Uppsala universitet
Institutionen för informationsteknologi
Beräkningsvetenskap

Tentamen i *Numeriska metoder och simulering* 5.0 hp, 2016-10-17

Skrivtid: $08^{00} - 13^{00}$
Hjälpmedel: En formelsamling ingår i detta uppgiftsäfte. Inga övriga hjälpmedel tillåts.

En komplett lösning ska innehålla utförliga resonemang samt motivering till alla svar.

För betyg 3 krävs: Att man klarar varje delmål för betyg 3 nedan.
För betyg 4/5 krävs: Att man klarar både betyg 3 och uppgiften för betyg 4/5

Uppgifter som testar måluppfyllelse för betyg 3

Delmål 1: *kunna skriva ett Matlab-program som gör en numerisk simulering av något fenomen, givet en matematisk modell av fenomenet*

För att visa att du nått delmålet behöver du klara minst en av deluppgifterna i uppgift 1.

1. (a) Skriv ett program i Matlab för att med Matlab-kommandot `ode45` lösa differentialekvationen $y'(t) + 2\sin(y(t)) = t^2$, för $0 \leq t \leq 10$, med $y(0) = 0.5$.

   (b) Antag att du har tillgång till en Matlab-funktion `dygnsrytm(t)`, som gör stokastisk simulering av dygnsrytmen i en cell under t timmar. (För enkelhets skull tänker vi oss att alla modellparametrar och begynnelsevärden är hårdkodade inuti funktionen `dygnsrytm`, så att det bara blir en inparameter). Funktionen returnerar värdet d, som anger genomsnittlig dygnslängd under den simulerade tidsperioden. Skriv ett program i Matlab, som använder `dygnsrytm` i en Monte Carlo-metod för att beräkna genomsnittlig dygnslängd i cellens dygnsrytm under 200 timmar.

Delmål 2: *känna till viktiga begrepp i anslutning till numerisk simulering*

För att visa att du har nått delmålet behöver du klara minst två av deluppgifterna i uppgift 2.

2. Nedan finner du förklaringar av fyra begrepp som har ingått i kursen. Ange för varje förklaring vilket begrepp det är som avses.

   (a) Det som händer när man försöker representera ett tal vars absolutbelopp är mindre än det till beloppet minsta normaliserade flyttalet.

   (b) Felet i ett steg med den numeriska metoden utgående från exakta lösningen i senaste tidpunkten.

   (c) En metod där högerledet innehåller y_{i+1}, så att man måste lösa en ekvation i varje steg.

   (d) Störningskänsligheten hos den numeriska metoden.

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

För att visa att du nått delmålet behöver du klara minst en av deluppgifterna i uppgift 3.

3. (a) Ställ upp implicita Eulers metod för ekvationen $y'(t) + 2\sin(y(t)) = t^2$.

   (b) När vi löste det sista problemet i kursen använde vi algoritmen *Inverse Transform Sampling* för att slumpa fram numret på nästa reaktion. För att visa att du kan använda den algoritmen ska du nu tillämpa den på ett fall med fyra kemiska reaktioner. I det läge när nästa reaktion ska äga rum är sannolikheterna för de fyra reaktionerna: $P_1 = 0.1$, $P_2 = 0.4$, $P_3 = 0.2$, $P_4 = 0.3$. Som första steg i algoritmen *Inverse Transform Sampling* slumpar du fram ett tal u. Antag att värdet på u blev 0.62. Visa med handräkning hur algoritmen fortsätter i detta fall och vilket numret på nästa reaktion blir. Det måste framgå tydligt hur du kommer fram till slutsatsen.

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 deluppgifterna i uppgift 4.

4. (a) Genomför analys för att visa att explicita Eulers metod har noggrannhetsordning 1.
   (b) Du beräknar en integral med Monte Carlo-metod och använder 10000 slumpmässigt valda beräkningspunkter. Ungefär hur mycket noggrannare skulle resultatet ha blivit om du i stället hade använt 20000 punkter? Kom ihåg att motivera svaret.

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 deluppgifterna i uppgift 5.

5. (a) Du ska simulera en kemisk reaktion mellan molekyler i en cell, där det finns få exemplar av varje slags molekyl. Är det för denna simulering lämpligast att utgå från en deterministisk eller en stokastisk modell? Ge argument för ditt svar. (OBS! Avsikten med den här uppgiften är att testa att du kan argumentera för val av modell, så **det är viktigt att du verkligen åstadkommer en tydlig argumentation, som utmynnar i att läsaren känner sig övertygad om att det slags modell du förespråkar skulle vara lämpligast**.)
   (b) Du tänker skriva ett program för simulering där du utgår från en ODE-modell som inte är styv. Programmet ska användas i ett projekt där det kommer att göras många olika simuleringar med olika värden på modellparametrarna. Därför är det viktigt att programmets exekveringstid blir så kort som möjligt och samtidigt måste den tolerans som du har valt givetvis uppfyllas. Vilken av Heuns metod och klassiska Runge-Kuttas metod skulle du välja för denna simulering och varför? (OBS! Avsikten med den här uppgiften är att testa att du kan argumentera för val av metod, så **det är viktigt att du verkligen åstadkommer en tydlig argumentation, som utmynnar i att läsaren känner sig övertygad om att den metod du förespråkar skulle ge kortare exekveringstid**.)

Uppgift som testar måluppfyllelse för betyg 4

6. I kursen har vi haft ett exempel på en modell (formulerad av Lotka och Volterra) för samspelet mellan rovdjur och bytesdjur. Även om den ena djurarten inte äter den andra, så kan det vara svårt för djuren att samsas inom ett begränsat geografiskt område, med begränsad tillgång på föda och andra resurser. Följande modell (också den formulerad av Lotka och Volterra) beskriver sådan konkurrens mellan två djurarter.

$$
\begin{aligned}
N_1'(t) &= r_1 N_1(t) \frac{k_1 - N_1(t) - b_{12} N_2(t)}{k_1} \\
N_2'(t) &= r_2 N_2(t) \frac{k_2 - N_2(t) - b_{21} N_1(t)}{k_2}
\end{aligned}
$$

   Här är $N_1(t)$ och $N_2(t)$ "populationstätheten" vid tidpunkten t för respektive art. (Populationstätheten är antal individer per ytenhet inom det geografiska området). Modellparametrarna r_1 och r_2 uttrycker hur populationstätheten skulle utvecklas om det fanns obegränsade resurser. Modellparametrarna k_1, k_2, b_{12} och b_{21} står för hur konkurrens inom och mellan de två arterna påverkar populationstätheten. Som tidsskala väljer vi att t står för antal dygn från starttiden $t = 0$.

   Skriv ett program i Matlab som simulerar populationstäthetens utveckling för de två arterna under 365 dygn och som presenterar resultatet av simuleringen grafiskt. Modellparametrarnas värden samt begynnelsevärden för N_1 och N_2 ska matas in interaktivt när programmet körs.

   För att lösningen ska godkännas behöver du också ge argument för att ditt program skulle vara lämpligt för parametervärden som gör att systemet av differentialekvationer inte är styvt. (OBS! Precis som i argumentationsuppgifterna för betyg 3 är det viktigt att du åstadkommer en tydlig argumentation, som gör att läsaren känner sig övertygad om att ditt program skulle vara lämpligt under den givna förutsättningen.)

Uppgift som testar måluppfyllelse för betyg 5

7. Du ska nu skriva ett program som simulerar hur en satellit rör sig runt en planet. Vi inför ett koordinatsystem där planeten befinner sig i punkten (0, 0) och där satelliten vid tiden t befinner sig i punkten $(x(t),\ y(t))$. Följande modell beskriver hur satellitens position ändras

med tiden:

$$\begin{aligned} x''(t) &= -Cx(t)/\left(\sqrt{x(t)^2+y(t)^2}\right)^3 \\ y''(t) &= -Cy(t)/\left(\sqrt{x(t)^2+y(t)^2}\right)^3 \end{aligned}$$

där parametern C beror på gravitationen och på planetens massa.

Skriv ett Matlab-program för simulering av satellitens bana, baserat på denna modell. Vid körning ska användaren interaktivt få mata in dels värde på parametern C, dels satellitens startposition samt dess starthastighet i x- och y-riktningen. Programmet ska skriva ut en figur som visar den beräknade satellitbanan runt punkten (0, 0). OBS! Det ska inte vara en animering utan det räcker att i efterhand rita en bild av samtliga punkter längs banan. OBS! Det räcker inte att plotta varje koordinat som funktion av tiden, utan det är själva satellitbanan som ska visas i figuren.

När programmet är klart ska du beskriva något sätt att *praktiskt* avgöra om modellen i detta problem är styv eller ej. Det ska alltså inte vara en teoretisk analys, utan något praktiskt tillvägagångssätt som bygger på körning av olika varianter av ditt program. För att svaret ska godkännas behöver du tydligt förklara varför det tillvägagångssätt du föreslår skulle gå att använda för att avgöra om modellen är styv.

Uppsala universitet
Institutionen för informationsteknologi
Avd. för beräkningsvetenskap

Blandade formler i Beräkningsvetenskap I och II

1. Flyttal och avrundningsfel

Ett flyttal $fl(x)$ representeras enligt

$$fl(x) = \hat{m} \cdot \beta^e, \quad \hat{m} = \pm(d_0.d_1d_2, \ldots, d_{p-1}), \quad 0 \leq d_i < \beta, \quad d_0 \neq 0, \quad L \leq e \leq U,$$

där β betecknar bas och p precision.
Ett flyttalssystem defineras $FP(\beta, p, L, U)$.
Maskinepsilon (avrundningsenheten) $\epsilon_M = \frac{1}{2}\beta^{1-p}$ och kan defineras som det minsta tal ϵ sådant att $fl(1+\epsilon) > 1$.

2. Linjära och ickelinjära ekvationer

Newton-Raphsons metod: $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$
För system: $x_{k+1} = x_k - [F']^{-1}F(x_k)$, där x_k och $F(x_k)$ är vektorer och F' är Jacobianen.

Fixpunktsiteration för $x = g(x)$: $x_{k+1} = g(x_k)$

Konvergenskvot, konvergenshastighet

$$\lim_{k\to\infty} \frac{|x_{k+1} - x_*|}{|x_k - x_*|^r} = C,$$

där C är en konstant, och r anger konvergenshastigheten ($r = 1$ betyder t ex linjär konvergens).

Allmän feluppskattning

$$|x_k - x^\star| \leq \frac{|f(x_k)|}{\min |f'(x)|}$$

Konditionstalet $\mathrm{cond}(A) = \|A\| \cdot \|A^{-1}\|$ mäter känsligheten för störningar hos ekvationssystemet $Ax = b$. Det gäller att

$$\frac{\|\Delta x\|}{\|x\|} \leq \mathrm{cond}(A)\frac{\|\Delta b\|}{\|b\|},$$

där $\Delta x = x - \hat{x}$ och $\Delta b = b - \hat{b}$.

Normer (vektor- respektive matrisnorm)

$$\| x \|_2 = \sqrt{|x_1|^2 + \ldots + |x_n|^2} \qquad \| x \|_1 = \textstyle\sum_i |x_i| \qquad \| x \|_\infty = \max_i\{|x_i|\}$$
$$\| A \|_1 = max_j(\textstyle\sum_i |a_{ij}|) \qquad \| A \|_\infty = max_i(\textstyle\sum_j |a_{ij}|)$$

3. Approximation

Newtons interpolationspolynom $p(x)$ då vi har n punkter $(x_1, y_1), \ldots (x_n, y_n)$ bygger på ansatsen

$$p(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2) + \ldots + a_{n-1}(x - x_1) \cdots (x - x_{n-1})$$

Minstakvadratapproximationen till punktmängden $(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)$ med ett n:egradspolynom $p(x) = a_0 1 + a_1 x + \ldots + a_n x^n$ kan formuleras som ett överbestämt ekvationssystem $Ax = b$, där A är $m \times n$, $m > n$. Minstakvadratlösningen kan fås ur normalekvationerna

$$A^T A x = A^T b$$

4. Ordinära differentialekvationer

Eulers metod (explicit Euler): $y_{k+1} = y_k + hf(x_k, y_k)$, n.o. 1
Implicit Euler (Euler bakåt): $y_{k+1} = y_k + hf(x_{k+1}, y_{k+1})$, n.o. 1
Trapetsmetoden: $y_{k+1} = y_k + \frac{h}{2}(f(x_k, y_k) + f(x_{k+1}, y_{k+1}))$, n.o. = 2
Heuns metod (tillhör gruppen Runge-Kuttametoder):

$$\begin{cases} K_1 = f(x_k, y_k) \\ K_2 = f(x_{k+1}, y_k + hK_1) \\ y_{k+1} = y_k + \frac{h}{2}(K_1 + K_2) \end{cases}$$

n.o. = 2
Klassisk Runge-Kutta:

$$\begin{cases} K_1 = f(x_k, y_k) \\ K_2 = f(x_k + \frac{h}{2}, y_k + \frac{h}{2}K_1) \\ K_3 = f(x_k + \frac{h}{2}, y_k + \frac{h}{2}K_2) \\ K_4 = f(x_{k+1}, y_k + hK_3) \\ y_{k+1} = y_k + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \end{cases}$$

n.o. = 4

5. Numerisk integration

Trapetsformeln
Beräkning på ett delintervall med steglängd $h_k = x_{k+1} - x_k$

$$\int_{x_k}^{x_{k+1}} f(x)\, dx = \frac{h_k}{2}[f(x_k) + f(x_{k+1})]$$

Sammansatt formel på helt intervall $[a\ b]$, då ekvidistant steglängd $h = h_k$:

$$\int_a^b f(x)\,dx \approx \frac{h}{2}[f(x_0) + 2f(x_1) + \ldots + 2f(x_{N-1}) + f(x_N)]$$

Diskretiseringsfelet R på helt intervall $[a\ \ b]$, dvs $\int_a^b f(x)\,dx = T(h) + R$ är

$$R = -\frac{(b-a)}{12}h^2 f''(\xi).$$

Funktionsfelet (övre gräns): $(b-a)\cdot\epsilon$, där ϵ är en övre gräns för absoluta felet i varje funktionsberäkning.

Simpsons formel

Beräkning på ett dubbelintervall med steglängd h

$$\int_{x_k}^{x_{k+2}} f(x)\,dx = \frac{h}{3}[f(x_k) + 4f(x_{k+1}) + f(x_{k+2})]$$

Sammansatt formel på helt intervall $[a\ b]$, då ekvidistant steglängd $h = h_k$:

$$\int_a^b f(x)\,dx \approx \frac{h}{3}[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \ldots + 2f(x_{N-2}) + 4f(x_{N-1}) + f(x_N)]$$

Diskretiseringsfelet R på helt intervall $[a\ \ b]$, dvs $\int_a^b f(x)\,dx = S(h) + R$ är

$$R = -\frac{(b-a)}{180}h^4 f''''(\xi).$$

Funktionsfelet: Samma som för trapetsformeln, se ovan.

6. Richardsonextrapolation

Om $F_1(h)$ och $F_1(2h)$ är två beräkningar (t ex ett steg i en beräkning av en integral eller en ODE) med en metod av noggrannhetsordning p med steglängd h respektive dubbel steglängd $2h$ så är

$$R(h) = \frac{F_1(h) - F_1(2h)}{2^p - 1}$$

en uppskattning av den ledande termen i trunkeringsfelet i $F_1(h)$. Kan även användas för att förbättra noggrannheten i $F_1(h)$ genom

$$F(h) = F_1(h) + \frac{F_1(h) - F_1(2h)}{2^p - 1}.$$

7. Numerisk derivering

För numerisk derivering används s k differensformler

$f'(x) \approx \frac{f(x+h)-f(x-h)}{2h}$, centraldifferens
$f'(x) \approx \frac{f(x+h)-f(x)}{h}$, framåtdifferens
$f'(x) \approx \frac{f(x)-f(x-h)}{h}$, bakåtdifferens
$f''(x) \approx \frac{f(x+h)-2f(x)+f(x-h)}{h^2}$

8. Monte Carlometoder

Den övergripande strukturen för Monte Carlosimuleringar är

```
Indata N (antal försök)
for i = 1:N
    Utför en stokastisk simulering
    resultat(i) = resultatet av simuleringen
end
slutresultat genom någon statistisk beräkning, t ex medelvärdet mean(resultat)
```

Noggrannhetsordning för Monte carlometoder är $\mathcal{O}(\frac{1}{\sqrt{N}})$, där N är antal samplingar.
Kumultativ fördelningsfunktion: $F(x) = \int_{-\infty}^{x} f_X(y)dy$
Normalfördelning

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

Aritmetiskt medelvärde baserat på N realisationer x_i av slumpvariablen X: $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$.

9. Taylorutveckling

Taylorutveckling av $y(x_k + h)$ kring x_k:

$$y(x_k + h) = y(x_k) + hy'(x_k) + \frac{h^2}{2!}y''(x_k) + \frac{h^3}{3!}y'''(x_k) + \mathcal{O}(h^4)$$