Tentamen för Matematisk överbryggningskurs

Tid och plats: 2017-03-17, 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. Betrakta följande begynnelsevärdesproblem

$$\begin{cases} u'' - \mu(1-u^2)u' + u = 0, \ 0 \le t \le T \\ u(0) = u_0, \ u'(0) = v_0 \end{cases}$$

(a). Skriv om ekvationen som ett första ordningens system.

(b). Skriv den kod i MATLAB som behövs för att lösa problemet med `ode45` och rita upp lösningen som funktion av tiden. Tag $\mu = 0.5$, $u_0 = -2$, $v_0 = 4$ och $T = 20$.

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

Uppgift 2. Betrakta egenvärdesproblemet

$$\begin{cases} -u'' + u = \lambda u, \ 0 < x < L \\ u(0) = 0, \ u'(L) + u(L) = 0 \end{cases}$$

(a). Gör en indelning av intervallet så att vi får n inre punkter. Skriv ned det egenvärdesproblem för matris vi får då vi i problemet ersätter derivator med differensapproximationer.

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

(c). Skriv ned den kod i MATLAB som behövs för att bygga upp matrisen samt lösa egenvärdesproblemet, komplettera lösningen med randvärden och rita upp. Matrisen skall lagras som en gles matris med `spdiags`. Vi skall beräkna de fem lösningarna med egenvärden närmast noll. Tag $L = 1$ och $n = 50$. Använd `eigs` för lösning av egenvärdesproblemet och `subplot` vid uppritning.

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

Uppgift 3. Antag att vi har en $n \times n$-matris $\boldsymbol{A}$ och att det för matrisens egenvärden λ_i gäller att $|\lambda_1| \le \cdots \le |\lambda_{n-1}| < |\lambda_n|$, samt att motsvarande egenvektorer $\boldsymbol{v}_i$, $i = 1, 2, \cdots, n$, är linjärt oberoende.

Givet en startvektor $\boldsymbol{u}_0$ visa att potensmetoden $\boldsymbol{u}_{k+1} = \boldsymbol{A}\boldsymbol{u}_k, \ k = 0, 1, \cdots$, konvergerar mot egenvektorn $\boldsymbol{v}_n$ sammanhörande med det till belopp största egenvärdet λ_n.

Uppgiften ger maximalt **(8p)**

Uppgift 4. Betrakta det icke-linjära ekvationssystemet

$$\begin{cases} x_1 \sin(x_1 + x_2) + x_1^2 + 2x_2 - 7 = 0 \\ \cos(x_1) + x_1 + 0.2x_1x_2 - 0.4 = 0 \end{cases}$$

(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 $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$.

Fortsätter >>>

(c). Tag ett steg med Newtons metod utgående från startapproximationen $\boldsymbol{x}_0 = (0, 0)$ för hand, dvs. utför beräkningarna med penna och papper.

(d). Skriv ned den kod i MATLAB som, utgående från startapproximationen $\boldsymbol{x}_0 = (0, 0)$, beräknar en noggrann lösning med Newtons metod.

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

Uppgift 5. Vi skall bestämma stationära punkter till funktionen

$$f(\boldsymbol{x}) = -x_1^3 + 5x_1x_2 - 3x_2^3 - 1$$

i området $-4 \leq x_1 \leq 4$, $-4 \leq x_2 \leq 4$.

(a). Beräkna gradienten $\boldsymbol{\nabla} f(\boldsymbol{x})$ och Hessematrisen $\boldsymbol{H}(\boldsymbol{x})$ för hand. Använd matris- och vektorbeteckningar.

(b). Skriv ned den kod i MATLAB som behövs för att lokalisera de stationära punkterna. Rita dels funktionsyta med `surf`, dels nivåkurvor med `contour`. Antag att vi genom att inspektera funktionsytan ser att de intressanta nivåerna ligger mellan -40 och 40. Använd detta vid uppritning av nivåkurvorna. Rita även i samma figur noll-nivåkurvorna till komponenterna i gradienten.

(c). Skriv ned den kod i MATLAB som, utgående från en startapproximation, beräknar stationär punkt med `fsolve` och sedan med `eig` avgör dess typ.

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

Uppgift 6. Följande partiella differentialekvation beskriver värmeutvecklingen $u(x, t)$ i en bromsskiva på bil vid kraftig inbromsning.

$$\begin{cases} u'_t - \kappa u''_{xx} = 0, \quad 0 < x < L, \ t > 0 \\ -Ku'_x(0, t) + H(u(0, t) - u_{omg}) = Q(t), \quad u'_x(L, t) = 0, \ t > 0 \\ u(x, 0) = u_{omg}, \quad 0 \leq x \leq L \end{cases}$$

där $\kappa = 8 \cdot 10^{-6}, L = 6 \cdot 10^{-3}, K = 100, H = 10^4, u_{omg} = 20$ och $Q(t) = (1 - t/3) \cdot 10^7$ är värden för en viss bil och vissa yttre betingelser.

(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 den kod i MATLAB som behövs för att beskriva högerledet i systemet av differentialekvationer från **(a)** som en funktionsfil.

(c). Skriv den kod i MATLAB som behövs för att med `ode45` beräkna och ritar upp temperaturen i bromsskivan under 3 sekunder.

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

Differensapproximationer

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

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

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

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

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

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)
```