Uppsala universitet
Institutionen för informationsteknologi
Beräkningsvetenskap

Tentamen i *Numeriska metoder och simulering* 5.0 hp, 2015-10-21

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 $y'(t) - sin(y^2(t)) = 0$, för $0 \leq t \leq 2$, med $y(0) = 0.5$.

   (b) Antag att du har tillgång till en Matlab-funktion `partikel(t)`, som gör en stokastisk simulering av hur en partikel inuti en cell rör sig under t sekunder. Funktionen returnerar värdet d, som anger hur långt från utgångspunkten partikeln har rört sig under den simulerade tidsperioden. Skriv ett program i Matlab, som använder `partikel` i en Monte Carlo-metod för att beräkna hur långt från utgångspunkten en partikel i cellen i genomsnitt hinner röra sig under 2 sekunder.

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) En modell utan slumpmoment, så att utdata beror entydigt på indata.

   (b) En metod för numerisk lösning av ODE, där högerledet i metoden bara beror på kända värden.

   (c) Förlust av signifikanta siffror vid subtraktion mellan jämnstora flyttal.

   (d) h-potensen i globala trunkeringsfelet hos en numerisk metod för lösning av ODE (där h är steglängden).

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 $y'(t) = sin(y^2(t))$.

   (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 tre kemiska reaktioner. I det läge när nästa reaktion ska äga rum är sannolikheterna för de tre reaktionerna: $P_1 = 0.4$, $P_2 = 0.2$, $P_3 = 0.4$. Som första steg i algoritmen *Inverse Transform Sampling* slumpar du fram ett tal u. Antag att värdet på u blev 0.53. Visa med handräkning hur algoritmen fortsätter i detta fall och vilket numret på nästa reaktion blir.

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) Härled stabilitetsvillkoret för implicita Eulers metod.

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) I vissa lägen kan det vara fördelaktigt med en stokastisk metod även om den matematiska modellen är deterministisk. Ett exempel är integraler. Ange minst ett skäl till att en Monte Carlo-metod i vissa fall kan vara det enda rimliga alternativet för integralberäkning.

   (b) Du tänker skriva ett program för simulering där du utgår från en ODE-modell som inte är styv. När programmet är färdigt ska det användas för en parameterstudie, där olika modellparametrars värden varieras, så att man kan studera effekten av olika kombinationer av parametervärden. Det kommer att bli en körning av ditt program för varje ny kombination av parametervärden. Eftersom du vill testa ett stort antal sådana kombinationer är det viktigt att programmets exekveringstid blir så kort som möjligt (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. Du har fått jobb på ett konsultföretag, där du blir inblandad i ett uppdrag för Räddningsverket. Det handlar om datorsimulering av hur ett objekt vinschas upp till en helikopter. Det kan exempelvis gälla en person som räddas ur en livbåt till havs. När objektet vinschas, så kommer det att göra en pendelrörelse under helikoptern.

   Som ett första steg i projektet ska du simulera pendelrörelsen med fix längd på pendeln, det vill säga hur objektet pendlar när helikoptern är på en fix plats i luften och ingen vinschning pågår. Den pendelrörelsen kan modelleras med differentialekvationen:

$$\theta''(t) + K\theta'(t) + \frac{g}{L}\sin(\theta(t)) = 0$$

där $\theta(t)$ är pendelns vinkel, L är pendelns längd i meter, g är tyngdaccelerationen och K är en koefficient som beskriver hur friktionen dämpar pendeln svängningar.

Skriv ett Matlab-program för simulering av pendelrörelsen enligt denna modell. För simuleringen kan du anta att $L = 10$, $g = 9.82$, $K = 0.1$. Begynnelsevärden ska matas in interaktivt när programmet körs. Resultatet av simuleringen ska presenteras grafiskt. Du ska vidare ge argument för att ditt program skulle vara lämpligt under förutsättning att differentialekvationen inte är styv.

Uppgift som testar måluppfyllelse för betyg 5

7. Du fortsätter med ditt konsultuppdrag för Räddningsverket. I projektets huvuddel gäller det att simulera hur objektet vinschas upp till helikoptern, det vill säga att objektet med tiden kommer allt närmare helikoptern. Vinschningsprocessen kan modelleras med en ordinär differentialekvation som beskriver en pendel av *variabel längd*:

$$\theta''(t) + 2\frac{r'(t)}{r(t)}\theta'(t) + \frac{g}{r(t)}\sin(\theta(t)) = 0$$

där $\theta(t)$ är pendelns vinkel, $r(t)$ är pendelns längd och g är tyngdaccelerationen. Själva vinschningen, alltså hur pendelns längd varierar med tiden, beskrivs av en ”vinschningsstrategi” i form av ytterligare en differentialekvation:

$$r'(t) = f(t, \theta(t), \theta'(t)).$$

Syftet med datorsimuleringarna är att Räddningsverket vill kunna studera effekterna av olika vinschningsstrategier.

Skriv ett Matlab-program som genomför simulering enligt denna modell. Begynnelsevärden ska matas in interaktivt när programmet körs. Resultatet av simuleringen ska presenteras grafiskt. Du ska vidare ge argument för att ditt program skulle vara lämpligt under förutsättning att differentialekvationen inte 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

$$fl(x) = \hat{m} \cdot \beta^e, \quad \hat{m} = \pm(d_0.d_1 d_2, \ldots, d_{p-1}), \quad 0 \leq d_i < \beta, \quad d_0 \neq 0, \quad L \leq e \leq U,$$

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

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

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

$$A^T Ax = A^T b$$

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

$$
\begin{aligned}
f'(x) &\approx \tfrac{f(x+h)-f(x-h)}{2h}, \quad \text{centraldifferens}\\
f'(x) &\approx \tfrac{f(x+h)-f(x)}{h}, \quad \text{framåtdifferens}\\
f'(x) &\approx \tfrac{f(x)-f(x-h)}{h}, \quad \text{bakåtdifferens}\\
f''(x) &\approx \tfrac{f(x+h)-2f(x)+f(x-h)}{h^2}
\end{aligned}
$$

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:

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