdiene av fourierkoeffisientene i figur 4.6, får vi figur 4.11. Vi ser klart symmetrien rundt $f_{N/2+1}$, og at konstantfunksjonsleddet er noe for seg selv. Symmetrien vi ser kaller vi “speiling” eller “folding”.

Ved fourieromvending av et reelt signal $\{x_n\}$ for $n = 0, \dots, (N - 1)$, er det bare de første $N/2 + 1$ koeffisientene som er av interesse. De representerer frekvenser fra null (konstantledd) opp til og med halve samplingsfrekvensen. De øvrige koeffisientene er bare kompleks konjugerte av de koeffisientene vi faktisk bruker.

4.6.4 Samplingsteoremet

Det er nok en grunn til at vi ikke er særlig interessert i den øvre halvdel av det symmetriske forløpet i frekvensbildet. Vi så et eksempel på dette allerede i figur 4.3. I figur 4.12 er det



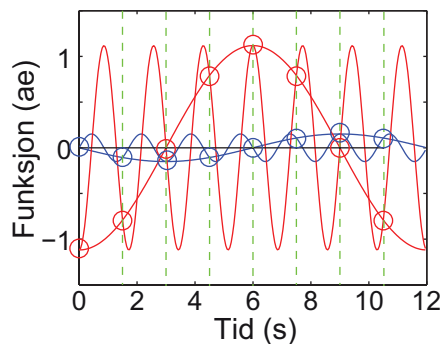
Figur 4.11: Absoluttverdien av fourierkomponentene viser klart speilingen eller foldingen omkring frekvensen $f_{N/2+1}$.

vist et lignende diagram, men denne gang viser vi de fire største sinus og cosinussignalene som tilsammen gir den skjeve sagtannfunksjonen vår i figur 4.5.

Funksjonene er tegnet ved “alle” tidspunkt, men tidspunktene der den opprinnelige funksjonen er definert er markert med vertikale stiplede linjer. Vi ser da at funksjonene med svært forskjellig frekvens likevel har nøyaktig samme verdi i disse tidspunktene, selv om verdiene utenfor disse tidspunktene er vidt forskjellige. Dette er i samsvar med ligning

$$e^{i\frac{2\pi}{N}kn} = e^{-i\frac{2\pi}{N}k(N-n)} \quad (4.29)$$

for k og $n = 1, \dots, N - 1$ i det tilfelle at disse indeksene generelt går fra 0 til $N - 1$.



Figur 4.12: De harmoniske funksjonene med frekvensene f_k og f_{N-k} (her: $k = 1$) har nøyaktig samme verdi akkurat i de tidspunktene den opprinnelige funksjonen var definert (forutsatt at $k = 0, 1, \dots, N - 1$). Vi kan derfor ikke skille mellom de to ved dette valget av tidspunkt der funksjonen er definert (for den samplingsfrekvensen som ligger bak). For å kunne skille funksjoner med ulik frekvens, må samplingsfrekvensen være minst dobbelt så stor som høyeste frekvenskomponent.

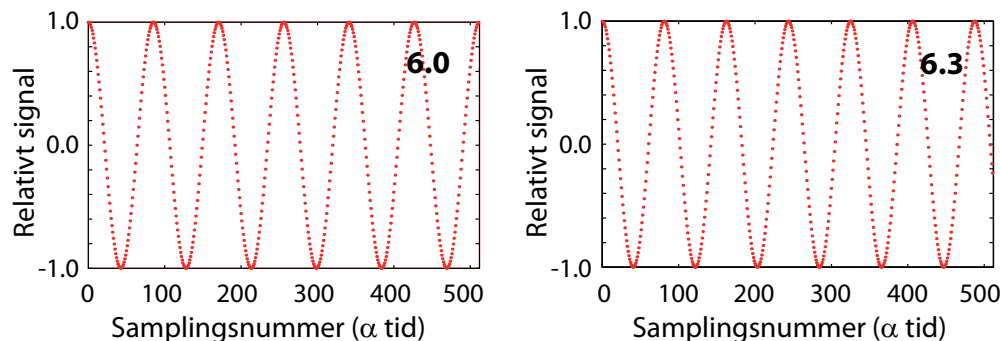
Dette er et eksempel på et generelt prinsipp:

Skal vi representere en harmonisk funksjon på en entydig måte ved et begrenset antall målinger, må måletettheten (målefrekvensen, samplingsfrekvensen) være så stor at vi får minst to målinger innen hver periode på det harmoniske signalet. “Nyquist-Shannons samplings-teorem” sier dette på en mer kompakt måte: Samplingsfrekvensen må være minst dobbelt så stor som høyeste frekvenskomponent i et signal for at et samplet signal skal gi et entydig bilde av signalet. Dersom det kan forekomme høyre frekvenser i det opprinnelige signalet, må disse filtreres bort før sampling for at resultatet skal bli entydig.

4.7 * En finurlighet

I høyre del av figur 4.2 så vi at en *kontinuerlig* fouriertransformasjon av en perfekt sinus eller cosinus gir “skarpe linjer” (“deltafunksjon”). Vi kan ledes til å tro at det alltid er slik, dersom signalet er en ren sinus, men for en diskret fouriertransformasjon er dette ikke slik, generelt sett. Vi får et sammenlignbart resultat bare dersom sinus-signalet vi starter ut med har *eksakt* et helt antall perioder innenfor samplingstiden vår.

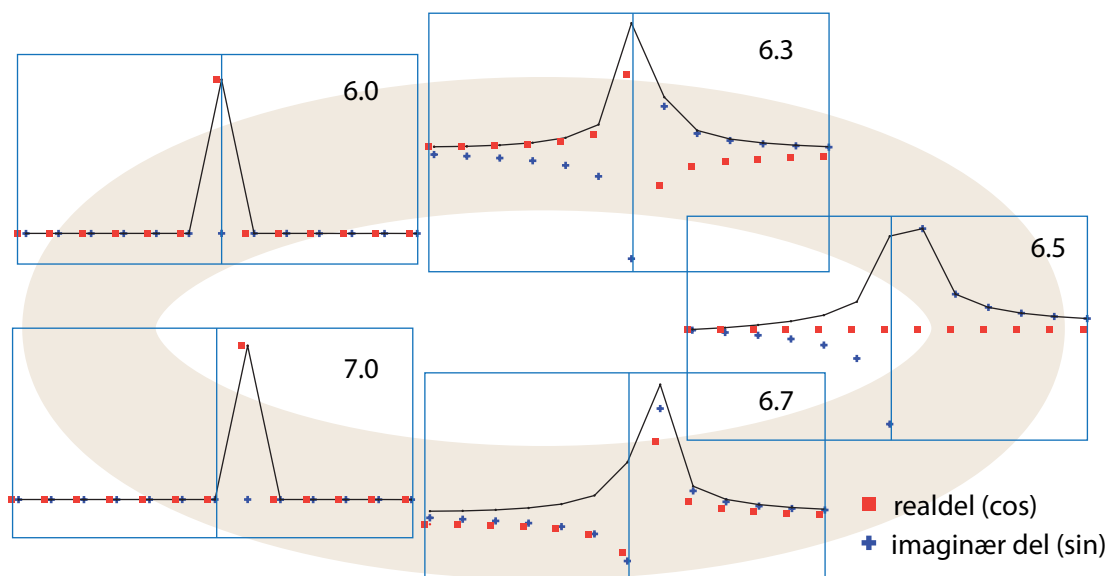
Figur 4.13 og 4.14 viser hvordan et fourierspekter ser ut dersom cosinus-signalet vi betrakter ikke har nøyaktig et heltall perioder innenfor den totale samplingstiden. Til venstre i figur 4.13 er det eksakt 6 hele perioder av signalet innenfor de 512 samplingsene som er foretatt. Til høyre er det 6.3 perioder innen det samme intervallet.



Figur 4.13: Eksempel på signal med eksakt (6.0) og ikke eksakt (6.3) antall hele perioder innenfor den totale samplingstiden. Signalet er samplet i 512 tidspunkter.

Figur 4.14 viser de første punktene i fourierspekteret. Røde firkanter svarer til cosinusleddene (reelle delen av fourierspekteret), mens de blå kryssene svarer til sinusleddene (imaginære delen av fourierspekteret). Den heltrukne sorte linjen svarer til amplituden bestemt fra ligning (4.23). Denne kontinuerlige kurven kan virke litt misvisende siden det ikke finnes noen verdier mellom de diskrete tallene.

Vi ser at fourierspekteret for et signal med seks hele periodetider innenfor samplingsperioden er som forventet. Den reelle delen av tall nummer 7 i rekken er det eneste tallet i



Figur 4.14: De første punktene i Fourierspekteret av signaler som i figur 4.13. Amplituder for cosinusfunksjoner (reell del av fourierspekteret) er markert med røde firkanter, mens blå kryss gir amplituder for sinusbidragene (imaginær del). Se tekst for detaljer.

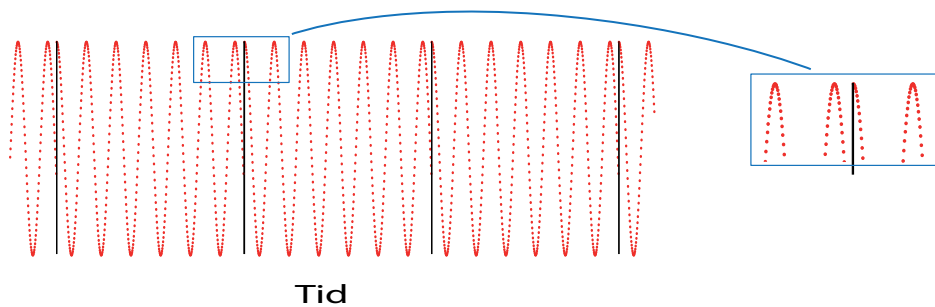
frekvensspekteret forskjellig fra null. Dette er rimelig siden vi jo kan beskrive det samplede signalet ved hjelp av én av frekvensene $f_{k+1} = k/T$, nemlig den hvor $k = 6$. Vi trenger ikke noe mer!

Når signalet har 6.3 perioder innenfor den totale samplingstiden blir ting annerledes. Frekvensspekteret inneholder en rekke tall forskjellig fra null, men alle tallene ligger i nærheten av punkt nummer syv. Det er bidrag både fra den reelle og imaginære delen.

Frekvensspekteret alene kunne tolkes dithen at signalet, allerede da vi laget det, besto av flere bidrag som alle har nøyaktig et helt antall bølgelengder innenfor T . I virkeligheten vet vi at signalet vi startet ut med var helt rent med én eneste frekvens. Problemet er bare at samplingstiden som ble valgt svarer til 6.3 perioder innenfor den totale samplingsperioden (ikke et heltall perioder). Dette er en svakhet ved diskret fouriertransformasjon som vi må ha i bakhodet når vi vurderer et frekvensspektrum.

Pragmatisk sett er det ikke så rart at det blir slik. Det finnes ingen basisfunksjon i den diskrete transformasjonen som svarer eksakt til 6.3 perioder innenfor den totale samplingstiden. Men signalet SKAL kunne gjengis 100 prosent riktig innenfor det aktuelle tidsintervallet etter at vi foretar en invers fouriertransformasjon av frekvensspekteret. Da må vi mikse nærliggende frekvenser og ha korrekte innbyrdes faser for å kunne matche det opprinnelige signalet.

Dette kan vi kanskje forstå enda bedre dersom vi går tilbake til forutsetningene for fourierrekker, nemlig at *signalet skal være periodisk med periode T* . Det betyr at signalet vi



Figur 4.15: Signalet vi i virkeligheten analyserer når vi bruker fourierrekke-analyse på signalet til høyre i figur 4.13. Signalet er periodisk med periode lik avstanden mellom de sorte strekene, og strekker seg fra minus til pluss uendelig.

faktisk analyserer når vi bruker en fourier-rekke-analyse av signalet med 6.3 perioder innenfor tiden vi sampler, ser ut som vist i figur 4.15. Da innser vi at et slikt signal kan ikke beskrives ved en enkel sinus! Og det er grunnen til at fourieranalyse inneholder flere komponenter!

Figur 4.14 viser også frekvensspekteret for signaler med 6.5, 6.7 og 7.0 perioder innenfor den totale samplingstiden. Hensikten er å vise at frekvensspekteret endres på en ikke-triviell måte når signalfrekvensen endres.

♠ ⇒ Et talleksempel kan kanskje være av interesse i denne sammenheng:

Anta at vi digitaliserer (sampler) et tidsavhengig signal ved å sample 2^m ganger hvor m er et heltall, nærmere bestemt $N = 512 = 2^9$ punkter. Anta at vi har en samplingsfrekvens f_s på 1.0 kHz. I så fall vil tidsstrengen vi sampler være $T = N/f_s = 512$ ms lang og frekvensoppløsningen i en fouriertransformasjon blir $\Delta f = 1.9531\dots$ Hz.

Oppløsningen i frekvensspekteret er med andre ord bestemt av total samplingstid, mens den maksimale frekvensen i frekvensdomenet er bestemt av den opprinnelige samplingsfrekvensen.

Anta at selve signalet vi samplet har frekvensen 50 Hz, slik at periodetiden T er 20 ms. Det samlede signalet inneholder da $512/20 = 25.6$ perioder. I fourierspekteret vil punkt nr 26 og punkt nr 27 dominere og ha nesten like stor absoluttverdi (antar at indeksene starter ved 1). Også noen nærliggende punkter i frekvensspekteret vil ha verdier forskjellig fra null.

Dersom vi hadde samplet signalet i $2^{14} = 16384$ punkter, ville total samplingstid vært 16.384 sekunder. Frekvensoppløsningen i frekvensbildet ville da blitt 0.061035... Hz, og 50 Hz signalet ville få en klart markant topp i punkt nr 820, men litt også i punkt 821 og i enda mindre grad punktene på begge sider av disse. Totalt sett ville da toppen i frekvensspekteret bli langt mer markant og "skarp" enn da vi bare samplet signalet i 0.512 s.

Vi konstaterer imidlertid i begge tilfeller at selv om signalet hadde en svært ren 50 Hz tidsvariasjon, og at vi samplet med 1000 Hz, som jo svarer nøyaktig til 20 samplinger per periode, er frekvensspekteret likevel ikke skarpt etter en FFT-transformasjon.

← ♠]

4.7.1 Fouriertransformasjon av mer kompliserte signaler

Fouriertransformasjon brukes mye til såkalt frekvensanalyse hvor vi bestemmer hvilke frekvenskomponenter som finnes i et signal. Frekvensspekteret er nyttig fordi det ofte gir et “fingeravtrykk” av de fysiske prosessene som ligger bak signalet vi betrakter.

Eksempelvis viser figur 4.16 et frekvensspekter fra et lydsignal fra en tverrfløyte. I figuren er relative amplituder i frekvensspekteret angitt, se ligning (4.23). Vi mister da faseinformasjonen, men “styrken” på ulike frekvenskomponenter kommer godt fram.

Spekteret består i hovedsak av en rekke topper med litt ulik høyde. Toppenes plassering har en viss lovmessighet. Det finnes en frekvens f_0 (burde kanskje heller vært kalt f_1), den såkalte *grunntonen*, slik at de resterende linjene i en gruppe av linjer tilnærmet har frekvensene kf_0 , der k er et heltall. Vi sier at frekvensene kf_0 for $k > 1$ er *harmoniske av grunntonen*, og vi omtaler dem som “*overtoner*”.

I figur 4.16 ser vi også en linje ved en frekvens nær null. Det er en topp ved 50 Hz, som er nettfrekvensen på lysnettet. Dette signalet har på et eller annet vis sneket seg inn sammen med lyden fra fløyta, kanskje ved at elektronikken har plukket opp elektriske eller magnetiske felter et sted i signalgangen.

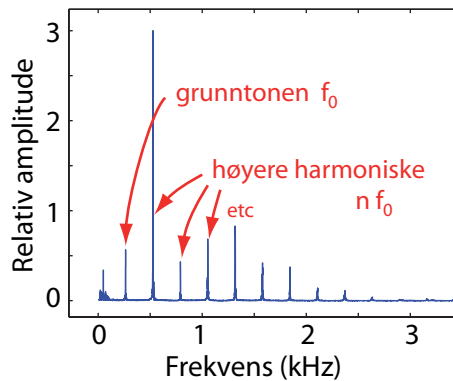
Frekvensspekteret viser at når vi spiller fløyte, vil luften ikke bare svinge ved én bestemt frekvens, men ved flere frekvenser samtidig. Ulike instrumenter kan karakteriseres ved frekvensspekteret av lyden de genererer. Noen instrumenter gir lite overtoner/harmoniske, mens andre (f.eks. obo) gir mange!

Frekvensspekteret kan benyttes som utgangspunkt også ved syntese av lyd. Siden vi kjenner intensitetsfordelingen i frekvensspekteret til fløyta, kan vi i prinsippet starte med denne fordelingen og foreta en invers fouriertransformasjon for å generere lyd som høres ut omtrent som en fløyte.

Det må likevel bemerkes at vårt inntrykk av lyd ikke bare bestemmes av frekvensspekteret for et vedvarende lydsignal, men også av hvordan lyden starter og dør ut. I den sammenhengen er fouriertransformasjon til lite hjelp.

La oss nevne noen få andre eksempler på bruk av fouriertransformasjon:

- Dersom vi spiller i et band og bruker ulike forsterkere, ønsker vi at signalet som kommer f.eks. fra en mikrofon skal bli forsterket uten å bli forvrengt. En type forvrengning viser seg ved at det dukker opp høyere harmoniske av de opprinnelige frekvensene. Ved å sende et rent sinussignal inn på forsterkeren, og digitalisere lyden etter at signalet har gått gjennom forsterkeren, kan vi ved hjelp av fouriertransformasjon undersøke om det er kommet noe ekstra signal til som ikke burde vært der.
- Ved å analysere lysintensiteten fra enkelte stjerner er det oppdaget periodiske variasjoner. Ved en fouriertransformasjonsanalyse kan vi bestemme periodetiden og vi kan



Figur 4.16: Et frekvensspekter av lyd fra en tverrfløyte. Amplitudeverdier (intensiteter) er vist.

få et mål for intensitetsfluktasjonen som kan arkiveres for å se på langtidsendringer på sikt.

- Solflekkaktiviteten er selvfølgelig også en tidsavhengig størrelse som lar seg analysere ved hjelp av fouriertransformasjon.
- Fouriertransformasjon kommer også inn for å finne hvor høye frekvenser et system må kunne håndtere for at vi skal kunne sende/motta digitale pulser (firkantpulser) uten at pulsene skal deformeres så mye at det går ut over lesbarheten.
- Fouriertransformasjon kan også brukes om romlige forhold i stedet for tidsavhengige forhold. I optikken brukes nå ofte uttrykket fourieroptikk.
- En fjærpendel som beskrevet i kapittel 1 svinger harmonisk såfremt Hookes lov gjelder og friksjonen er neglisjerbar. Ved fouriertransformasjon av posisjon versus tid, kan vi oppdage når bevegelsen ikke lenger er harmonisk, og vi kan utnytte denne muligheten i ulike typer studier.

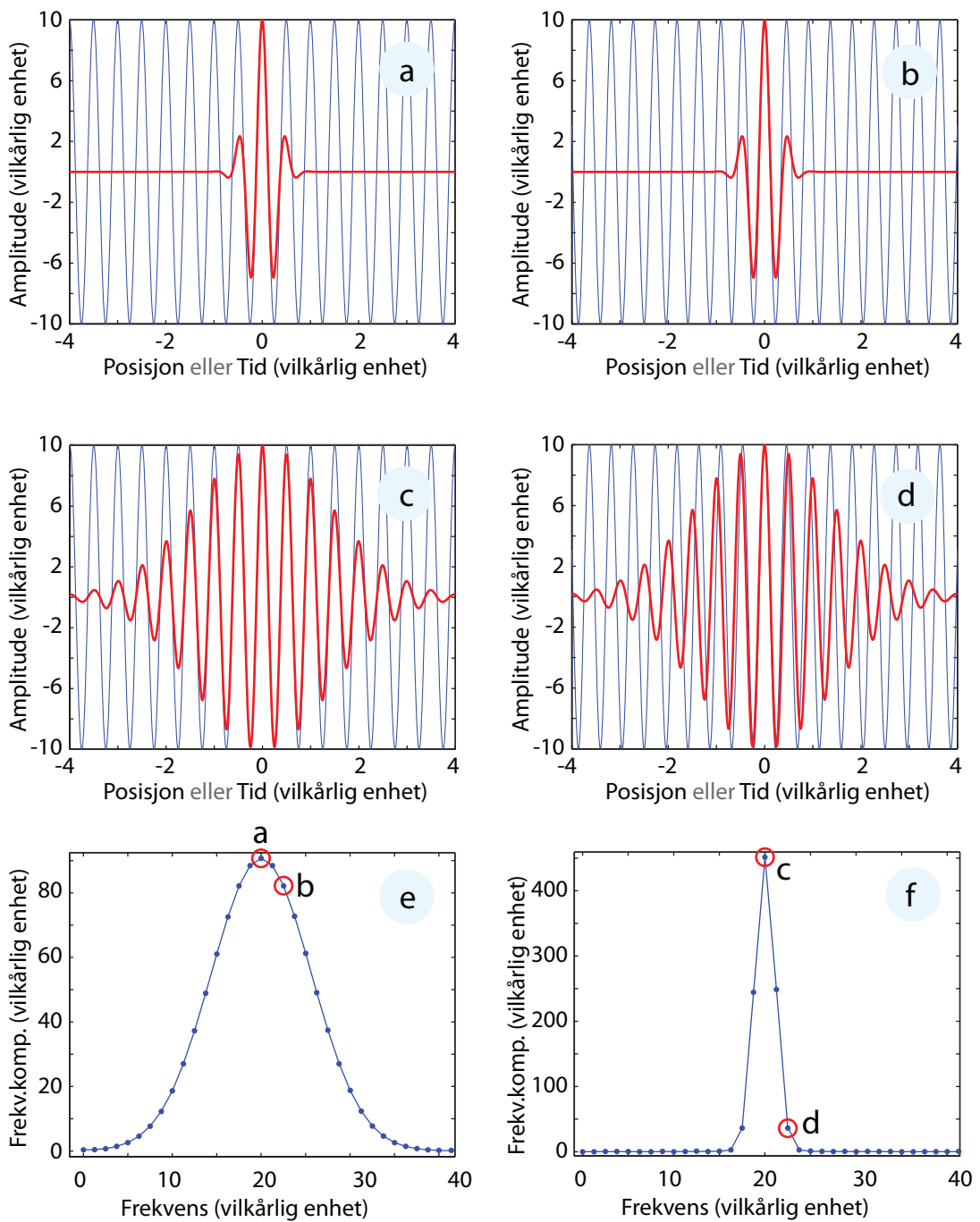
En mengde detaljer knyttet til fouriertransformasjon finner du på f.eks. Wikipedia.

4.8 Tidsbegrenset signal

Fra ligning (4.20) går det fram at en fouriertransformasjon egentlig er en sum (integrasjon) av produktet mellom signalet som skal transformeres og en ren sinus eller cosinus. Figur 4.17 viser med røde kurver to ulike signaler som er tidsbegrensede. Det er rett og slett vår tidsbegrensede kraft fra kapittel 2 som er vist.

Matematisk er kraften gitt som:

$$F(t) = F_0 \cos(\omega(t - t_0))e^{-((t-t_0)/\sigma)^2}$$



Figur 4.17: *Fouriertransformasjon av et cosinussignal som er konvolutert med en gaussisk funksjon. Bare en liten del av det totale frekvensspekteret er vist. Se teksten for detaljer.*

hvor σ angir varigheten på kraften (fra amplituden har hatt sin maksimale verdi til amplituden har sunket til $1/e$ av max). ω er vinkelfrekvensen til den underliggende cosinusfunksjonen, og t_0 er tiden der kraften har maksimal amplitude (toppen av kraftpulsene forekommer ved tiden t_0).

I a og b er kraften kortvarig (σ liten), mens i c og d varer kraften lenger tid (σ fem ganger så stor som i a og b).

I a og c har vi i tillegg til kraftpulsene i rødt, tegnet inn cosinussignalet med frekvens eksakt lik $\omega/2\pi$ med blå tynnere strek. I b og d har cosinussignalet 10 % høyere frekvens.

Vi ser at integralet (summen) av produktet mellom den røde og blå kurven i a og i b vil være omtrent det samme. Derimot ser vi at det tilsvarende integralet for d må være betydelig mindre enn integralet for c siden kraft-signalet og cosinussignalet kommer i motfase litt vekk fra sentrum av pulsen i d.

Dersom vi foretar en fouriertransformasjon av selve den røde kurven i a (den kortvarige kraften), og tar absoluttverdien av fourierkoeffisientene, blir resultatet som i e. Fouriertransformasjonen av kraften i c (kraften som varer litt lenger), er vist i figurens nedre høyre hjørne f. Vi ser at fouriertransformasjonen fanger opp de prediksjonene vi kunne gjøre ut fra figurene a til d.

Merk at det kortvarige signalet ga et bredt frekvensspekter, mens signalet med flere perioder i den underliggende cosinusfunksjonen, ga et smalere frekvensspekter. Dette er på ny en manifestasjon av prinsippet vi har vært borti tidligere, og som har klare likheter med Heisenbergs uskarphetsrelasjon.

[♠ \Rightarrow I kapittel 2 sammenlignet vi frekvensresponsen til et svingende system med frekvensspekteret til kraften. Poenget var at dersom en påtrykt kraft virker lenge nok, vil systemets frekvensrespons være bestemt av Q-verdien til systemet. Derimot, dersom kraftpulsene varte svært kort tid, ville frekvensresponsen til systemet flyte mer ut. I ekstreme tilfeller vil bredden på frekvensresponsen til systemet bli identisk med bredden til frekvensspekteret av kraften alene. I figur 4.17 er det kun frekvensspekteret til kraften vi betrakter!

\Leftarrow ♠]

4.9 Til ettertanke

Resultater slik vi ser i figur 4.17 kan lett føre til alvorlige feiltolkninger. I a ser vi at kraften varer kun en meget kort tid (få periodetider). Resten av tiden er kraften rett og slett null (eller vi kunne satt den til eksakt lik null med ingen nevneverdig forskjell i frekvensspekteret).

Hva viser fouriertransformasjonen? Fra delfigur e kan vi se at det er om lag 30 frekvens-

komponenter som er klart forskjellig fra null. Det betyr at vi må ha i størrelsesorden 30 ulike sinus- og cosinus-funksjoner *som varer ved HELE tiden* for å beskrive det opprinnelige signalet (jamfør ligning (4.21)).

Noen trekker da den slutning at *egentlig* så er kraften ikke null der den ser ut for å være null, men rett og slett summen av om lag 30 ulike sinus- og 30 ulike cosinusfunksjoner overalt i hele tiden. Dette er vrøvl!

Det er korrekt at vi kan beskrive kraften i delfigur a ved hjelp av alle disse sinus- og cosinusfunksjonene, men det er en ren matematisk greie og har lite med fysikk å gjøre. Ikke for det, det er en del fysikk og fysiske realiteter som samvarierer med bredden på frekvensspekteret. Men det finnes andre metoder å få fram dette poenget på uten at det impliserer at det er noe fysisk til stede den tiden kraften faktisk er lik null. Vi skal i et senere kapittel ta for oss såkalt wavelet-transformasjon, og da vil dette komme bedre fram.

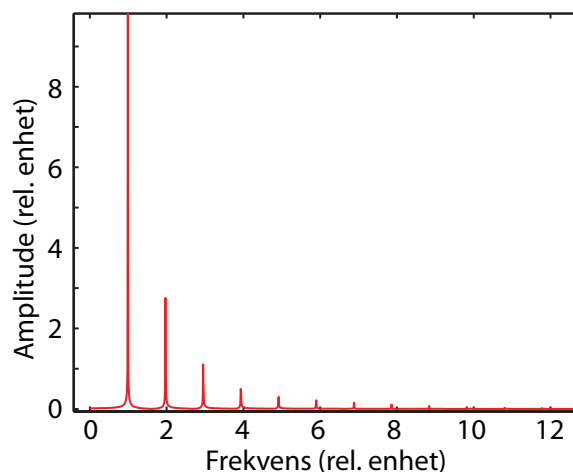
[♠ ⇒ I mitt eget forskningsfelt, kvanteoptikk, ser vi hvor uheldig denne type kortslutning er. Noen sier at vi må “bruke mange ulike fotoner” for å skape en lyspuls, og at hvert foton må ha energien $E = hf$ hvor h er Plancks konstant og f frekvensen. Da tillegges det en fysisk virkelighet til hver enkelt fourierkoeffisient, mens det burde vært mer fokus på hva som er fysikk og hva som er matematikk. ⇐ ♠]

Et vesentlig poeng her er at all tidsinformasjon om et signal forsvinner så snart vi tar absoluttverdien av fourierkoeffisientene. Så lenge vi beholder komplekse fourierkoeffisienter, er tidsinformasjonen intakt, men den er ofte svært godt gjemt. Tidsinformasjonen er nemlig spredt ut over hele fourierspekteret. Det er bare en full tilbaketransformasjon fra frekvensbildet til tidsbildet som gir oss tidsinformasjonen tilbake. Fouriertransformasjon har derfor begrenset verdi for signaler som er null i enkelte tidsperioder eller som på annet vis helt endrer karakter i løpet av samplingstiden.

Også i en annen sammenheng er kan en fourieranalyse føre til uheldige konklusjoner. Figur 4.18 viser en fouriertransformasjon for en periodisk bevegelse. På sett og vis ligner denne figuren på figur 4.16 som viste frekvensspekteret av lyd fra en tverrfløyte. I den anledning omtalte vi de høyere harmoniske ved å si at “når vi spiller fløyte, vil lufta ikke bare svinge ved én bestemt frekvens, men ved flere frekvenser samtidig”. En slik formulering er vanlig, men ikke problematisk.

Sier vi at “flere frekvenser finnes samtidig” i bevegelsen som ligger bak fourierspekteret i figur 4.18, passer utsagnet dårlig i forhold til fysikken som ligger bak! Figuren ble nemlig laget slik: Vi beregnet først en planets bane rundt Sola. Banen ble beskrevet ved et sett koordinater som funksjon av tid $[x_i(t), y_i(t)]$. Figur 4.18 er da rett og slett den fouriertransformerte av $\{x_i(t)\}$ for en tid som er mange runder rundt sola for den aktuelle planeten.

Grunnen til at vi får en rekke “harmoniske” i dette tilfellet, er at fouriertransformasjon



Figur 4.18: *Fouriertransformasjon av en periodisk bevegelse. Se teksten for forklaring.*

er basert på harmoniske funksjoner, som svarer til sirkelbevegelser. Planetbevegelsen er imidlertid ikke sirkelformet, men elliptisk! Fouriertransformasjonen er rett og slett ikke spesielt egnet for analyser av elliptiske fenomener.

Onde tunger sier at dersom vi hadde hatt datamaskiner på Keplers tid og fouriertransformasjonen var tilgjengelig, ville vi fortsatt operert med middelalderens *episykler* den dag i dag. For fourieranalysen vår i figur 4.18 viser nettopp at vi kan erstatte ellipsen med en rekke sirkelbevegelser med passe amplitude (og fase). De fleste vil imidlertid være enige i at det er bedre å bruke en beskrivelse av planetbevegelse basert på ellipser og ikke sirkler. Jeg skulle ønske at vi var like åpne for å droppe matematisk formalisme basert på fourieranalyse også i enkelte andre sammenhenger.

Fourieranalyse kan gjennomføres for praktisk talt alle fysiske tidsvariable systemer, siden settet med sinus- og cosinusfunksjoner som inngår i analysen danner et fullstendig sett med funksjoner. Pass på at du *ikke* av dette trekker slutningen at “når noe er mulig, så er det også gunstig”. I kapitlet om wavelet-transformasjon vil vi komme tilbake til denne problemstillingen, siden vi i waveletanalyse kan velge et helt andre basissett av funksjoner enn sinuser og cosinuser.

Kort oppsummert kan vi si:

Fouriertransformasjon er et meget godt hjelpemiddel, men har mer eller mindre samme basis som Middelalderens episykel-beskrivelse av planetbevegelser. Det er fullt mulig å beskrive planetbevegelser ved hjelp av episykler, men det er lite fruktbart å bruke en slik analyse. På tilsvarende vis er det en rekke fysiske fenomener som i dag beskrives ved formalisme basert på fourieranalyse, der denne formalismen egentlig er lite egnet. Den kan føre til fysiske bilder som villeder mer enn de er til hjelp for oss. Et eksempel finnes i kvanteoptikk.

4.10 Fouriertransformasjon, eksempel på et dataprogram

```
% Enkelt eksempelprogram for å vise hvordan fouriertransformasjon
% kan gjennomføres i praksis i Matlab. Eksemplet er en modifikasjon
% av et eksempelprogram på hjelpesidene i Matlab.

Fs = 1000;           % Samplingsfrekvens
delta_t = 1/Fs;     % Tid mellom hver sampling
N = 1024;           % Antall samplinger
t = (0:N-1)*delta_t; % Tidsvektor

% Lager her et kunstig signal som en sum av et 50 Hz sinussignal
% og en 120 Hz cosinus pluss legger til et random signal:
x = 0.7*sin(2*pi*50*t) + cos(2*pi*120*t);
x = x + 1.2*randn(size(t));

plot(Fs*t,x)        % Plotting av signalet i tidsbilet
title('Opprinnelig signal')
xlabel('tid (millisekunder)')

X = fft(x,N)/N;     % Fouriertransformasjon

frekv = (Fs/2)*linspace(0,1,N/2); % Frekvensvektor (for plot)

% Plotter bare lengden på frekvenskomponentene i frekvensspekteret.
% Velger å bare ta med frekvenser opp til halve samplingsfrekvensen.
figure;             % Hindrer overskriving av forrige figur
plot(frekv,2*abs(X(1:N/2))) % Plotter halvparten av fourierspekteret
title('Absolutt-verdier av frekvensspekteret')
xlabel('Frekvens (Hz)')
ylabel('|X(frekv)|')
```

4.11 Læringsmål

Etter å ha jobbet deg gjennom dette kapitlet bør du vite at:

- En integrerbar tidsavhengig kontinuerlig funksjon kan transformeres ved kontinuerlig fouriertransformasjon til et “frekvensbilde”, som siden kan entydig transformeres med en invers fouriertransformasjon tilbake til utgangspunktet.
- En diskret funksjon kan transformeres ved en diskret fouriertransformasjon til et “frekvensbilde”, som siden kan entydig transformeres med en diskret invers fouriertransformasjon tilbake til utgangspunktet.
- Frekvensbildet i en diskret fouriertransformasjon består av komplekse tall hvor realdelen representerer cosinus-bidrag ved de ulike frekvensene, mens imaginærdelen representerer sinus-bidragene. “Pythagoras” gir amplituden på signalet ved den aktuelle frekvensen. Arkustangens av forholdet mellom imaginær og realdel angir fasen til den aktuelle frekvenskomponenten (relativt til en $\cos(\omega t + \phi)$ -beskrivelse).
- For et reelt signal, er siste halvpart av fourierkoeffisientene kompleks konjugerte av den første halvparten, og “speiling” forekommer. Vi tar derfor oftest bare vare på den første halvparten av frekvensspekteret.
- Dersom et signal “samples” med en samplingsfrekvens f_s , vil vi bare kunne behandle signaler med frekvenser under halve samplingsfrekvensen på en entydig måte.
- For å unngå problemer med “folding”, må derfor et lavpassfilter benyttes for å fjerne signalkomponenter som kan ha høyere frekvens enn halve samplingsfrekvensen. Ved numeriske beregninger må vi selv passe på at “samplingsfrekvensen” er høy nok for signalet vi behandler.
- Fouriertransformasjon er et ypperlig hjelpemiddel ved studier av stasjonære tidsvariable fenomener i store deler av fysikken. Eksempelvis er fouriertransformasjon i utstrakt bruk ved analyse og syntese av lyd.
- Fouriertransformasjon egner seg (nesten) bare for å analysere signaler som har mer eller mindre samme karakter i hele samplingstiden. For transiente signaler som endrer karakter kraftig i løpet av samplingstiden, kan et fourierspekter være mer villedende enn til hjelp.
- Fouriertransformasjonen er mulig å gjennomføre (nesten) alltid, men egner seg dårlig for en del tidsforløp.

4.12 Oppgaver

Forståelses- / diskusjonsspørsmål

1. Hvordan kan vi ved å ta utgangspunkt i et frekvensspekter lage en syntetisk lyd? Vil en slik lyd lyde som et ordentlig instrument?
2. For CD lyd er samplingsfrekvensen 44.1 kHz. Ved lydinnspilling må vi ha et lavpass-filter mellom mikrofonforsterker og samplingskretsene som fjerner alle frekvenser over ca 22 kHz. Hva ville kunne skje med lyden ved avspilling dersom vi ikke tok denne regelen høytidelig?
3. Etter en fouriertransformasjon (FFT) bruker vi ofte bare å plote en del av alle dataene som produseres. Nevn eksempler på hva som kan påvirke vårt valg.
4. Dersom du fourieranalyserer lyd fra en CD-innspilling av et instrument og finner at grunntonen har frekvensen 440 Hz. Hvor finner du da den foldete frekvensen?
5. Finn ut hvor mange samplinger det er hver periode i hver linje i figur 4.3 (angi dette som et desimaltall). Har tallene du kommer fram til relasjon til Nyquist-Shannons samplingsteorem? Påpek hvorfor vi får “speiling” omkring halve samplingsfrekvensen.

Regneoppgaver

6. Vis at første punkt i en digital fouriertransform av et signal er lik halvparten av gjennomsnittsverdien til signalet vi startet ut med.
7. Noen hevder at månefasene virker inn på alt fra vær til humøret til oss mennesker. Sjekk om du kan finne holdepunkter for at temperaturen (maksimums- og/eller minimumstemperaturen hvert døgn) varierer litt med månefasene (i tillegg til all annen variasjon).

Data kan du hente ned fra api.met.no for det stedet du er interessert i og for det tidsrommet du er interessert i. Alternativt kan du bruke en allerede nedhentet og litt forenklet fil *oslotemp.txt* på kurswebsidene. Filen gir temperaturen på Blindern i tiden 1. januar 2003 til og med 31. desember 2012. Fjerde kolumne i filen gir minimumstemperaturene, mens femte kolumne gir max-verdiene.

Forklar nøye hvordan du kan trekke en slutning om hvorvidt månefasen virker inn på temperaturen eller ikke.

Nedenfor er noen linjer Matlab-kode som viser hvordan data kan leses inn fra vår fil til et Matlabprogram (filen har fem kolonner):

```
filnavn = 'tempBlindern10aar.txt';
fileID = fopen(filnavn, 'r');
A = fscanf(fileID, '%d %d %f %f %f', [5,inf]);
minT = A(4,:);
maxT = A(5,:);
```

```

plot(minT, '-r');
hold on;
plot(maxT, '-b');

```

8. Hent opp solflek-data fra web og lag en figur lignende vår figur 4.1. Vær spesielt oppmerksom på å få korrekte verdier langs aksene i fourierspekteret. Er det samsvar mellom høydene i toppene i tidsbildet og amplitudene i frekvensspekteret? Nedenfor er noen linjer Matlab-kode som viser hvordan data kan leses inn i et Matlabprogram (to kolonner):

```

filnavn='soldata.txt';
fileID = fopen(filnavn, 'r');
A = fscanf(fileID, '%f %f', [2,inf]);
plot(A(1,:),A(2,:), '-b');

```

9. Anta at du skal lage et frekvensspekter liknende det i figur 4.16 for en lydskutt du henter fra en CD. Samplingsfrekvensen er 44.1 kHz. Du lagrer $2^{14} = 16384$ datapunkter og foretar en “fast fourier transform” og ender opp med 16384 nye datapunkter som representerer frekvensspekteret. Hvordan skal du i programmet ditt gjøre om fra punkt-nummer til frekvens langs x-aksen når frekvensspekteret skal plottes?
10. Hva blir oppløsningen langs x-aksen i plottet i forgående oppgave? Sagt med andre ord: Hvor stor endring i frekvens får vi ved å gå fra ett punkt i frekvensspekteret til det neste? Ville oppløsningen vært den samme selv om vi bare brukte 1024 punkter som utgangspunkt for fouriertransformasjonen?
11. Skriv et fouriertransformasjons-program i Python eller Matlab (eller hvilket som helst programmeringsspråk) og sjekk at et harmonisk signal med eksakt 13 perioder innenfor 512 punkter gir et frekvensspekter som du forventer. La gjerne signalet være en rent sinussignal eller et kombinert sinus- og cosinussignal.
12. Modifiser programmet såvidt slik at signalet nå får 13.2 perioder innenfor de 512 punktene. Hvordan ser frekvensspekteret ut nå? Beskriv så godt du kan!
13. Modifiser programmet slik at du får 16 hele perioder med *FIRKANT*signal innenfor 1024 punkter. Hvordan ser frekvensspekteret ut nå? Finn på internett et uttrykk for hvordan amplituden for ulike frekvenskomponenter skal være for et firkantsignal, og verifiser at du får omtrent det samme fra dine numeriske beregninger.
14. Modifiser programmet slik at du får 16 hele *sagtenner* (trekantsignal) innenfor de 1024 punktene. Beskriv også dette frekvensspekteret!
15. I et eksempel i kapittel 3 beregnet vi vinkelutslaget til en fysisk pendel ved store utslag. Gjennomfør disse beregningene for 3-4 ulike vinkelutslag og foreta en fourieranalyse av bevegelsen i hvert tilfelle. Kommenter resultatene.
16. Lag et dataprogram hvor du genererer et signal med lengde 1024 punkter. Anta at samplingsfrekvensen er 1 kHz, og beregn det samlede signalet for signalfrekvensen 200 Hz. Fouriertransformer signalet, og kontrollér at frekvensspekteret er slik du

forventer. Gjenta det samme f.eks. for følgende signalfrekvenser: 400 Hz, 750 Hz, 1020 Hz, 1400 Hz, 1800 Hz og endelig 2100 Hz. Forsøk ut fra resultatene å finne ut hvordan speiling/folding arter seg når signalfrekvensen blir til dels betydelig høyere enn halve samplingsfrekvensen.

17. AM-radio (AM: Amplitude-Modulert). Beregn hvordan signalet som sendes fra en AM-sender ser ut, og finn frekvensspekteret av signalet. Det er enklest å gjøre dette for et radiosignal på langbølge-båndet (153 - 279 kHz). La bærebølgen ha frekvensen $f_b = 200$ kHz, og velg at talesignalet er en enkel sinus med frekvens (etter tur) $f_t = 440$ Hz og 4400 Hz. Signalet bør samples med en samplingsfrekvens $f_s = 3.2$ MHz og det kan være passe å bruke $N = 2^{16} = 65536$ punkter. AM-signalet er gitt ved:

$$f(t) = (1 + A \sin(2\pi f_s t)) \cdot \sin(2\pi f_b t)$$

hvor A angir normert amplitude på lydsignalet (for den aller sterkeste lyden som kan sendes uten forvrengning er $A = 1.0$. Bruk en litt mindre verdi, men test gjerne ut hvordan signalet påvirkes av A .)

Plott AM-signalet både i tidsdomenet og frekvensdomenet. Velg ut passe utsnitt i forhold til det fulle datasettet for å få fram det du ønsker å vise. Husk å sette på korrekte tidsangivelser langs x-aksen i tidsdomenet og korrekte frekvensangivelser langs x-aksen i frekvensdomenet.

Hver radiostasjon på mellombølge og langbølge får bare strekke seg ut over et frekvensbånd på 9 kHz totalt. Hvilke følger har dette for kvaliteten på lyden som overføres?

18. FM-radio (FM: Frekvens-Modulert). Beregn hvordan signalet som sendes fra en FM-sender ser ut, og finn frekvensspekteret av signalet. Bruk samme parametre som i forrige oppgave (selv om det ikke i praksis brukes FM på langbølge). FM-signalet kan gis på følgende måte:

```
f(t) = sin(fase(t)); % Prinsipielt, implementeres litt annerledes i Matlab
```

hvor fasen integreres opp i en løkke på følgende måte:

```
fase(1) = 0.0;
for i=1:(N-1)
    fase(i+1)=fase(i) + omega_b*delta_t*(1.0 + A*sin(omega_t*t(i)));
end;
```

hvor “omega_b” og “omega_t” er vinkelfrekvensen for bærebølgen og talesignalet henholdsvis. Tidsstrengen “t(i)” antas å være beregnet på forhånd (avstand mellom punktene er “delta_t”, som bestemmes av samplingsfrekvensen).

A er igjen en normert amplitude for lydsignalet hvor også en såkalt modulasjonsgrad inngår. Du kan velge etter tur f.eks. $A = 0.2$ og 0.7 og se hvordan dette påvirker både tidsbildet og frekvensbildet.

Plott FM-signalet både i tidsdomenet og frekvensdomenet etter samme retningslinjer som i forrige oppgave. (Hint: Det kan være enklest å plote tilfellet hvor talefrekvensen er 4400 Hz og at $A = 0.7$.)

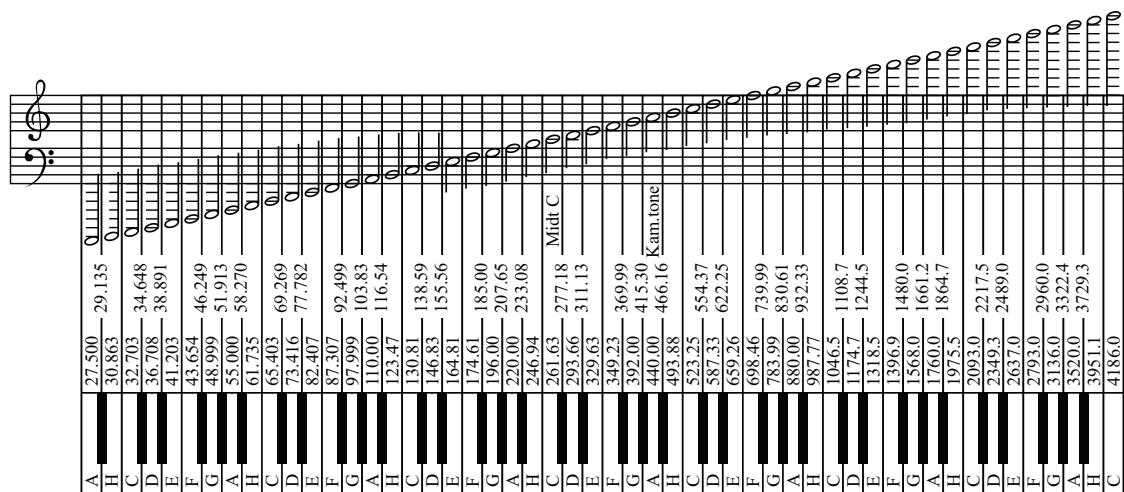
Er det noen klare forskjeller i hvordan frekvensbildet fremstår for FM-signaler sammenlignet med AM-signaler?

19. Bruk invers fouriertransformasjon for å generere en enkel sinus, og spill av lyden på datamaskinen. Helt konkret anbefales følgende: Bruk CD-samplingfrekvensen $Fs = 44100$ Hz og $2^{16} = 65536$ punkter. Verdiene av signalet f bør ligge strengt innenfor intervallet $[-1, +1]$. Bruk den innebygde funksjonen `wavplay(f, Fs)` i Windows eller `sound(f, Fs)` i Linux. Forsøk å lage lyd med frekvensene 100 Hz, 440 Hz, 1000 Hz og 3000 Hz. Du kan gjerne lage et signal som består av flere samtidige sinuser også? Husk å skalere totalsignalet før vi bruker `wavplay` eller `sound`.
20. Les en lydfilen "transient.wav" og foreta fouriertransformasjon for å få fram frekvensspekteret. Lydfilen er tilgjengelig fra kursets websider, samplingfrekvensen er $Fs = 44100$ Hz. Bruk gjerne 2^{17} punkter i analysen. En aktuell Matlabfunksjon er

```
s = 'lydfil1.wav';  
[f,Fs,type] = wavread(s, [nstart nslutt]);  
g = f(:,1); % Henter ut ett monosignal fra stereosignalet f
```

Dersom du lytter til lyden og dernest betrakter frekvensbildet, håper jeg at du reflekterer over det du har gjort. Fourieranalyse blir iblant misbrukt. Hva er problemet med analysen som er foretatt for det aktuelle lydsignalet?

21. Foreta frekvensanalyse for lyd fra en tuba og fra en piccolofløyte (lydfiler tilgjengelig fra kursets websider). Bruk 2^{16} punkter i analysen (se forøvrig forrige oppgave). Bestem tonens plassering i en temperert skala ved å bruke figur 4.19. Frekvensspekteret viser forøvrig overtoner slik det er beskrevet i dette kapitlet (vi kommer tilbake til dette i senere kapitler).
22. "Åpen oppgave" (Det vil si at svært få føringer og tips er gitt): Fouriertransformasjon kan brukes i digital filtrering. Forklar prinsippet og hvordan dette kan gjennomføres i praksis. Lag et lite program som foretar selvvalgt digital filtrering av en virkelig lydfile, hvor det er mulig å lytte til lyden både før etter filtrering. (Vær litt omhyggelig i beskrivelsen av detaljer i det du gjør!)



Figur 4.19: Toneskalaen for en temperert skala slik vi finner den på et piano. Frekvenser for tonene er gitt.

