Tentamen för Matematisk överbryggningskurs

Tid och plats: 2018-03-16, kl 8:30-12:30, L. **Ansvarig:** Jacques Huitfeldt, 031-7721093.

Betygsgränser: 20, 30 resp. 40 poäng. Tentan omfattar totalt 50 poäng.

Hjälpmedel: Inga hjälpmedel tillåtna, utom bifogat formelblad.

Uppgift 1. Antag $\hat{x}$ är en lösning till ekvationen $f(x) = 0$ med $f'(\hat{x}) \neq 0$. Visa att om Newtons metod konvergerar mot $\hat{x}$ så är konvergensen kvadratisk.

Uppgiften ger maximalt **(8p)**

Uppgift 2. En kula med massan m hänger i ett gummiband med neutrallängden L och fjäderkonstanten k. Gummibandet är fäst i en punkt som vi väljer som origo i koordinatsystemet. Kulans rörelse i xy-planet ges av begynnelsevärdesproblemet

$$\begin{cases} mx'' = -s(r)\frac{x}{r} \\ my'' = -s(r)\frac{y}{r} - mg \end{cases}$$

där $r = \sqrt{x^2 + y^2}$ och $s(r)$ ges av: $s(r) = k(r - L)$ då $r \geq L$ och $s(r) = 0$ då $r < L$.

(a). Skriv om begynnelsevärdesproblemet som ett första ordningens system. Använd vektorbeteckningar.

(b). Skriv den kod i MATLAB som behövs för att beskriva högerledet i ekvationen som en funktionsfil.

(c). Skriv den kod i MATLAB som behövs för att lösa problemet med `ode45` och rita upp kulans läge som funktion av tiden. Tag $m = 0.12$, $L = 0.4$, $k = 10$, begynnelseläget $(x(0), y(0)) = (0.3, -0.4)$, begynnelsehastigheten $(x'(0), y'(0)) = (0, 0)$ och tidsintervallet $0 \leq t \leq 30$.

Hela uppgiften ger maximalt **(8p)**

Uppgift 3. Betrakta randvärdesproblemet

$$\begin{cases} -u'' + 15x^2 u = \exp(-x^2),\ 0 < x < 1 \\ u'(0) + u(0) = 1,\ u(1) = 5 \end{cases}$$

(a). Gör en indelning av intervallet så att vi får n inre punkter. Skriv ned matrisekvationen vi får då vi i problemet ersätter derivator med differensapproximationer. Vi vill se matrisen och högerledsvektorn.

(b). Härled de differensapproximationer du använde i **(a)**. Det skall framgå vilken noggrannhetsordning approximationerna har.

(c). Skriv den kod i MATLAB som behövs för att bygga upp matris och högerled från **(a)**, samt beräkna lösningen och rita upp den. Matrisen skall lagras som en gles matris med `spdiags`. Även randvillkoren skall finnas med vid uppritningen. Tag $n = 30$.

Hela uppgiften ger maximalt **(8p)**

Fortsätter >>>

Uppgift 4. Betrakta följande linjära system av ODE

$$\begin{cases} \boldsymbol{u}' = \boldsymbol{A}\boldsymbol{u},\ t > 0 \\ \boldsymbol{u}(0) = \boldsymbol{u}_0 \end{cases}, \quad \boldsymbol{A} = \begin{bmatrix} 1 & -3 \\ -2 & 2 \end{bmatrix}, \quad \boldsymbol{u}_0 = \begin{bmatrix} 5 \\ 0 \end{bmatrix}$$

(a). Lös problemet med egenvärdesmetoden.

(b). Härled formeln du använde i **(a)** utgående från att $\boldsymbol{A}$ är diagonaliserbar ($\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V} = \boldsymbol{D}$).

(c). Tag ett steg med Euler framåt med steglängden $h = 0.1$.

Hela uppgiften ger maximalt **(8p)**

Uppgift 5. Betrakta det icke-linjära ekvationssystemet

$$\begin{cases} x_2 - x_1 x_2^2 = -1 \\ x_1^2 + 4x_2^2 - x_2 = 10 \end{cases}$$

(a). Skriv systemet på kompakt form $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$. Vad blir $\boldsymbol{f}(\boldsymbol{x})$ och vad blir $\boldsymbol{Df}(\boldsymbol{x})$, dvs. beräkna derivatamatrisen.

(b). Härled Newtons metod för ekvationen $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$.

(c). Tag ett steg med Newtons metod utgående från startapproximationen $\boldsymbol{x}_0 = (1, 1)$ för hand, dvs. utför beräkningarna med papper och penna.

(d). Skriv med den kod i MATLAB som utgående från startapproximationen $\boldsymbol{x}_0 = (1, 1)$ beräknar lösningen med Newtons metod.

Hela uppgiften ger maximalt **(8p)**

Uppgift 6. Betrakta följande värmeledningsproblem

$$\begin{cases} u'_t - \kappa u''_{xx} = C(u_{omg} - u) + f(x),\ 0 < x < L,\ 0 < t \leq T \\ u'_x(0, t) = 0,\ u(L, t) = g_L,\ t > 0 \\ u(x, 0) = u_{beg}(x),\ 0 < x < L \end{cases}$$

där $\kappa = 2$, $C = 3$, $u_{omg} = 15$, $f(x) = \exp(-(x - \frac{L}{2})^2)$, $L = 1$, $g_L = 60$ och $u_{beg}(x) = 15$.

(a). Vi skall lösa problemet med linjemetoden. Skriv ned det begynnelsevärdesproblem vi får då vi approximerar x-derivatorna med differenskvoter. Vi vill ha svaret på vektorform.

(b). Skriv ned den kod som i MATLAB behövs för att med `ode45` beräkna och sedan rita upp temperaturen under $T = 1$ sekunder. Tänk på att även värdena vid $x = 0$ och $x = L$ skall finnas med vid uppritningen. Tag $n = 30$.

Hela uppgiften ger maximalt **(10p)**

Differensapproximationer

$$D_+u(x) = \frac{u(x+h)-u(x)}{h} = u'(x) + \mathcal{O}(h), \quad D_-u(x) = \frac{u(x)-u(x-h)}{h} = u'(x) + \mathcal{O}(h)$$

$$D_0u(x) = \frac{u(x+h)-u(x-h)}{2h} = u'(x) + \mathcal{O}(h^2)$$

$$D_+D_-u(x) = \frac{u(x+h)-2u(x)+u(x-h)}{h^2} = u''(x) + \mathcal{O}(h^2)$$

$$\frac{u(x+2h)-2u(x+h)+2u(x-h)-u(x-2h)}{2h^3} = u'''(x) + \mathcal{O}(h^2)$$

$$\frac{u(x+2h)-4u(x+h)+6u(x)-4u(x-h)+u(x-2h)}{h^4} = u^{(4)}(x) + \mathcal{O}(h^2)$$

Taylorutveckling

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots$$

$$+ \frac{f^{(n)}(a)}{n!}(x-a)^n + R_{n+1}(x)$$

$$R_{n+1}(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-a)^{n+1} = \mathcal{O}((x-a)^{n+1})$$

Taylorutveckling i två variabler

$$f(x,y) = f(a,b) + f'_x(a,b)h + f'_y(a,b)k + \tfrac{1}{2}\big(f''_{xx}(a,b)h^2 + 2f''_{xy}(a,b)hk + f''_{yy}(a,b)k^2\big) + \cdots$$

där $h = x-a$ och $k = y-b$.

Lite stolpar i Matlab som stöd för minnet.

Elementära funktioner

```
abs(x), sqrt(x), exp(x), log(x), cos(x), sin(x), tan(x)
```

Matris- och vektorfunktioner

```
size(A), length(v)
sum(v), prod(v), max(v), min(v), sort(v), norm(v), dot(u,v)
```

Matrisuppbyggande funktioner

```
A=ones(m,n), A=zeros(m,n), A=eye(m,n), A=diag(d,k), A=spdiags(D,K,m,n)
```

Villkorssatser

```
if uttryck          if uttryck               if uttryck
    satser              satser                   satser
end                 else                     elseif uttryck
                        satser                   satser
                    end                      end
```

Repetitionssatser

```
while uttryck      for variabel=uttryck     for variabel=start:steg:slut
    satser             satser                   satser
end                end                      end
```

Egna funktioner

```
function ut=funktionsnamn(parametrar)       handtagsnamn=@(parametrar) sats
    satser
```

Beräkningsfunktioner

```
x=fzero(f,x0), x=fminbnd(f,x0,x1), q=integral(f,a,b), [t,U]=ode45(f,tspan,u0)
x=A\b, [V,D]=eig(A), [V,D]=eigs(A,K,'SM')
x=fsolve(f,x0), x=fminunc(f,x0), q=integral2(f,a,b,c,d)
```

Grafik

```
x=linspace(a,b,n)
plot(x,y), plot3(x,y,z), quiver(u,v,du,dv,sc), quiver3(...)
[X,Y]=meshgrid(x,y)
surf(X,Y,Z), contour(X,Y,Z,lv)
```