

Datorövning: Fouriertransform med Python

Skicka in individuellt skrivna rapporter på **engelska inom två veckor** till er övningsledares email (se ovan). Gå innan inlämningen igenom rapporten och granska att samtliga frågor har besvarats. Detta inkluderar förberedelseuppgifterna A-C och datoruppgifterna 1-7. Två bonuspoäng (+2p) till tentamen erhålls om ni också bifogar korrekt lösning till extrauppgiften vid första inlämningen. Extrauppgiften är valfri och krävs inte för att bli godkänd på datorövningen.

Förberedelse:

Läs kapitel 1-2 i "Introduktion till Fourieranalys" från kurshemsidan:

<http://www.teori.atom.fysik.su.se/~marcus/EM/em17.htm>

och lös följande förberedelseuppgifter:

Förberedelseuppgift A:

Beräkna integralen analytiskt:

$$\int_0^{2\pi} \sin(mx) \sin(nx) dx,$$

där m och n är naturliga tal: $0, 1, 2, \dots$. Notera att n och m kan vara lika eller olika!

Förklara kortfattat begreppet att de två sinusfunktionerna är ortogonala.

Förberedelseuppgift B:

Beskriv med ord och ge en ekvation för hur en godtycklig periodisk funktion (med period på 2π) kan approximeras som ett trigonometriskt polynom (av grad N).

Förberedelseuppgift C:

Betrakta en oscillerande funktion $f(x) = \sin(mx)$ på intervallet $0 \leq x < 2\pi$. Beräkna Fourierkoefficienterna a_n och b_n till $f(x)$ genom att utföra Fourierintegralerna:

$$a_n = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos(nx) dx$$

och

$$b_n = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin(nx) dx$$

där m och n naturliga tal.

Vilka frekvenser innehåller den oscillerande signalen $f(x)$? Är det vad du väntade dig?

Datorövning:

Din uppgift under datorlaborationen är att analysera signaler och att utföra Fouriertransformationer med hjälp av en dator. I kapitel 1-2 i "Introduktion till *Fourieranalys*" betraktade vi funktioner som var periodiska över en period på 2π . I denna övning kommer vi att gå vidare till kapitel 3 och anta att signalen är en periodisk tidsfunktion som har samplats digitalt i ett antal ekvidistanta mätpunkter N över en godtycklig periodtid T .

Uppgift 1: Skapa en input-signal

För att studera Fouriertransformer använder vi Python med NumPy-paketet som är avsett för vetenskapliga beräkningar. I denna laboration ska du att använda NumPy (förkortas np) för att programmera vektorer (np.array) och utföra Fouriertransformer (np.fft) för att analysera signaler. Du ska även producera 2D grafer med hjälp av Python-paketet matplotlib (förkortas plt).

a) Börja ditt Pythonskript med att läsa in NumPy och matplotlib-paketet:

```
import numpy as np # Pythons vetenskapliga beräkningsmetoder
import matplotlib.pyplot as plt # Rita grafer med Python
```

b) Definiera sedan ett tidsintervall för *en* sekund, $0 \leq t < T_s$, som samplas i $N = 1024$ punkter enligt $t = n/N$ som en NumPy-array:

```
N=1024 # antal samplingspunkter där signalen är känd
T0=0.0 # start av sampling i sekunder
T1=1.0 # slut av sampling i sekunder
t=np.linspace(T0,T1,N,endpoint=False) # lista samplingstider
```

Notera att samplingstiderna är $T_0, T_0 + dt, T_0 + 2dt, \dots, T_1 - dt$.

c) Konstruera en signal $x(t) = \sin(\omega t + \theta)$, med fasvinkel $\theta = 30^\circ$ och frekvensen $f = 20$ Hz ($\omega = 2\pi f$) genom att skriva följande Python kod:

```
f=20.0 # Hz
w=2*np.pi*f # angulär frekvens
fasvinkel=30.0 # grader
theta=fasvinkel/360.0*2*np.pi # radianer
signal=np.sin(w*t+theta) # signalen: x(t)
```

d) Du kan nu plotta $x(t)$ mot t med hjälp av matplotlib så här:

```
plt.plot(t, signal)
```

e) Kontrollera att du förstår följande begrepp och hur de förhåller sig till varandra: Beräkna periodtid ($T = \dots$) för den oscillerade signalen. Vad är den totala samplingstiden ($T_s = \dots$) och tiden mellan varje sampling (dt)? Beräkna frekvensupplösning (df), samplingsfrekvens (f_s) för signalen (se kapitel 3).

Uppgift 2: Transformera signalen till frekvensdomänen

Din uppgift är nu att beräkna Fouriertransformen $F(f)$ av $x(t)$, genom att utnyttja *Fast Fourier Transform* (FFT) och sedan plotta resultatet.

a) Eftersom vår signal är reell använder vi den reella transformationen `rfft` (istället för den mer generella `fft` som fungerar för komplexa signaler) för att effektivt beräkna den diskreta Fouriertransformen med NumPy-paketet:

```
FT=np.fft.rfft(signal) # Diskret FT av reell signal
plt.plot(FT) # Plotta FT över frekvens-index n
```

Notera att FT är *komplex* och en funktion av frekvens (medan signal är *reell* och en funktion av tiden t). Hur många Fourieramplituder beräknas av `rfft`? *Motivera kortfattat och återkoppla till förberedelseuppgift C!*

b) För att kunna plotta FT mot sin korrekta frekvensskala måste du nu programmera en array med frekvenser `frek`, där frekvensskalan börjar på noll och går upp till den så kallade Nykvistfrekvensen $f_N = f_s/2$ i $N/2+1$ steg. Plotta sedan FT med absolutbelopp mot sin korrekta frekvensaxel `frek` och förklara resultatet med ord:

```
frek=np.linspace(0, fN, N/2+1, endpoint=True)
plt.plot(frek, np.absolute(FT)) # |FT| kallas magnitudspektrum
```

Kontrollera att du förstår vilka index `n`, och motsvarar noll-frekvensen och 20 Hz genom att plotta dessa direkt mot t (i detta specialfall varför är index och frekvens samma sak?):

```
plt.plot(t, np.sin(2*np.pi*frek[n]*t))
```

Formen på `frek` följer av *samplingsteomet* som beskrivs i Fourierkompendiet. Motivera kortfattat varför det inte går att upplösa frekvenser lägre än df eller högre än f_N !

c) Den inversa transformationen beräknas genom `irfft` rutinen. Kontrollera att denna transformation gör så att du får tillbaka den ursprungliga signalen:

```
signal_tillbaka=np.fft.irfft(FT) # Transform tillbaka till t
```

d) Fourieramplituderna påverkas även av fasen θ . Hur ska θ väljas för att FFT ska ge en helt reel eller helt imaginär Fourieramplitud? Ändra fasen `theta` ovan och undersök hur realdelen `np.real(FT)` och imaginärdelen `np.imag(FT)` förändras. Försök att motivera varför det är så!

Uppgift 3: Kvadrera signalen och leta efter övertoner

Kvadrera signalen $x(t)$, och Fouriertransformera resultatet. Vilka frekvenser kan du hitta?

Tips: Med NumPy kan du kvadrera genom att skriva: `signal_kvadrat=signal**2`

Uppgift 4: Analys och rekonstruktion av en fyrkantspuls

Du ska nu undersöka en *fyrkantspuls*, dvs en funktion av typen

$$y(t) = |\sin(\omega t + \theta)| / \sin(\omega t + \theta),$$

med lämpligt val av ω och θ (alla element ska vara är ändliga i alla samplingspunkter).

a) Vilka är de fem lägsta frekvenserna i signalen $y(t)$?

b) *Rekonstruera* Fourierutvecklingen ur svaret i a) med hjälp av den inversa Fouriertransformen. Hur påverkas rekonstruktionen om de höga frekvenserna tas bort? Vid vilka index n förbättras rekonstruktionen märkbart? *Motivera kortfattat och återkoppla till förberedelseuppgift B!*

Uppgift 5: Undersök enhetspulser

Enhetspulser är de kortaste pulser som kan beskrivas med en given sampling τ . Skapa en enhetspuls $P(t)$, alltså en puls som är noll i alla samplings utom i en punkt med index n (där $x_n = 1$). Beräkna Fouriertransformen $F(f)$ och plotta $Re(F)$ och $|F|$ för:

a) $n = 0$

b) $n = 20$

och beskriv med ord hur mycket av olika frekvenser som enhetspulsen innehåller!

Tips: Med NumPy kan du skapa en array med nollor av samma form som samplingsarna:
`np.zeros(shape(\tau))`

Uppgift 6: Undersök signaler med andra frekvenser

Återanvänd signalen $x(t)$ från Uppgift 1, men låt $f = 400, 510, 514, 1000, 2000$ Hz. Plotta sedan *magnitudspektrum* (alltså absolutbeloppet av FT). Vilka frekvenser hittar du i respektive fall? Förklara vad som händer med hjälp av konceptet "*aliasing*".

Extra: Det kan vara roligt att använda ett skript som hittar de starka frekvenserna själv:

```
fi=[i for i, j in enumerate(np.absolute(FT)) if j > N/100.0]
print "Hittade signal vid f = ", frek[fi], " Hz"
```

Uppgift 7: Frekvenser som inte riktigt passar i lådan

I Uppgift 1 - 4 har vi använt heltalsvärden på frekvensen f . Nu ska vi undersöka vad som händer om vi använder icke-heltalsvärden på f (med sampling i $0 \leq t < 1$ s som tidigare).

Plotta magnitudspektrum med $f = 100.4$ Hz i $x(t)$. Vad ser du? Motivera med ord varför du inte längre bara ser att *en* frekvens i ditt magnitudspektrum.

Extrauppgift: Studie av ultra-korta laserpulser (+2p till tentan)

(denna uppgift krävs ej för att bli godkänd på datorövningen)

Laserpulser är en typ av elektromagnetisk strålning som utbreder sig som en våg med ljusets hastighet $c = 3 \times 10^8$ m/s. Laserljusets frekvens $f = \omega/2\pi$ och period $T = 1/f$ ges av ljusets våglängd $\lambda = cT$. Moderna lasertekniker, som till exempel titan-safir-lasern med en våglängd kring 800 nm, kan generera pulser som är nästan lika korta som perioden för det svängande elektriska fältet. Modellera det elektriska fältet i en laserpuls som

$$E(t) = \cos[\omega(t - t_0)] \times \exp\{-[(t - t_0)/\tau]^2\},$$

där $t_0 = x/c$ är tiden då laserpulsen är maximal vid en given punkt x i rummet och där τ är pulsens ungefärliga utsträckning i tiden. I denna modell beskriver *cos*-funktionen hur det elektriska fältet oscillerar medan *exp*-funktionen (en så kallad *Gaussisk funktion*) ger pulsen sin begränsade utsträckning i tiden. Börja med att anta att $\tau = 10T$ och $x = 0$.

a) Skapa en lämplig array \mathbf{t} för att sampla $E(t)$ över hela laserpulsen och så att dess oscillationer syns tydligt när du plottar $E(t)$. *Tänk på att pulsen är mycket kort!*

b) Fouriertransformera laserpulsen. Vid behov förbättra \mathbf{t} för att tydligt upplösa laserpulsens frekvenskomponenter. Vad händer om du gör laser pulsen kortare i tiden genom att minska τ lite? Observera att en fysikalisk laserpuls måste ha en längd som är längre än perioden på lasern, så du måste se till att $\tau \geq T$.

Fouriertransformen av en gaussisk funktion är själv en Gaussisk funktion (men med en annan bredd)! Kan du hitta hur bredderna i de två domänerna förhåller sig till varandra?

c) Förklara med egna ord detta fenomen som kallas för en *osäkerhets princip*: Vad krävs av laserns spektrum för att kunna ha en kort puls? Vad krävs av laserpulsens längd för att ha en väldefinierad frekvens?

Tips: Inom kvantmekaniken är Heisenbergs osäkerhets princip är en viktig princip som säger att man inte kan bestämma både var en partikel är och hur fort den rör på dig. För elektromagnetiska vågor gäller alltså en liknande relation mellan τ och Δf !

d) Hur ser spektrum ut om du gör pulsen mycket kortare än perioden? Återkoppla till Uppgift 6, kan du nu förklara varför *alla* frekvenser behövs för att beskriva *enhetspulsen*?

e) Om laserpulsen innehåller noll-frekvensen (alltså ett *statiskt* elektriskt fält) så kan den *inte* längre utbreda sig som en våg. Verifiera att detta inträffar då $\tau < T$. Fenomenet kallas för *femtosekundsbarriären* och förklarar varför laserpulser inte kan göras kortare än perioden på ljuset (som för optisk ljus är ungefär en femtosekund: 1 fs = 10^{-15} s).