1 Introduktion

Denne opgave er en af de to opgaver, som eksamen i kurset bliver baseret på. Der er tale om en gruppe-eksamen, så opgaven har et omfang svarende dertil. Opgaven er delt i to dele. Delopgaverne omfatter både teorien og de numeriske eksperimenter.
Bemærk: Ved eksamen stilles der spørgmål i både teori og numeriske eksperimenter.

2 Delopgave 1: Det hængende kabel

Et kabel ophængt i to pæle af samme højde vil under påvirkning af tyngdekraften antage en form, der kan beskrives ved en kurve, som på dansk kaldes en kædelinie. På engelsk er navnet 'catenary'. Det er grafen for en funktion af følgende form

$$f(x) = h_0 + \lambda \cosh(\frac{x}{\lambda}). \tag{1}$$

Kablets massetæthed betegnes med ρ og spændingen i kablet i dets laveste punkt med T. Så er $\lambda = T/\rho$. Koordinaterne er valgt, så at kablets laveste punkt er $x = 0$. Vi minder om, at

$$\cosh(x) = \tfrac{1}{2}(e^x + e^{-x}). \tag{2}$$

Se [2] for udledelse af formlen og meget mere om kædelinien.

Antag at kablet er ophængt mellem to lige høje pæle hver med højde h og med afstand L. På grund af symmetrien kan vi antage, at pælene er placeret i punkterne $x = -L/2$ og $x = L/2$. Vi lader s betegne den afstand, som kablet synker i midten.

1. Tegn grafen for funktionen $f(x)$ fra (1) for forskellige valg af parametrene h_0 og λ. Forsøg at forklare hvilken effekt parametrene har på kurvens form.

2. Vis, at følgende gælder:

$$f(L/2) = h_0 + \lambda \cosh(\frac{L}{2\lambda}) = h, \tag{3}$$

$$f(0) = h_0 + \lambda = h - s. \tag{4}$$

   Vis, at disse formler giver ligningen

$$\lambda \cosh(\frac{L}{2\lambda}) = \lambda + s. \tag{5}$$

3. Gennemgå bisektionsmetoden, Newtons metode og sekantmetoden til bestemmelse af nulpunkt af en funktion. Gennemgå iterationsmetoden til bestemmelse af fixpunkt for en funktion.

4. Vi ser nu på et konkret eksempel, hvor vi har $L = 150$ (målt i meter) og $s = 15$. Indsætter vi disse parametre i (5), så får vi følgende ligning til bestemmelse af λ.

$$\lambda \cosh(\frac{75}{\lambda}) = \lambda + 15. \tag{6}$$

   (a) Bestem ud fra (6) en numerisk approksimation til værdien af λ, både ved at bruge bisektionsmetoden, og ved at bruge Newtons metode. Python scripts har I allerede lavet til disse udregninger. Argumentér ud fra teorien i Turner [1] for anvendelsen af disse to metoder.

(b) Skriv et Python script der implementerer sekantmetoden. Det kan gøres ved at modificere scriptet beregnet til Newtons metode. Sammenlig sekantmetoden med Newtons metode.

(c) Ligningen (6) kan omskrives til en fixpunktligning. Gør rede for de to omskrivninger her

$$\lambda = \frac{\lambda + 15}{\cosh(\frac{75}{\lambda})} \tag{7}$$

$$\lambda = \lambda \cosh(\frac{75}{\lambda}) - 15 \tag{8}$$

Undersøg anvendeligheden af disse ligninger til bestemmelse af λ på to forskellige måder.

(1) Lav computereksperimenter. Skriv de nødvendige Python scripts.

(2) Brug [1, Theorem 13] til teoretiske overvejelser.

(d) Sammenlig de forskellige metoder til bestemmelse af λ ud fra ligning (6). Sammenligningen skal omfatte både de konkrete tal og konvergenshastighed af metoderne.

3 Delopgave 2: Interpolation af funktioner

I denne delopgave skal vi se på interpolation af funktioner af én variabel. Vi har givet datapunkter $x_0, x_1, \ldots, x_N$ og $f_0, f_1, \ldots, f_N$. Som hjælp til de numeriske beregninger er der to Python scripts til rådighed: `Interpolation_bin.py` og `Interpolation_dec.py`. Det første bruger den indbyggede binære artimetik og det andet bruger `decimal` modulet.

1. Gennemgå teorien for Lagrange interpolation baseret på datapunkterne $x_0, x_1, \ldots, x_N$ og $f_0, f_1, \ldots, f_N$.

2. Vi definerer funktionen f ved

$$f(x) = x^2 - \sin(10x), \quad x \in \mathbf{R}. \tag{9}$$

Vi ser på intervallet $[0, 3]$ og definerer punkterne $x_j = 3j/N$, $j = 0, 1, \ldots, N$, så at vi har $N + 1$ ækvidistante punkter. Vi definerer også $f_j = f(x_j)$, $j = 0, 1, \ldots, N$. Vi betegner med p_N det interpolerende polynomium baseret på disse data.

3. Scriptet `Interpolation_bin.py` giver en vurdering af den maksimale fejl

$$\max_{x \in [0,3]} |f(x) - p_N(x)|$$

for forskellige værdier af N. Det beregner også en teoretisk vurdering af fejlen, se nedenfor. Gennemgå dette script i detaljer. **Bemærk** at dette script ikke bruger `numpy` men er baseret på lister. Begrundelsen er at vi skal være i stand til at modificere dette script til at anvende `decimal` modulet. Efter gennemgangen skal I modificere scriptet til at generere plots for små værdier af N, der i samme figur viser grafen for $f(x)$, grafen for $p(x)$ og knudepunkterne.

4. Lav en grafisk undersøgelse af fejlen $|f(x) - p_N(x)|$, $x \in [0, 3]$, for en række forskellige værdier af N baseret på en modificeret udgave af `Interpolation_bin.py`.

5. Brug Theorem 16 i afsnit 6.2 i [1] til at lave en teoretisk vurdering af fejlen

$$\max_{x\in[0,3]} |f(x) - p_N(x)|. \tag{10}$$

Hjælp: Lad $L(x) = (x - x_0)(x - x_1)\cdots(x - x_N)$. For ækvidistante punkter med $x_{j+1} - x_j = h$ kan man vise, at der gælder

$$|L(x)| \leq \frac{1}{4}(N+1)!h^{N+1}, \quad x \in [x_0, x_N]. \tag{11}$$

6. Hvor stor skal N være for at fejlen er mindre end 10^{-10}? Brug den oprindelige version af `Interpolation_bin.py` til at sammenligne teori og numerisk eksperiment. Hvad er konklusionen? Hvor kommer problemerne fra? I den forbindelse kan I inddrage den grafiske undersøgelse fra punkt 4.

7. Gentag undersøgelserne af den maksimale fejl og sammenligning med den teoretiske vurdering ved at bruge decimal aritmetik scriptet `Interpolation_dec.py`. Prøv med forskellige værdier for antallet af betydende cifre. Hvad er konklusionen?

8. Vi erstatter nu de ækvidistante punkter x_j, $j = 0, 1, \ldots N$ med *Chebyshevpunkterne* $\tilde{x}_j = 3(1 - \cos(j\pi/N))/2$, $j = 0, 1, \ldots N$. Gennemfør den grafiske undersøgelse fra punkt 4. ovenfor med dette valg af knudepunkter. Hvad kan man konkludere ud fra denne undersøgelse?

9. (ekstra spørgsmål, ikke obligatorisk) Bevis vurderingen i (11). Bemærk at vurderingen ikke er den bedst mulige, så måske kan I bevise én der er bedre.

Litteratur

[1] P. Turner, T. Arildsen, and K. Kavanagh, *Applied Scientific Computing*, Springer 2018.

[2] `http://en.wikipedia.org/wiki/Catenary`