

Uppsala universitet  
Institutionen för informationsteknologi  
Beräkningsvetenskap

## Tentamen i *Numeriska metoder och simulering* 5.0 hp, 2016-01-16

**Skrivtid:** 09<sup>00</sup> – 14<sup>00</sup>

**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  $y'(t) - \cos(t + y(t)) = 0$ , för  $0 \leq t \leq 2$ , med  $y(0) = 0.5$ .
- (b) Antag att du har tillgång till en Matlab-funktion `aktiekurs(x0,T)`. Funktionen utgår från dagens aktiekurs,  $x_0$  kronor, och beräknar med stokastisk simulering hur aktiekursen kommer att utvecklas  $T$  dagar framåt i tiden. Det värde som returneras är aktiekursen om  $T$  dagar enligt den stokastiska simuleringen. Skriv ett program i Matlab, som använder `aktiekurs` i en Monte Carlo-metod för att beräkna en uppskattning av vad aktiekursen kommer att vara om  $T$  dagar.

**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 med slumpinslag, så att utdata inte beror entydigt på indata.
  - (b) En metod för numerisk lösning av ODE, som bygger på att derivatan approximeras med en differens.
  - (c) Det minsta tal  $x$  som vid flyttalsräkning ger att  $1 + x$  blir större än 1 (det vill säga att  $x$  är det minsta tal som kan flyttalsadderas till 1 utan att "avrundas bort").
  - (d) Den numeriska lösningen till ett ODE-problem går mot den exakta lösningen då  $h$  går mot noll.

**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 Trapetsmetoden för ekvationen  $y'(t) - \cos(t + y(t)) = 0$ .
  - (b) Formulera en översiktlig pseudokod för Gillespies algoritm. Pseudokoden behöver inte innehålla formler, utan det räcker att i text beskriva vad som händer i de olika stegen i algoritmen.

**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 implicita Eulers metod har noggrannhetsordning 1.
  - (b) Härled stabilitetsvillkoret för explicita Eulers metod.

**Delmål 5:** *kunna använda kunskap om egenskaper för att värdera och argumentera för olika metodens 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 ämnena A och B. Reaktionen är tänkt att ske i ett provrör där de två ämnena är väl blandade med varandra och där det finns stora mängder av både A och B. 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 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 Trapetsmetoden och Heuns metod skulle du välja för denna simulering? Ge argument för ditt svar. (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 en ekolv blandas tre kemikalier som vi kallar  $A$ ,  $B$  respektive  $C$ . En reaktion äger rum och koncentrationerna av dessa kemikalier kommer att ändras med tiden. Koncentrationen av  $A$ ,  $B$  respektive  $C$  vid tidpunkten  $t$  sekunder betecknas med  $\alpha(t)$ ,  $\beta(t)$ ,  $\gamma(t)$ . Den matematiska modellen för reaktionen är ett system av ordinära differentialekvationer:

$$\begin{aligned}\alpha'(t) &= s(\beta(t) - \alpha(t)\beta(t) + \alpha(t) - q\alpha(t)^2) \\ \beta'(t) &= s^{-1}(-\beta(t) - \alpha(t)\beta(t) + \gamma(t)) \\ \gamma'(t) &= w(\alpha(t) - \gamma(t))\end{aligned}$$

Skriv ett program i Matlab som simulerar reaktionernas förlopp under de första tio sekunderna och som presenterar resultatet av simuleringen grafiskt. Modellparametrarnas värden ( $s$ ,  $q$  och  $w$ ) samt begynnelsevärdena ska matas in interaktivt när programmet körs. Du ska vidare ge argument för att programmet skulle vara lämpligt för parametervärden som gör att systemet av differentialekvationer är styvt.

## Uppgift som testar måluppfyllelse för betyg 5

7. Problemet i föregående uppgift kommer att vara styvt för vissa värden på  $s$ ,  $q$  och  $w$  men inte för andra. Därför kunde det vara praktiskt att ha en numerisk metod som i sig har en parameter, som kan väljas olika beroende på om problemet är styvt eller inte. En sådan numerisk metod är:

$$y_{i+1} = y_i + h((1 - \theta)f(t_i, y_i) + \theta f(t_{i+1}, y_{i+1})),$$

där  $\theta$  är ett reellt tal och  $0 \leq \theta \leq 1$ . Detta är i själva verket inte *en*, utan en hel *familj* av metoder. För varje specifikt värde på  $\theta$  får vi en specifik metod.

Tänk dig nu att denna familj av metoder finns implementerad som en funktion `thetamethod` i Matlab och att metodanropet ser ut som följer:

```
[t,y] = thetamethod(odefun, tspan, y0, theta)
```

Metoden har alltså utöver metodparametern  $\theta$  (`theta`) samma in- och utparametrar som Matlabs inbyggda ODE-lösare. Vi förutsätter att `thetamethod` automatiskt ställer in en lämplig steglängd, så att  $h$  inte behöver vara inparameter.

Tänk dig att du dessutom har tillgång till en Matlab-funktion `isStiff` som kan bedöma om ODE-systemet är styvt eller ej. Anropet `isStiff(odefun)` ger värdet 'sann' om den ODE som beskrivs i `odefun` bedöms vara styv, annars värdet 'falsk'.

Med användning av dessa två funktioner kan man skriva ett program i Matlab som simulerar reaktionernas förlopp enligt modellen i föregående uppgift. Vid anropet av `thetamethod` ska då  $\theta$  väljas på lämpligt sätt beroende på om ODE-systemet i den aktuella körningen är styvt eller inte. Med "lämpligt" menas att vi väljer ett värde på  $\theta$  som kan förväntas ge kort exekveringstid (jämfört med vad som skulle bli fallet med andra  $\theta$ -värden) för att genomföra simuleringen med önskad noggrannhet. Gör nu följande:

- Skissa ett sådant simuleringsprogram. Du behöver inte skriva in precis vilka  $\theta$ -värden som ska väljas, eftersom det diskuteras nedan.
- För fallet när ODE-systemet är styvt finns ett val av  $\theta$ -värde som är direkt olämpligt. Vilket värde är det och varför är det olämpligt?
- Bland de  $\theta$ -värden som är lämpliga när ODE-systemet är styvt finns ett som är bäst därför att det ger högre noggrannhetsordning än övriga val. Vilket  $\theta$ -värde är det? Redovisa en analys som visar att det valet av  $\theta$  faktiskt ger högst noggrannhetsordning.
- För fallet när ODE-systemet *inte* är styvt finns det två värden på  $\theta$  som kan tänkas vara lämpligast. Vilka värden är det? Varför skulle just dessa två vara lämpligare än de övriga värdena på  $\theta$  i det fallet? Varför är det inte självklart vilket av de två värdena som är allra lämpligast?

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

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

där  $\beta$  betecknar bas och  $p$  precision.

Ett flyttalssystem definieras  $FP(\beta, p, L, U)$ .

Maskinepsilon (avrundningsenheten)  $\epsilon_M = \frac{1}{2}\beta^{1-p}$  och kan definieras som det minsta tal  $\epsilon$  sådant att  $fl(1 + \epsilon) > 1$ .

### 2. Linjära och icke linjära 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*

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

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

*Allmän feluppskattning*

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

*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)*

$$\begin{aligned} \|x\|_2 &= \sqrt{|x_1|^2 + \dots + |x_n|^2} & \|x\|_1 &= \sum_i |x_i| & \|x\|_\infty &= \max_i \{|x_i|\} \\ \|A\|_1 &= \max_j (\sum_i |a_{ij}|) & \|A\|_\infty &= \max_i (\sum_j |a_{ij}|) \end{aligned}$$

### 3. Approximation

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

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

*Minstakvadratapproximationen* till punktmängden  $(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)$  med ett  $n$ :egradspolynom  $p(x) = a_0 + a_1x + \dots + a_nx^n$  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  $h_k = x_{k+1} - x_k$

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

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

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

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  $\epsilon$  är en övre gräns för absoluta felet i varje funktionsberäkning.

*Simpsons formel*

Beräkning på ett dubbelintervall med steglängd  $h$

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

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

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

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

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

## 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.

Kumulativ 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 slumpvariabeln  $X$ :  $\mu = \frac{1}{N} \sum_{i=1}^N x_i$ .

## 9. Taylorutveckling

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

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