Information 1TD403, 231026

För betyg 3 krävs: Att man klarar varje delmål för betyg 3 nedan. För att visa att du har nått delmålet behöver du klara minst en av de två uppgifterna för respektive delmål. Du kan också komplettera ett missat delmål genom att klara motsvarande mål på del B.
För betyg 4 krävs: Att man klarar både betyg 3 och minst en uppgift för höre betyg 4.

För betyg 5 krävs: Att man klarar betyg 3, samt båda uppgifterna för högre betyg (smärre felaktigheter är tillåtna).

Hjälpmedel: Formelbladet för kursen och Scipy/Python online. Dessa finns under "Resurser" i Inspera. Du kan använda Python som miniräknare och givetvis också för att skriva program. Dessutom är vanlig miniräknare tillåten (men inte nödvändig).
Det finns länkar till två olika online Python resurser, men den ena får fungera som reserv eftersom SciPy inte går att använda där (däremot kan det fungera som "miniräknare".

Lösningarna skriver du in direkt här i Inspera men du kan även lämna in kompletterande svar på papper (gäller även uppgifterna i del A). Om du lämnar in på papper, ange detta i textfältet.

Delmål 1, 1TD403

Delmål 1: *kunna skriva ett program (i Python eller Matlab) som gör en numerisk simulering av något fenomen, givet en matematisk modell av fenomenet*

För att visa att du har nått delmålet behöver du klara minst en av de följande två uppgifterna.

1 ODE implementation

Skriv ett Python-program som med hjälp av **solve_ivp** löser följande ODE-problem

$$\begin{cases} y'(t) = sin(y \cdot t) \\ y(0) = 1, \end{cases}$$

för $t \in [0, 5]$ och plottar lösningen.

Du kan antingen lösa problemet i SciPy online och klistra in koden i rutan nedan, eller skriva koden mer i "algoritmform". Mindre syntaxfel ger inga avdrag.

Skriv in ditt svar här

Totalpoäng: 2

2 ODE, fjäder

En vibration med dämpning, t ex en fjäder, kan beskrivas med en andra ordningens ODE:

$$\begin{cases} m \cdot y''(t) + c \cdot y'(t) + k \cdot y(t) = 0 \\ y(0) = 0 \\ y'(0) = 1 \end{cases}$$

där *m* är massan, *y(t)* är position vid tid *t, c* en dämpningskonstant, *k* är en fjäderkonstant. Skriv ett Pythonprogram som beräknar y(t) på tidsintervallet $t \in [0, 2]$ med hjälp av funktionen **solve_ivp**. Använd *m* = 2, *k* = 100, *c* = 0.1. Du behöver inte plotta lösningen.

Du kan antingen lösa problemet i SciPy online och klistra in koden i rutan nedan, eller skriva koden mer i "algoritmform". Mindre syntaxfel ger inga avdrag.

Skriv in ditt svar här

```
1
```

Totalpoäng: 2

i Delmål 2, 1TD403

Delmål 2: *känna till viktiga begrepp i anslutning till numerisk simulering*
För att visa att du har nått delmålet behöver du klara minst en av de följande två uppgifterna.

3 Begrepp Markovprocess

Markovprocesser är en av de viktigaste stokastiska processerna, och används mycket i samband med Monte Carlometoder.
Minst tre korrekta svar krävs för godkänt på den här uppgiften.

(a) Vad utmärker en Markovprocess (definitionen av en Markovprocess)?

Skriv in ditt svar här

(b) Ge ett exempel på en diskret Markovprocess

Skriv in ditt svar här

(c) Ge ett exempel på en kontinuerlig Markovprocess.

Skriv in ditt svar här

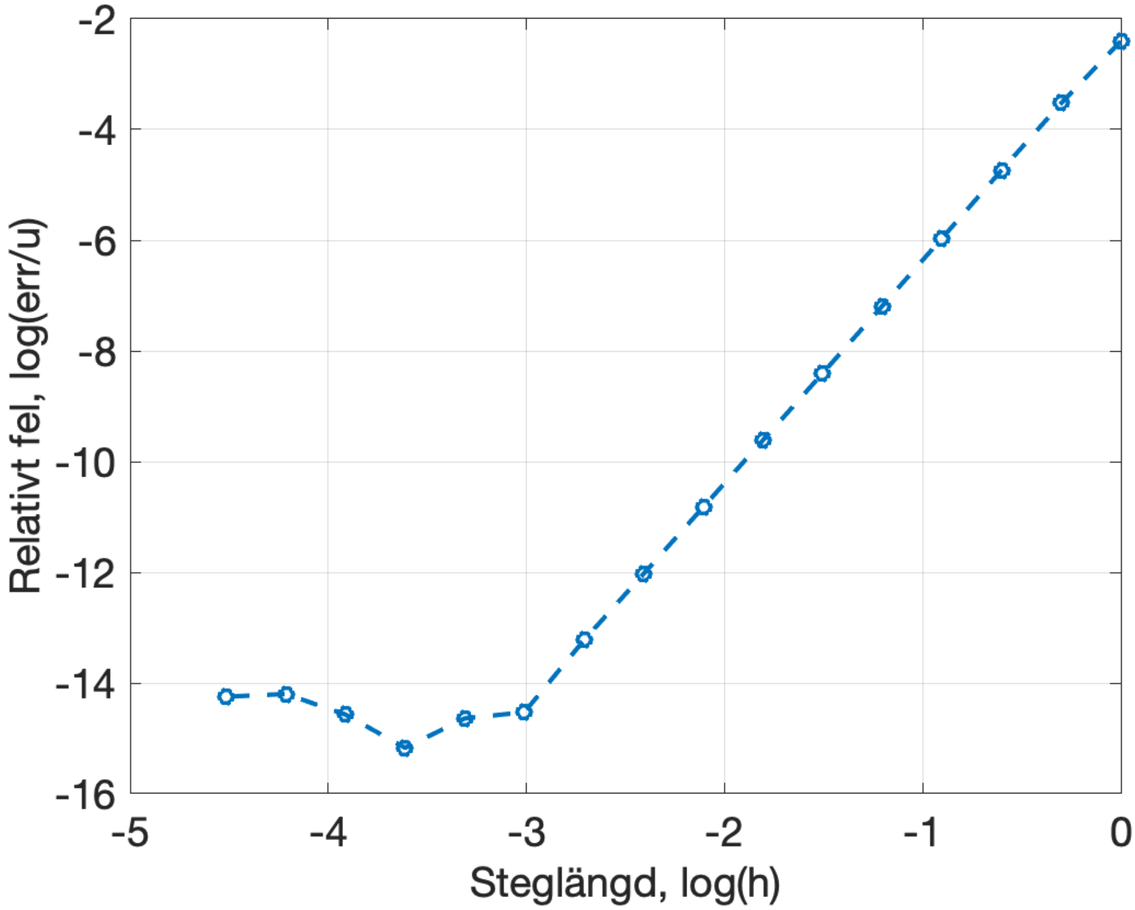
(d) Ge ett exempel på en frågeställning som skulle kunna besvaras med en *Monte Carlometod* som innehåller en kontinuerlig Markovprocess

Skriv in ditt svar här

Totalpoäng: 2

4 ODE graf

Bilden visar hur relativt fel beror av steglängden *h* hos en viss numerisk ODE-lösare (notera att det är logaritmisk skala). Besvara nedanstående frågor i relation till bilden. För godkänt på uppgiften måste samtliga svar vara korrekta



(a) Relatera begreppet *noggrannhetsordning* till bilden

Skriv in ditt svar här

(b) Relatera begreppet *maskinepsilon* till bilden

Skriv in ditt svar här

(c) Grafen är framställd genom en av de metoder som ingått i kursen. Ange vilken metod det är, inklusive motivering. Rätt svar utan motivering ger inte godkänt på deluppgiften.

Skriv in ditt svar här

Totalpoäng: 2

Delmål 3, 1TD403

Delmål 3: *kunna formulera och använda de olika algoritmer och numeriska metoder som ingår i kursen*

För att visa att du har nått delmålet behöver du klara minst en av de följande två uppgifterna.

5 Kostnad Monte Carlo

Kostnaden för att framställa en viss produkt **A** består av arbetskostnad (löner etc), kostnad för energiåtgång samt materialkostnad. Antag att arbetskostnad per tillverkad enhet är 3 kr och materialkostnaden är normalfördelad med väntevärde μ_p och standardavvikelse σ_p. Energikostnaden är fördelad enligt en sannolikhetsfördelning $f_{el}(x)$.

Skriv en algoritm som besvarar frågan "vilken är kostnaden i medeltal att tillverka produkt **A**" med hjälp av en Monte Carlo-simulering. Antag att den stokastiska process som beskriver energikostnaden $f_{el}(x)$ finns implementerad i funktionen **el_pris( )**. Funktionen returnerar alltså vid anrop ett slumptal $\in f_{el}(x)$. Inparametern *x* skapas inne i funktionen och behövs alltså inte i anropet.

Skriv in ditt svar här

1

Totalpoäng: 2

6 Stokastisk modell

En deterministisk modell över antalet rovdjur respektive bytesdjur i ett visst geografiskt område kan beskriva med följande deterministiska modell:

$$\begin{cases} y_1'(t) = k_1 y_1(t) - k_2(y_1(t))^2 - k_3 y_1(t) y_2(t) \\ y_2'(t) = k_3 y_1(t) y_2(t) - k_4 y_2(t) \end{cases}$$

Här betecknar y_1 antalet bytesdjur och y_2 antalet rovdjur. Parametrarna k_1-k_4 är olika utbyteskonstanter. Termen $k_2(y_1(t))^2$ anger hur bytesdjuren dör av svält pga brist på föda.

(a) Skriv om modellen som en stokastisk modell

(b) Skriv ned stochiometrimatrisen och propensitetsfunktionerna för den stokastiska modellen i (a).

Du kan skriva textfältet nedan, skriv t ex matriser som
[1 2 3
  4 5 6]
och pil kan skrivas som --->

Skriv in ditt svar här

Totalpoäng: 2

Delmål 4, 1TD403

Delmål 4: *känna till egenskaper hos numeriska metoder och matematiska modeller samt kunna genomföra analys för att undersöka dessa egenskaper*

För att visa att du nått delmålet behöver du klara minst en av de följande två uppgifterna.

7 Stabilitetsområden ODEer

Numeriska metoder för ODEer har olika *stabilitetsområden*. Förklara vad det är och hur det definieras. I relation till stabilitetsområden, beskriv också vad man menar med att en metod är *ovillkorligt stabil.*

Skriv in ditt svar här

Totalpoäng: 2

8 Noggrannhetsordning ODE

Hitta de faktorer *a>0* och *b>0* som medför att metoden som definieras

$$y_{i+1} = y_i + h \cdot (af(t_i, y_i) + bf(t_{i+1}, y_{i+1}))$$

har så hög noggrannhetsordning som möjligt.

Den här uppgiften är det sannolikt enklast att skriva på papper som lämnas in. Men du har också möjligheten att skriva i textfältet nedan. Ange i textfältet om du lämnar in på papper.

Skriv in ditt svar här

Totalpoäng: 2

i Delmål 5, 1TD403

Delmål 5: *kunna använda kunskap om egenskaper för att värdera och argumentera för olika metoders och modellers lämplighet i anslutning till en given problemställning*

För att visa att du nått delmålet behöver du klara minst en av de följande två uppgifterna.

OBS! Med tanke på vad det här delmålet handlar om är det ***viktigt att du verkligen åstadkommer en tydlig argumentation, som övertygar läsaren om att det alternativ du förespråkar skulle vara lämpligast***.

9 Argumentation stokastisk/deterministisk

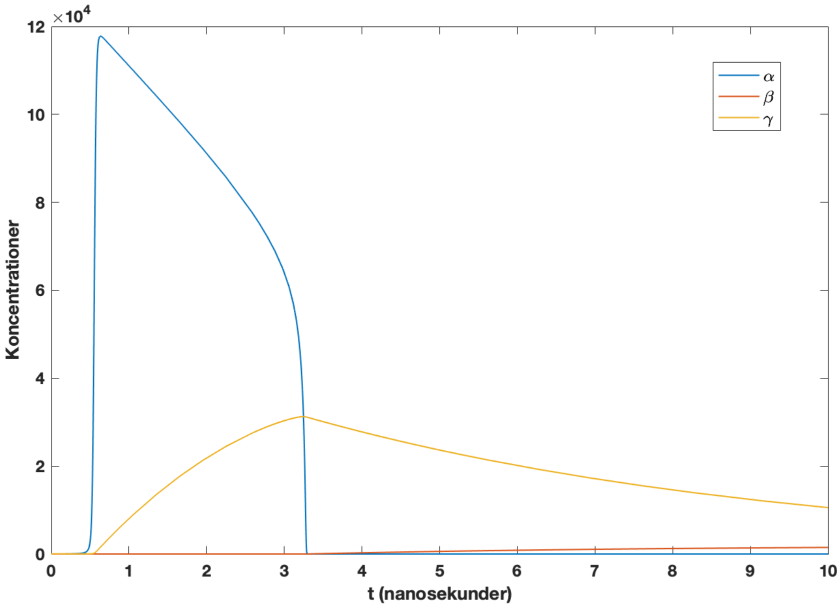
Ange vilken typ av metod (kolumn) är lämplig i vilken tillämpning (rad)

	Deterministisk metod	Stokastisk metod
Matematisk ODE	○	○
Kemisk reaktion med många molekyler	○	○
Integral 20D	○	○
Kemisk reaktion med få molekyler	○	○
Simulera trafikflöden i en stad	○	○
Lösa en väderprognosmodell	○	○
Optionsprissättning	○	○
Integral 2D	○	○

Totalpoäng: 2

10 BV2 - Argumentation - implicit/explicit

Bilden nedan visar en lösning av ett system av ODE:er, som beskriver en kemisk reaktion mellan tre kemikalier. Kurvorna som betecknas $\alpha(t), \beta(t), \gamma(t)$ visar hur koncentrationerna av respektive ämne förändras med tiden.



Antag att det finns ODE-lösare av samma typ som som finns i Scipy tillgängliga, dvs adaptiva explicita lösare respektive adaptiva implicita lösare.

Argumentera för val av lösare för just det här fallet. För godkänt måste det framgå
- vilken lösare man sannolikt bör välja i det här fallet
- varför den lösaren (sannolikt) är bättre i det här fallet
- vilken effekten (sannolikt) blir om man istället väljer den andra lösaren.
(Obs. Det krävs inga formler eller liknande här.)

Skriv in ditt svar här

Teckenf…

Ord: 0

Totalpoäng: 2

Högre betyg, 1TD403

De följande två uppgifterna motsvarar betyg 4 respektive betyg 5. För betyg 4 räcker det om du klarar en av uppgifterna fullständigt. För betyg 5 bör du klara av båda uppgifterna, med endast mindre fel. Du måste också ha klarat av alla kursmålen, kravet för betyg 3, för att nå upp till de högre betygen. Observera att om du har missat ett delmål i del A kan du eventuellt komplettera det i del B.

11 Stabilitet 2-steg RK-metoder

Heuns metod är ett exempel på en 2-stegs Runge-Kutta metod, men det finns även andra 2-stegs RK-metoder. En generell 2-stegs Runge-Kutta metod definieras

$$\begin{cases} k_1 = f(t_i, y_i) \\ k_2 = f(t_i + p \cdot h, y_i + q \cdot h \cdot k_1) \\ y_{i+1} = y_i + h(a_1 k_1 + a_2 k_2) \end{cases}$$

För att metoderna överhuvudtaget ska fungera kan man visa att följande krav gäller för koefficienterna a_1, a_2, p och q:

$$a_1 + a_2 = 1$$
$$a_2 \cdot p = \frac{1}{2}$$
$$a_2 \cdot q = \frac{1}{2}$$

Visa att alla 2-stegs Runge-Kutta metoder (enligt beskrivningen ovan) har samma stabilitetsområde och ange även stabilitetsvillkoret.

Den här uppgiften är det (sannolikt) enklast att besvara på papper, men du har också möjlighet att skriva i textfältet nedan. Ange i textfältet om du lämnar in på papper.

Skriv in ditt svar här

Totalpoäng: 10

12 Ny uppgift

Ersätt med din uppgiftstext...

Bakgrund:

I stokastisk simulering behöver man kunna generera slumptal i olika situationer. I den här uppgiften ska vi generera *n* stycken slumpmässiga punkter (x_i, y_i) i ett cirkelformat område ("disc" på engelska). För detta ska vi skriva en funktion *randomPointsOnDisc*. Vi ska använda polära koordinater, så att:

$x_i = r_i \cos(\theta_i), \qquad i = 1, \ldots, n$

$y_i = r_i \sin(\theta_i), \qquad i = 1, \ldots, n$

Vinklarna θ_i ska vara likformigt fördelade slumptal i intervallet ($-\pi$, π). Radierna r_i ska vara slumptal i intervallet (0, R), där R är det cirkelformade områdets radie.

Radierna ska *inte* vara likformigt fördelade. För att slumpa fram radierna ur lämplig fördelning använder vi en funktion *radie(R, n)*, som genererar en kolonnvektor med n stycken slumpmässiga radievärden.

Anropet till *randomPointsOnDisc* ska vara: *[x, y] = randomPointsOnDisc(R, n)*

(a) Skriv funktionen *randomPointsOnDisc* enligt ovanstående anvisningar. Du får anta att det finns en *färdig* funktion *radie*, som du inte själv behöver skriva.

För att visa hur funktionen *randomPointsOnDisc* kan användas ska du sedan skriva ett script som genererar 1000 punkter i ett område med radie 5 och plottar de diskreta punkterna. Plotten ska ha en förklarande rubrik.

(b) Skriv den ovan nämnda funktionen *radie(R, n)*. För att punkterna (x_i, y_i) ska blir någorlunda jämnt fördelade över området ska radierna slumpas enligt följande täthetsfunktion:

$$f(r) = \begin{cases} 0 & r < 0 \\ 2r/R^2 & 0 \leq r \leq R \\ 0 & r > R \end{cases}$$

Den här uppgiften lämnas lämpligen in på papper. Ange i textfältet om du lämnar in på papper.

Skriv in ditt svar här

Totalpoäng: 10