Tentamen för Matematisk överbryggningskurs

Tid och plats: 2017-08-23, 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. Rörelsen hos en boll som kastas beskrivs av differentialekvationen

$$\begin{cases} mx'' = -cx'v \\ my'' = -mg - cy'v \end{cases}$$

där $v = \sqrt{(x')^2 + (y')^2}$ är bollens fart och m är dess massa. Vi har tagit med luftmotståndet men bortsett från rotation.

(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 bollens bana. Tag $m = 0.1$, $c = 0.01$, begynnelseläget $(x(0), y(0)) = (0, 1.8)$, begynnelsehastigheten $(x'(0), y'(0)) = (12, 10)$ och tidsintervallet $0 \leq t \leq 3$.

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

Uppgift 2. Betrakta egenvärdesproblemet

$$\begin{cases} -u'' + u = \lambda u, \ 0 < x < 1 \\ u(0) = 0, \ u(1) = 0 \end{cases}$$

(a). Gör en indelning av intervallet i $n + 1$ likformiga delintervall. Skriv ned det egenvärdesproblem för matris vi får då vi i problemet ersätter derivator med differensapproximationer. Vi vill se matrisen.

(b). Härled centraldifferensapproximationen av $u''(x)$. Det skall framgå vilken noggrannhetsordning den 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 tre lösningarna med egenvärden närmast noll. Använd `eigs`.

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

Uppgift 3. Visa att om $\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_r$ är egenvektorer till en matris $\boldsymbol{A}$ och sammanhörande egenvärden $\lambda_1, \lambda_2, \cdots, \lambda_r$ alla är olika så är $\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_r$ linjärt oberoende.

Uppgiften ger maximalt **(8p)**

Uppgift 4. Betrakta det icke-linjära ekvationssystemet

$$\begin{cases} x_1^2 + x_2 + (x_1 + 1)\exp(-x_2) - 5 = 0 \\ \exp(x_1 x_2) + x_1 + x_2 - 0.4 = 0 \end{cases}$$

(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.

Fortsätter >>>

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

(c). Tag ett steg med Newtons metod utgående från startapproximationen $\boldsymbol{x}_0 = (-1, 0)$ för hand, dvs. utför beräkningarna med papper och penna.

(d). Skriv med den kod i MATLAB som, utgående från en startapproximation, beräknar lösningen med Newtons metod.

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

Uppgift 5. Vi skall använda Steepest descentmetoden för att minimera

$$f(\boldsymbol{x}) = x_1^2 - x_1 + x_1 x_2 + x_2^2$$

(a). Utgå från approximationen $\boldsymbol{x}_k = (1, 1)$ och låt

$$\boldsymbol{x}(s) = \boldsymbol{x}_k - s\boldsymbol{\nabla} f(\boldsymbol{x}_k)$$

Bestäm $\hat{s} \geq 0$ som minimerar $g(s) = f(\boldsymbol{x}(s))$ och bilda

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \hat{s}\,\boldsymbol{\nabla} f(\boldsymbol{x}_k)$$

som vi tar som nästa approximation. Redovisa alla beräkningar.

(b). Visa att $\boldsymbol{\nabla} f(\boldsymbol{x}_{k+1}) \cdot \boldsymbol{\nabla} f(\boldsymbol{x}_k) = 0$. Du skall visa det allmänt, inte bara för vår funktion.

(c). Skriv med den kod i MATLAB som, utgående från en startapproximation, beräknar lösningen till minimeringsproblemet med Steepest descentmetoden. Använd `fminbnd` för att lösa delproblemen (minimeringen av $g(s)$).

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

Uppgift 6. Betrakta följande hyperboliska partiella differentialekvation

$$\begin{cases} u''_{tt} = u''_{xx}, \ 0 < x < 1, 0 \leq t \leq 5 \\ u(x,0) = \sin(\frac{\pi}{2}x), \ u'_t(x,0) = 0, \ 0 < x < 1 \\ u(0,t) = 0, \ u(1,t) = \cos(3t), \ 0 \leq t \leq 5 \end{cases}$$

(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 lösningen $u(x,t)$.

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)
```