Programmering og Modellering (PoM)
Ugeseddel 12 — Uge 49 — Deadline 5/12

Kim Steenstrup Pedersen, Katrine Hommelhoff Jensen, Knud Henriksen,
Mossa Merhi og Hans Jacob T. Stephensen

29. november 2014

1 Plan for ugen

I denne uge tager vi fat på et nyt emne: Modellering med differentialligninger. Mange naturvidenskabelige problemer kan modelleres ved hjælp af differentialligninger og vi skal se på nogle eksempler. Visse differentialligninger kan løses analytisk, dvs. vi kan opskrive en formel for løsningen. Men for flertallet af differentialligninger kan vi ikke finde en analytisk løsning, hvorfor vi må ty til numeriske metoder. For differentialligninger anvender man differensmetoder til at finde numeriske løsninger.

Vi vil også gennemgå Python biblioteket `numpy` som giver os en ny effektiv datastruktur kaldet `ndarray` som tillader os at repræsentere multi-dimensionelle tabeller. Dertil giver `numpy` os en række effektive metoder til at lave numeriske beregninger på data repræsenteret i tabelform, som eksempelvis matricer og billeder. Biblioteket `matplotlib` understøtter også `ndarray` og kan derfor anvendes sammen med `numpy`.

Nøgleord for denne uge vil være: Modellering med differentialligninger, `numpy`'s ndarray (herunder creation, indexing, slicing, I/O, broadcasting, og Matplotlib og `numpy`).

Til tirsdag:

Læs noter om modellering med differentialligninger.

Læs NumPy User Guide Kapitler 2.1-2.7 (`http://docs.scipy.org/doc/numpy/user/`).

Til forelæsningen behandles `numpy`'s ndarray: creation, indexing, slicing, broadcasting, I/O.

Til torsdag:

Orienter jer i NumPy Reference Guide (`http://docs.scipy.org/doc/numpy/reference/`), og læs afsnit om `numpy.linalg` og `numpy.random`. Læs desuden Matplotlib User's Guide (`http://matplotlib.sourceforge.net/users/index.html`).

Til forelæsningen behandles matrixberegninger med `numpy` samt generering af grafer især med fokus på matplotlib.pyplot.

1.1 Gruppeopgave

Den 5/12 senest klokken 15:00 skal besvarelse af følgende opgave afleveres elektronisk via Absalon. Opgaven skal besvares i grupper bestående af 1 til 3 personer og skal godkendes, for at gruppedeltagerne kan kvalificere sig til den afsluttende tag-hjem eksamen. Opgavebesvarelsen skal oploades via kursushjemmesiden på Absalon (find underpunktet `ugeseddel12` under punktet `Ugesedler og opgaver`). Kildekodefiler ("script"-filer) skal afleveres som "ren tekst", sådan som den dannes af `emacs`, `gedit`, `Notepad`, `TextEdit` eller hvilket redigeringsprogram man nu bruger (*ikke* PDF eller HTML eller RTF eller MS Word-format). Filen skal navngives *efternavn1.efternavn2.efternavn3*.49.py, mens andre filer skal afleveres som en PDF-fil med navnet *efternavn1.efternavn2.efternavn3*.49.pdf.

12g1 **Bølgeligningen:** Vi befinder os på et stadion. Stemningen er god, og en fan beslutter sig for at starte en bølge og resten af stadion følger trop. Men hvordan kan vi undersøge og beskrive dette fænomen matematisk?

Differentialligninger på formen

$$c^2 \frac{\partial^2}{\partial x^2} u(x,t) = \frac{\partial^2}{\partial t^2} u(x,t), \quad x \in (0,\ell), \quad t > 0 \ , \tag{1}$$

hvor $c > 0$ er en strengt positiv konstant, beskriver udviklingen af bølger. Hvor ℓ angiver længden af bølgen.

Vi vil nu undersøge, hvordan bølgen, som vores ærkefan har startet, udvikler sig. I vores pragmatiske tilgang, klipper vi tilskuermængden i stadion op og lægger tilskuerne ned på et linjestykke med længden ℓ (se figur 1). Desuden undersøger vi kun bølgens udvikling rundt om stadion og er kun interesseret i at modellere hvor bølgen rejser sig på stadion når tiden går. Vi placerer vores ærkefan i starten af linjestykket (se vores initialbetingelse nedenfor). Dermed beskriver løsningen $u(x,t)$ til den endimensionelle differentialligning (1), tilskuerbølgen til tiden t. Da vi ønsker at bevare den runde struktur af vores stadium, antager vi periodiske randbetingelser, dvs. $u(0,t) = u(\ell,t)$, som vi sætter lig 0 for enkelhedens skyld, dvs. $u(0,t) = u(\ell,t) = 0$. Det betyder at når bølgen når til enden af linjestykket så er den kommet hele vejen rundt om stadion. Her begynder vores model at bryde sammen idet modelbølgen vil vende ved endepunktet af linjestykket og begynde at bevæge sig baglæns tilbage til vores ærkefan. Men pyt med det.

Til diskretisering af differentialligningen, bruger vi centrale approksimationer for den 2. afledte, dvs.

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u(x_{i+1},t_j) - 2u(x_i,t_j) + u(x_{i-1},t_j)}{h^2} \tag{2}$$

$$\frac{\partial^2 u}{\partial t^2} \approx \frac{u(x_i,t_{j+1}) - 2u(x_i,t_j) + u(x_i,t_{j-1})}{k^2} \ , \tag{3}$$

hvor h angiver skridtstørrelsen mellem x_{i-1} og x_i, dvs. $h = x_i - x_{i-1}$ og tilsvarende angiver k skridtstørrelsen mellem t_{j-1} og t_j, dvs. $k = t_j - t_{j-1}$.

Vores initialbetingelse er

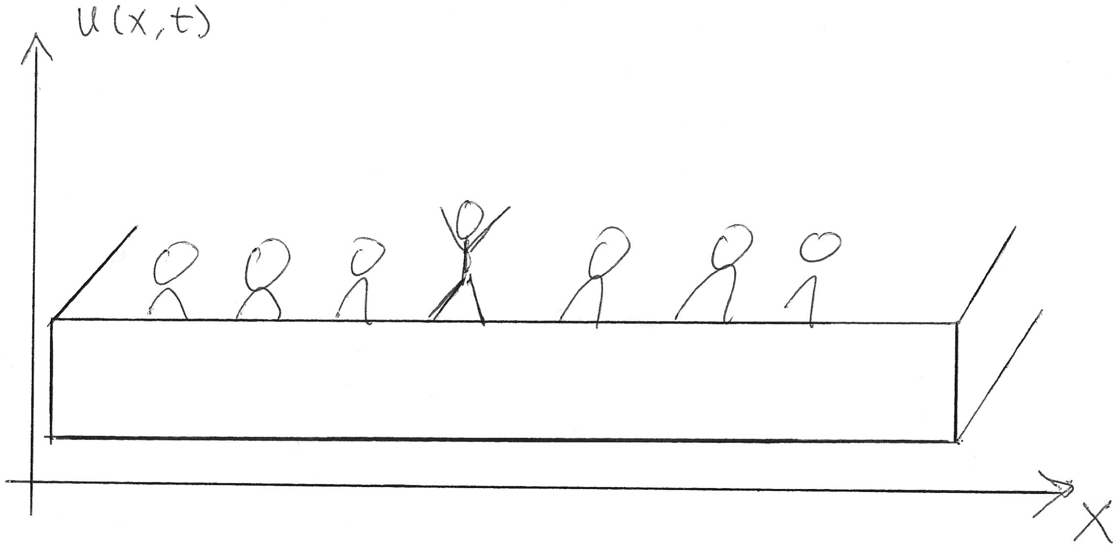
$$\begin{aligned} u(x,t=0) &= \exp\left(-\frac{(x-8)^2}{4}\right) \\ u(x,t=k) &= \exp\left(-\frac{(x-8-ck)^2}{4}\right) \ . \end{aligned}$$

Bemærk, dette er nødvendigt da vi skal bruge to tidsskridt, t_{j-1} og t_j, for at kunne beregne funktionen u til tiden t_{j+1}, så vi har brug for at kende en initial funktionsværdi for u til tiden $t = 0$ og $t = k$.

(a) Indsæt approksimationerne (2) og (3) i differentialligningen (1) og opskriv derved differensligningen med ovenstående approximationer (2), (3), gruppér samtlige led og isolér $u(x_i, t_{j+1})$. Hermed får vi en formel til beregning af funktionen u som funktion af x til næste tidspunkt t_{j+1}.

(b) Implementér et program, der beregner en approksimativ løsning $u(x,t)$ til differentialligningen (1) ved at anvende differensligningen fra (a), samt kan producere nogle grafer af løsningen $u(x,t)$ til udvalgte tidspunkter t (dvs. fasthold tidspunktet t og tegn $u(x,t)$ som funktion af x).

(c) Brug $c = 1$, $\ell = 250$, hvor lang tid er bølgen om at komme hele vejen rundt om stadion, dvs. til hvilket tidspunkt når bølgen frem til enden af vores linjestykke? Undersøg dette ved hjælp af dit program og besvar spørgsmålet i rapporten som også skal indeholde nogle grafer der illustrere bøglens udvikling ved bestemte tidspunkter herunder for $t = 0$ og $t = k$ samt når bølgen er halvvejs og når bølgen når enden af linjestykket. Hvad sker der når der ændres på skridtstørrelserne h og k?



Figur 1: Publikum på stadion laver en bølge som bevæger sig langs x-aksen og funktionen $u(x,t)$ modellere tilskuerbølgen som en endimensionel bølge.

1.2 Tirsdagsøvelser

Besvarelser af disse opgaver skal ikke afleveres, men opgaverne forventes løst inden torsdag.

12ti1 Løsning af differentialligninger er en fundamental opgave for mange videnskabsgrene. I skal skrive et program, som simulerer et kast med en bold. I det følgende vil vi benytte, et 3 dimensionalt koordinatsystem $[x,y,z]^T$, hvor y-aksen er den lodrette.

Newtons 2. lov definerer kraft som:

$$\vec{F}(t) = M\frac{d^2\vec{P}(t)}{dt^2}, \tag{4}$$

hvor $\vec{F}$ er den kraft et punktlegeme påvirkes med til tiden t, M er legemets masse, og $\vec{P}(t)$ er legemets position, således at $\vec{A}(t) = d^2\vec{P}(t)/dt^2$ er legemets acceleration og $\vec{V}(t) = d\vec{P}(t)/dt$ dets hastighed. Tæt på jordens overflade bliver alle legemer påvirket af en konstant tyngdeacceleration på ca. $-9.82\,\mathrm{m/s^2}$, hvorfor et punktlegeme på $1\,\mathrm{kg}$ vil blive påvirket af en konstant kraft på $\vec{F}_{\text{tyngde}} = [0,-9.82,0]^T\,\mathrm{m\,kg/s^2}$. Hvis vi betragter et legeme, der vejer $1\,\mathrm{kg}$, og som sendes af sted fra punktet $P(0) = [0,0,0]^T\,\mathrm{m}$ med en initial hastighedsvektor på $\vec{V}(0) = [100,100,0]^T\,\mathrm{m/s}$, så vil vi kunne udregne dets bane ved integration:

$$\vec{V}(\tau) = \int_0^\tau \vec{A}(t)dt \tag{5a}$$

$$= \int_0^\tau \frac{\vec{F}_{\text{tyngde}}}{M}dt \tag{5b}$$

$$= \left[\frac{\vec{F}_{\text{tyngde}}}{M}t\,\mathrm{m/s^2}\right]_0^\tau + \text{konst.} \tag{5c}$$

$$= \frac{\vec{F}_{\text{tyngde}}}{M}\tau\,\mathrm{m/s^2} + \text{konst.}. \tag{5d}$$

Man kan indse, at 3. koordinaten altid vil være 0, og derfor vil vi ignorere den i det følgende, således at $\vec{V}(t) = [0,-9.82]^T t\,\mathrm{m/s^2} + [100,100]^T\,\mathrm{m/s}$. Integration endnu en gang og indsættelse af begyndelsespunktet giver,

$$\vec{P}(\tau) = \frac{1}{2}[0,-9.82]^T\tau^2\,\mathrm{m/s^2} + [100,100]^T\tau\,\mathrm{m/s} \tag{6}$$

I denne opgave skal du skrive et program, der producerer den eksakte graf vha. (6).

1.3 Torsdagsøvelser

Besvarelser af disse opgaver skal ikke afleveres, men opgaverne forventes løst inden tirsdag i efterfølgende uge.

12to1 I denne og de efterfølgende opgaver antager vi at du allerede har importeret `numpy` biblioteket med `import numpy as np`. Opret et `numpy.ndarray` med `A = np.arange(10)`. Omform `A` til en 2×5 matrix uden at kopiere over i en ny variabel. Omform derpå A til en 5×2 matrix.

12to2 Opret et `numpy.ndarray` `A` med 4 elementer med element datatypen `int`. Tildel alle elementerne værdien 3.5 og se derpå hvad der er gemt i elementerne i `A`.

12to3 Opret et `numpy.ndarray` med `A = np.arange(10)`. Benyt indeksering til at udtrække hvert andet element fra `A` og gem i en variabel kaldet `B`. Prøv at tildele elementerne i `B` værdien -1. Hvad sker der med elementerne i `A` og kan du forklare det?

12to4 Hvad er forskellen på resultaterne af følgende beregninger

```
A = np.arange(4)
B = np.asmatrix(A)
C = A.T * A
D = B.T * B
```

Kan du forklare forklare hvorfor der er forskel på C og D?

12to5 Vi kan i `numpy` oprette identitetsmatricen ved hjælp af funktionen `numpy.eye`. Skriv en funktion som anvender `numpy.ndarray` til lave en matrix med værdien 1 på anti-diagonal elementerne. Et 3×3 eksempel på sådan en anti-diagonal matrix se såldes ud

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} .$$

Kan du skrive funktionen uden brug af løkker og ved at anvende `numpy.ndarray` indekseringsmekanisme?

12to6 Betragt det lineære ligningssystem,

$$2x_1 + x_2 + 5x_3 + x_4 = 9, \tag{7a}$$
$$x_1 + x_2 - 3x_3 - x_4 = -5, \tag{7b}$$
$$3x_1 + 6x_2 - 2x_3 + x_4 = 8, \tag{7c}$$
$$2x_1 + 2x_2 + 2x_3 - 3x_4 = 3. \tag{7d}$$

Er ligningssystemet tæt på at være singulært? Benyt `numpy.inv` til at løse for variablene x_i - start med at læse op på hvordan man bruger funktionen `numpy.inv`.

12to7 Hvis $A \in \mathbb{R}^{n \times m}$ så kaldes følgende udtryk for den pseudoinverse til A

$$A^+ = (A^T A)^{-1} A^T$$

Benyt `numpy.mat` datatypen og opret en tilfældig matrix `A` ved hjælp af `numpy.random.random`. Beregn den pseudoinverse til `A` ved at anvende ovenstående udtryk.

12to8 Definer en tilfældig og symmetrisk `ndarray`, x, med `numpy.random.random`, og udregn dens egenværdier og -vektorer, e og v, med `numpy.linalg.eig`. Verificer, at v er en orthogonal matrix, og at $v * \mathrm{diag}(e) * v^T$ er tæt på x ved brug af `numpy.dot`-metoden.

12to9 Udregn 5'te potens af x fra opgave 12to8 med `**`-operatoren, og sammenlign med $v * \mathrm{diag(e)}^5 * v^T$. Giver det samme resultat?

12to10 Desværre har langt de fleste differentialligninger ikke en „closed-form“ løsning som den beskrevet i opgave 12ti1. Istedet kan f.eks. benyttes differensmetoder. Til løsning af samme ligningssystem med differensmetoder kan man anvende følgende approksimation,

$$\vec{V}(t + \Delta t) \simeq \vec{V}(t) + \vec{A}(t)\Delta t, \tag{8a}$$

$$\vec{P}(t + \Delta t) \simeq \vec{P}(t) + \vec{V}(t)\Delta t, \tag{8b}$$

således at hvis man begynder i et kendt punkt f.eks. $V(0)$ og $P(0)$, så kan man udregne en approksimation til kurven $P(t)$ ved at iterere ovenstående ligninger. Denne form er nyttig, hvis man f.eks. tager vindmodstand i betragtning. Vindmodstand kan modelleres som en kraft, der er parallel med $-\vec{V}(t)$, og hvis længde er proportional med $||\vec{V}(t)||^2$, $\vec{F}_{\text{luft}}(t) = -C||\vec{V}(t)||\vec{V}(t)$, hvor konstanten f.eks. kunne være $C = 1/3600$ med passende enheder. For en model af et kast med vindmodstand vil den resulterende kraft være $\vec{F}(t) = \vec{F}_{\text{tyngde}} + \vec{F}_{\text{luft}}(t)$, og $\vec{A}(t) = \vec{F}(t)/M$.

I denne opgave skal du skrive et program, der producerer den eksakte graf vha. (6) (genbrug din løsning til opgave 12ti1), samt grafer, der sammenligner approksimationer vha. (8) for de 3 forskellige værdier af $\Delta t = 4, 1, 1/4$ og med og uden vindmodstand, hvor $C = 1/3600$. Du kan med fordel benytte `numpy.ndarray`.

12to11 Denne opgave går ud på at simulere trafik: Antag, at der er N objekter (biler, cykler, eller gående) som bevæger sig på en cirkelbane med radius 1. Parametriser positionen for objekt i ved dets vinkel, θ_i, således at dets position i 2D er

$$\vec{x}_i = \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix},$$

og således at dets vinkel-hastighed og -acceleration er givet ved $0 \leq v_i \leq V$ og $|a_i| < A$, hvor V og A er maksimumgrænserne for hastighed og acceleration.

1. Skriv et program, som simulerer trafikken ved brug af følgende opdateringsregel:

$$\begin{aligned} \theta_i(t + \Delta t) &= \theta_i(t) + v_i(t)\Delta t, \\ v_i(t + \Delta t) &= v_i(t) + a_i(t)\Delta t, \\ a_i(t + \Delta t) &= g(\theta_i(t), \theta_{i+1}(t)), \end{aligned}$$

   tilrettet således at 1) objekterne tilstræber afstanden $\frac{2\pi}{N}$ til objektet foran; 2) objekterne bremser helt op, hvis de i næste tidsskridt ville kollidere med objektet foran.

2. Afprøv programmet for $\Delta t = .1$, $N = 10$ og $N = 100$ og fornuftige valg af V, A, og g, og argumenter kort for dine valg (max. 5 linjer).

3. Når trafikken er tæt opstår der typisk stående bølger som bevæger sig modsat retning af trafikken. Kan du fremprovokere dette fænomen i dit program ved passende valg af N, V, A, og g?