

Projekt 2 Del B: Spektralmetoder

Detta projekt redovisas muntligt för assistent vid schemalagt tillfälle via Zoom, för datum se kursshemsidan. Tider bokas via Canvas. Innan redovisning ska individuella obligatoriska uppgifter utföras i Matlab Grader. I slutet av detta dokument finns mer information om den muntliga redovisningen.

Bakgrund

Numeriska metoder baserade på Fourier serier kan ge väldigt noggranna resultat. I detta projekt ser vi hur vi numeriskt kan differentiera funktioner och lösa partiella differentialekvationer med periodiska randvillkor. Den snabba Fouriertransformen (FFT) används för att accelerera beräkningarna, och vi börjar med att bekanta oss med Matlabs `fft` kommando.

Den diskreta Fouriertransformen (DFT)

Betrakta en 1-periodisk funktion f . Introducera beräkningsnätet $x_j = j/N$, $j = 0, 1, 2, \dots, N-1$, och inför beteckningen $f_j = f(x_j)$. Definiera DFT koefficienterna \hat{f}_k enligt

$$\hat{f}_k = \frac{1}{N} \sum_{j=0}^{N-1} f_j e^{-2\pi i k x_j}, \quad k = -N/2, \dots, N/2 - 1. \quad (1)$$

Matlabs `fft` kommando beräknar DFT koefficienter, vi kommer även att kalla dem Fourierkoefficienter. Notera den annorlunda $1/N$ skalningen ovan jämfört med detta kommando. Matlabs `fft` beräknar koefficienterna ovan för $k = 0, \dots, N-1$. Eftersom koefficienterna repeteras cykliskt, så kan DFT koefficienterna för $k = -N/2, \dots, N/2 - 1$ fås genom att använda `fftshift` på resultatet från `fft`. Vi behöver dessa *centrerade* koefficienter för det vi ska göra i senare uppgifter.

Så här kan koden se ut för att beräkna DFT koefficienterna för den konstanta funktionen $f \equiv 1$ (som förstasenkelt byts ut mot önskad funktion):

```
N=10;
xv=(0:N-1)/N;
fv=ones(size(xv));
fk=1/N*fftshift(fft(fv));
```

Den inversa diskreta Fouriertransformen

När vi beräknat DFT koefficienterna för en funktion så kan vi från dessa beräkna funktionsvärden i det fysiska rummet. Om vi först tar Fouriertransformen för en funktion, och sedan den inversa Fouriertransformen på resultatet, så ska vi förstås få tillbaka exakt det vi startade med i det analytiska fallet.

Den här lilla kodsnutten utgår från värdena av funktionen på ett beräkningsnät, beräknar DFT koefficienterna med `fft`, och sedan värdena på funktionen med `ifft`. Skillnaden mot värdena som vi startade med bör då ligga på en nivå som avrundningsfel. Kontrollera detta.

```
N=100;
xv=(0:N-1)/N;
fv=sin(2*pi*xv).^8.*cos(24*pi*xv).^3;
fk=fftshift(fft(fv));
fv2=ifft(ifftshift(fk));
```

Kodsnutten visar också att om vi gjort `fftshift` på koefficienterna för att få de centrerade koefficienterna så måste vi skifta tillbaka med `ifftshift` innan vi kan applicera `ifft`. Notera att vi här inte har applicerat skalningen $1/N$ på resultatet från Matlabs `fft`. Vi kan göra detta, men ska då också multiplicera resultatet från `ifft` med N , och här har vi valt att inte göra det.

Förberedelseuppgift (behöver inte redovisas): Vad menas med ett *exponentiellt avtagande*? I en log-log plot, plotta både e^{-qN} mot N för något q och sedan N^{-q} mot N . Hur ser kurvorna ut?

Fourier spektrum

Låt följande funktion vara given

$$f(x) = e^{-\alpha(x-\pi/5)^2} \quad (2)$$

på intervallet $[0, 1]$. Denna funktion är inte en 1-periodisk funktion. När vi använder FFT så periodiseras funktionen enligt $f(x + pL) = f(x)$, för alla heltal p där $L = 1$ är den periodiska längden.

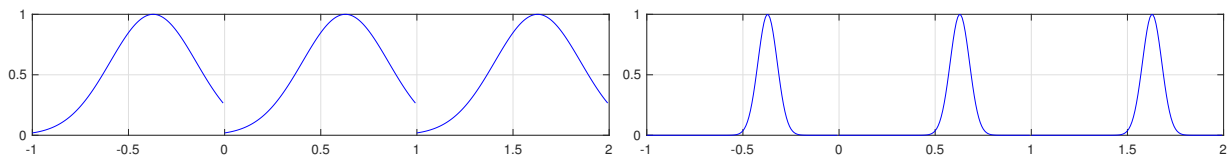


Figure 1: Den periodiserade funktionen från (2) med $\alpha = 10$ till vänster och $\alpha = 200$ till höger.

Vi beräknar nu DFT koefficienterna för $f(x)$ given i (2). För att studera avtagandet av koefficienterna, så plottar vi $|\hat{f}_k|$ mot k för $k = 0, \dots, N/2 - 1$, med $N = 128$, för olika värden på α . För

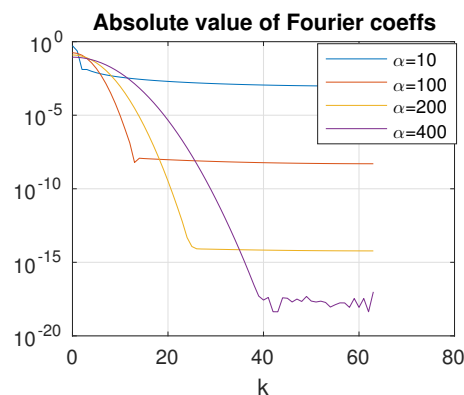


Figure 2: $|\hat{f}_k|$ mot k för $k = 0, \dots, N/2 - 1$, med $N = 2^7$, för olika värden på α .

en C^∞ funktion, så förväntar man sig ett exponentiellt avtagande av koefficienterna. För $\alpha = 400$ ser vi ett sådant avtagande i figur 2, ner till avrundningsnivå. För mindre värden på α så planar kurvorna ut vid större värden. Detta beror på den diskontinuitet som finns i funktionen när vi betraktar den som en periodisk funktion (figur 1) - Fourierkoefficienterna för en styckvis konstant funktion avtar inte. För $\alpha = 10, 100, 200$ är diskontinuiteten i storleksordningen $0.2, 10^{-6}, 10^{-12}$, och förhindrar fortsatt avtagande i Fourierkoefficienterna under en miniminivå. Fourierkoefficienterna för $\alpha = 400$ har ett långsammare avtagande än de andra kurvorna för små värden på k . Det

beror på att "Gaussklockan" (peaken i funktionen) blir smalare och smalare, så att vi behöver fler Fourier moder för att beskriva den väl.

Spektral differentiering

Nu vill vi beräkna derivator av f med hjälp av Fouriertransformen. Vi beräknar då först de centrerade Fourierkoefficienterna och utför differentieringen i Fourierrummet. Sedan beräknar vi den inversa Fouriertransformen för att gå tillbaka till det fysiska rummet, så att vi får värdena på den differentierade funktionen. Beteckna approximationen av $f'(x_j)$, x_j , $j = 0, \dots, N-1$ med Df_j , och approximationen av $f''(x_j)$ med D^2f_j .

Den diskreta 2-normen för en vektor g av längd N definieras av,

$$\|g\|_2 = \sqrt{\frac{1}{N} \sum_{j=0}^{N-1} |g_j|^2}. \quad (3)$$

U1 b i Matlab Grader går upp på att beräkna dessa derivator för ett givet N och i **U1 c**) vill vi se hur felet avtar med N .

U1:

- a) Gör den obligatoriska uppgiften *Diskreta Fourierkoefficienter* i Matlab Grader.
- b) Gör den obligatoriska uppgiften *Spektral differentiering* i Matlab Grader.
- c) Sätt $\alpha = 400$. Plotta felet i approximationen av första derivatan $\|Df - f'\|_2$ mot N (använd tex $N = 2^m$, med $m = 4, 5, 6, 7, 8$). Gör detta i en log-log plot. Beräkna även en approximation till andra derivatan. I samma figur, plotta $\|D^2f - f''\|_2$ vs N . Kommentera på skillnaden mellan de två felkurvorna. Hur ser det ut och vad beror det på?

Spektralmetoder för differentialekvationer

Vi betraktar nu värmeledningsekvationen

$$u_t = du_{xx} + g(x, t), \quad 0 \leq x \leq 1, \quad t \geq 0, \quad u(x, 0) = f(x) \quad (4)$$

med $d \geq 0$ och periodiska randvillkor $u(0, t) = u(1, t)$.

Inför expansionen

$$u^N(x, t) = \sum_{k=-N/2}^{N/2-1} \hat{u}_k(t) e^{2\pi i k x}, \quad (5)$$

samt

$$f^N(x) = \sum_{k=-N/2}^{N/2-1} \hat{f}_k e^{2\pi i k x}, \quad g^N(x, t) = \sum_{k=-N/2}^{N/2-1} \hat{g}_k(t) e^{2\pi i k x}.$$

Definiera också

$$\eta(y, \delta) = \begin{cases} 1 & 0 \leq y \leq \delta \\ \frac{h((y-\delta)/a)}{h((y-\delta)/a) + h((1-(y-\delta)/a))} & \delta < y < 1 - \delta \\ 0 & 1 - \delta \leq y \leq 1, \end{cases} \quad (6)$$

$a = 1 - 2\delta$ och $h(s) = \exp(2\exp(-1/s)/(s-1))$.

En av uppgifterna i Matlab Grader är att skriva en funktion som evaluerar $\eta(y, \delta)$. Den kan användas i **U3**.

U2:

a) Härled de ordinära differentialekvationer som fås för $\hat{u}_k(t)$, $k = -N/2, \dots, N/2 - 1$ i ekvation (5).

b) Visa hur du för $g = 0$ kan finna exakta lösningar till differentialekvationerna för Fourierkoefficienterna i **U2 a**). Skriv upp formlerna för $\hat{u}_k(t)$, $k = -N/2, \dots, N/2 - 1$.

c) Låt $d = 1$ och $g = 0$. Sätt initialdata till $f(x) = \sin(2\pi x)$, så att den exakta lösningen till (4) är känd. Beräkna lösningen för $t = 0.1$ och för $N = 16$ genom att använda `fft` och `ifft` och där emellan den analytiska lösningen för Fourierkoefficienterna. Plotta den exakta lösningen och felet i fft-lösningen i intervallet $[0, 1]$. (Om det blivit rätt ska felet vara på avrundningsnivå).

U3:

a) Implementera funktionen $\eta(y, \delta)$ i ekvation (6) som en Matlab-funktion. Uppgiften testas som en obligatorisk uppgift i Matlab Grader.

b) Vi studerar samma problem som i **U2 c**), dvs ekvation (4) med periodiska randvillkor och $d = 1$, $g = 0$. Lös problemet på samma sätt som i **U2 c**), men sätt nu initialdata till $f(x) = \eta(2|x - 1/2|, 0.05)$ med $\eta(y, \delta)$ definierad i (6) och $\delta = 0.05$ (se figur 3).

Beräkna en referenslösning (en noggrann numerisk lösning) till problemet vid $t = 0.1$ med $N = 2^{11} = 2048$ (alternativt $N = 2^{10} = 1024$). Beräkna sedan numeriska lösningar för $N = 2^m$ med $m = 4, 5, 6, 7, 8, 9$. Använd den diskreta 2-normen tidigare definierad i (3) och mät de diskreta 2-felen jämfört med referenslösningen. Plotta 2-felen mot N i en log-log plot (dvs $\log(N)$ på x -axeln och $\log(\text{fel})$ på y -axeln). Hur avtar felet?

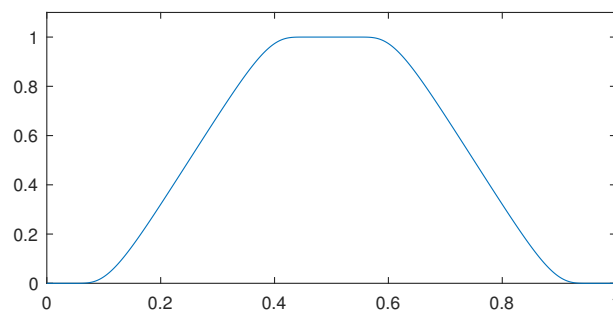


Figure 3: $\eta(2|x - 1/2|, \delta)$ (med $\delta = 0.05$) plottad mot x på intervallet $[0, 1]$. Detta är en glatt funktion (C^∞) - den har oändligt många kontinuerliga derivator också när den periodiseras.

Inför redovisningen

För den muntliga redovisningen så gäller att alla Matlab-filer som använts ska finnas tillgängliga på datorn som ni redovisar vid, och namngivna så att ni snabbt kan hitta dem och köra dem. De

ska även laddas upp via Canvas innan redovisningen (under Uppgifter). För de resultat och plottar som efterfrågas så ska de finnas tillgängliga och färdiga enligt lista nedan, och kunna visas upp genom att dela skärm. I övrigt förväntas du kunna diskutera och kommentera dina resultat och svara på de frågor som ställs i uppgifterna.

U1 c): Den plot som efterfrågas med felen för första- och andraderivator.

U2 a)-b): De härledningar och resultat som efterfrågas, c): De plottar som efterfrågas: exakt lösning och fel över intervallet $[0 \ 1]$.

U3 b) Log-log plot som visar hur felet avtar med N .