

TENTAMEN, DEL 2

SF1524

GRUNDLÄGGANDE NUMERISKA METODER OCH PROGRAMMERING

Fredag 16 mars 2018 kl 8.00-11.00

Rättas endast om del 1 är godkänd. Max antal poäng på denna del är 50. Betygsgräns: 10p D, 20p C, 30p B, 40p A. Svar skall motiveras och uträkningar redovisas. Korrekt svar utan motivering eller med felaktig motivering medför poängavdrag.

Inga hjälpmedel är tillåtna (ej heller miniräknare).

P1. Givet är följande tabell av mätvärden

t	0.2	0.3	0.4	0.5	0.6
$f(t)$	2	3	4	2	2

- (5 p) **a.** Använd värdena i tabellen och trapetsregeln med steget $h = 0.1$ för att beräkna integralen

$$\int_{0.2}^{0.6} f(t) dt$$

Gör en feluppskattning. Funktionen f antas vara snäll.

Lösning:

$$T(0.1) = 0.1(2/2 + 3 + 4 + 2 + 2/2) = \frac{11}{10} = 1.1$$

$$T(0.2) = 0.2(2/2 + 4 + 2/2) = \frac{12}{10} = 1.2$$

$$\text{Feluppskattning : } \frac{|T(0.2) - T(0.1)|}{3} = \frac{1}{3} \cdot 10^{-1}$$

-
- (3 p) **b.** Värdena på $f(t)$ är uppmätta med en osäkerhet på ± 0.2 . Uppskatta en gräns för det totala felet i integralvärdet som du beräknade i **a**.

Lösning: Totala felet:

$$\|I(f) - T(h, \tilde{f})\| = \|I(f) - T(h, f) + T(h, f) - T(h, \tilde{f})\|$$

består av:

Diskretiseringsfel: $\|I(f) - T(h, f)\|$ uppskattades i a) till $\frac{1}{3} \cdot 10^{-1}$

+

Funktionsfel: $\|T(h, f) - T(h, \tilde{f})\| \leq E_f(b - a) = 0.2 \cdot (0.6 - 0.2) = 0.08$

- (2 p) **c.** Minskar det totala felet om du får fler mätvärden?
-

Lösning: Fler mätvärden minskar diskretiseringsfelet men inte funktionsfelet. Eftersom det totala felet domineras av funktionsfelet så hjälper det inte att få fler mätvärden.

P2. Vi vill beräkna $y(t)$ som uppfyller den ordinära differentialekvationen

$$y''' + 3y' - 3e^{-y} + t^2 = 0, \quad y(0) = 0, \quad y'(0) = -1, \quad y''(0) = 1.$$

- (3 p) **a.** Inför lämpliga beteckningar och skriv ekvationen som ett system av första ordningens differentialekvationer. Glöm inte begynnelsevillkoren!
-

Lösning: Inför tre nya variabler och formulera differentialekvationer för var och en av de nya variablerna

$$\begin{array}{lll} u_1 = y & u_1' = u_2 & u_1(0) = 0 \\ u_2 = y' & u_2' = u_3 & u_2(0) = -1 \\ u_3 = y'' & u_3' = -3u_2 + 3e^{-u_1} - t^2 & u_3(0) = 1 \end{array}$$

- (3 p) **b.** Vad blir approximationen av $y'(1)$ med Framåt Euler (explicit Euler) och steglängden 0.5?
-

Lösning: Framåt Euler för $u' = f(t, u)$: $u^{n+1} = u^n + hf(t^n, u^n)$. Med steglängd $h=0.5$ och $f(t,u)$ från **a.** får vi

$$\begin{array}{ll} t^0 = 0 & t^1 = 0 + h = 0.5 \\ u^0 = \begin{pmatrix} 0 \\ -1 \\ 2 \end{pmatrix} & u^1 = u^0 + h \begin{pmatrix} -1 \\ 1 \\ -3(-1) + 3e^0 - 0^2 \end{pmatrix} = \begin{pmatrix} -0.5 \\ -0.5 \\ 4 \end{pmatrix} \end{array}$$

$$t^2 = 0.5 + h = 1$$

$$u^2 = \begin{pmatrix} -0.5 \\ -0.5 \\ 4 \end{pmatrix} + 0.5 \begin{pmatrix} -0.5 \\ 4 \\ - \end{pmatrix}$$

$$y'(1) = -0.5 + 0.5 \cdot 4 = 1.5$$

- (8 p) c. Skriv en detaljerad algoritm i MATLAB som beräknar $y(t)$ fram till $t = 5$ och plottar $y(t)$. Algoritmen ska vara baserad på Framåt Euler-metoden.
-

Lösning:

```
f=@(t,u) [u(2); u(3); -3*u(2)+3*exp(-u(1))-t^2];
N=500; %antal delintervall, ändra N för att ändra noggrannheten
h=5/N;
u0=[0 -1 1]; %begynnelsevärden
t=[0:h:5]';
U=zeros(N+1,length(u0)); U(1,:)=u0; %initiera tabell för lösningen
for i=1:N
    U(i+1,:)=U(i,:)+h*f(t(i),U(i,:))';
end
figure;
plot(t,U(:,1))
```

- (7 p) d. Utöka MATLAB-programmet så att det beräknar båglängden L av $y(t)$ i intervallet $t \in [0, 5]$. Båglängden ges av

$$L = \int_0^5 \sqrt{1 + y'(t)^2} dt.$$

MATLABs inbyggda funktioner för integration får ej användas här.

Lösning: Koden i b. utökas med följande satser

```
F=sqrt(1+U(:,2).^2);
L=h*(F(1)/2+sum(F(2:end-1))+F(end)/2);
disp([h L])
```

- (2 p) e. Beskriv hur man kan gå tillväga för att uppskatta tillförlitligheten i approximationen av L .
-

Lösning: Lös problemet med $N=500, 1000, 2000$, osv. Skillnaden i L -värden ger en skattning av felet.

- P3.** Du vill beräkna derivatan av en snäll funktion $f(x)$. Du känner inte till dess precisa form, men har tillgång till en MATLAB-funktion med funktionshuvud

```
function f=func(x)
```

som givet x ger dig $f(x)$. Du har också tillgång till en funktion i MATLAB där en numerisk metod för att approximera $f'(x)$ finns implementerad. Den har funktionshuvudet

```
function Dh=numder(x,h)
```

där h är en steglängd och x punkten där man vill approximera derivatan. Till exempel kan differensapproximationen

$$D(h) = \frac{f(x+h) - f(x)}{h} \quad (1)$$

vara implementerad i **numder**.

- (5 p) a. Visa att differensapproximationen i (1) minst har noggrannhetsordning ett.
-

Lösning: Använd Taylorutveckling

$$\begin{aligned} \text{Felet} &= D(h) - f'(x) = \frac{1}{h}(f(x+h) - f(x)) - f'(x) = \\ &= \frac{1}{h} \left(f(x) + hf'(x) + h^2 \frac{f''(x)}{2} + O(h^3) - f(x) \right) - f'(x) = h \frac{f''(x)}{2} + O(h^2). \end{aligned}$$

dvs vi har minst noggrannhetsordning ett.

- (5 p) b. Du vill veta vad den numeriska metoden som är implementerad i **numder** har för noggrannhetsordning. Tyvärr är funktionen **numder** dåligt kommenterad. Förklara hur du skulle gå till väga för att bestämma noggrannhetsordningen genom numeriska experiment. Motivera ditt svar.
-

Lösning: Jag beräknar $D(4h)$, $D(2h)$ och $D(h)$ för t.ex. $h = 10^{-3}$ i en punkt x och antar att ordningen är p . Då är

$$D(4h) = D(0) + (4h)^p c_1 + O(h^{p+1})$$

$$D(2h) = D(0) + (2h)^p c_1 + O(h^{p+1})$$

$$D(h) = D(0) + (h)^p c_1 + O(h^{p+1})$$

Bilda kvoten

$$q = \frac{D(4h) - D(2h)}{D(2h) - D(h)} = \frac{(4h)^p - (2h)^p}{(2h)^p - (h)^p} = \frac{4^p - 2^p}{2^p - 1} = 2^p \frac{2^p - 1}{2^p - 1} = 2^p$$

Om $q \approx 2$ så är $p=1$, om $q \approx 4$ så är $p=2$, osv.

- (7 p) c. Du vill nu hitta det x som ger $f(x) = 10$. Det finns en lösning i närheten av $x = 2$. Skriv ett MATLAB-program som hittar detta x med ett fel som är mindre än $0.5 \cdot 10^{-8}$. Programmet kan anropa på funktionerna som givits ovan.
-

Lösning: Använd Newton-Raphsons metod för att lösa den icke linjära ekvationen $f(x) - 10 = 0$.

```
x=2; %startgissning
h=0.01 %steglängd
dx=1;
while abs(dx)>0.5e-8
    f=func(x)-10;
    fprim=numder(x,h);
    dx=-f/fprim;
    x=x+dx;
end
```
