Solution to the written exam

Tecnical University of Denmark

Written 4-hour exam, 19. Decemeber 2018 Page 1 of 5

Course name: Introduction to Numerical Algorithms Course nr.: 02601

Aids and materials: All aids and materials permitted.

Multiple choice (25%)

A – Gaussian elimination with partial pivoting. A3.

B – Interpolation. B4.

C – Convergence of secant method. C2.

D – Sensitivity analysis. D2: $\|\delta \boldsymbol{x}\|_2/\|\boldsymbol{x}\|_2 \leq \kappa(\boldsymbol{A})\|\delta \boldsymbol{b}\|/\|\boldsymbol{b}\|$.

E – Numerical integration. E3: $\frac{1}{180} \cdot 4 \cdot (\frac{4}{n})^4 \cdot (4+4)e^4 \leq 10^{-4}$ and n must be even number.

Question 1: Gaussian elimination with partial pivoting (10%)

1.1)

$$\begin{bmatrix} 2 & 4 & 5 \\ 3 & 6 & -3 \\ 1 & -2 & 8 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 13 \\ 9 \\ 4 \end{bmatrix} \qquad \begin{matrix} (1) \\ (2) \\ (3) \end{matrix}$$

Pivoting gives:

$$\begin{bmatrix} 3 & 6 & -3 \\ 2 & 4 & 5 \\ 1 & -2 & 8 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 9 \\ 13 \\ 4 \end{bmatrix} \qquad \begin{matrix} (2) \\ (1) \\ (3) \end{matrix}$$

Elemination of the (2,1) and (3,1) elements gives:

$$\begin{bmatrix} 3 & 6 & -3 \\ 0 & 0 & 7 \\ 0 & -4 & 9 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 9 \\ 7 \\ 1 \end{bmatrix} \qquad \begin{matrix} & (2) \\ (1') = (1) - \frac{2}{3} \cdot (2) \\ (3') = (3) - \frac{1}{3} \cdot (2) \end{matrix}$$

Pivoting gives:

$$\begin{bmatrix} 3 & 6 & -3 \\ 0 & -4 & 9 \\ 0 & 0 & 7 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 9 \\ 1 \\ 7 \end{bmatrix} \qquad \begin{matrix} (2) \\ (3') \\ (1') \end{matrix}$$

Now the matrix has upper triangular form.

1.2) Back substitution gives:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} .$$

Question 2. Data fitting (20%)

2.1) The Matlab code:

```
load data

m = length(xk);
A = [ ones(m,1), xk, cos(3*xk), exp(xk)];

At=A';
At*A  % system matrix
At*yk  % right-hand side
c=(At*A)\(At*yk) % solution

sum(abs(yk-A*c))

xx = linspace(0,5);
yy = c(1) + c(2)*xx + c(3)*cos(3*xx) + c(4)*exp(xx);

plot(xk,yk,'o',xx,yy,'-')
```

The system matrix $\boldsymbol{A}^T\boldsymbol{A}$ and the right-hand side $\boldsymbol{A}^T y$ of the normal equation are

$$\begin{bmatrix} 26 & 65 & 1.1713 & 814.23 \\ 65 & 221 & 2.3488 & 3359.1 \\ 1.1713 & 2.3488 & 12.928 & 25.245 \\ 814.23 & 3359.1 & 25.245 & 66810 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 509.77 \\ 1669.5 \\ 46.333 \\ 28265 \end{bmatrix}.$$

2.2) The solution is obtained by `c = (At'*A)\(At'*yk)`:

$$a = 10.04, \quad b = 0.08869, \quad c = 2.082, \quad d = 0.2955 \ .$$

2.3) The sum of the absolute errors is obtained by `e = sum(abs(A*c-yk))`: 18.351.

2.4) Since the system matrix is not changed when y changed, we can use factorization to speed up the computations. Without factorization, we need several times Gaussian elimination ($n^3/3$ LOps) with forward and back substitutions (n^2 LOps). But with factorization, we would only need once factorization ($n^3/3$ LOps for LU or $n^3/6$ LOps for Cholesky) with several times of forward and back substitutions (n^2 LOps).

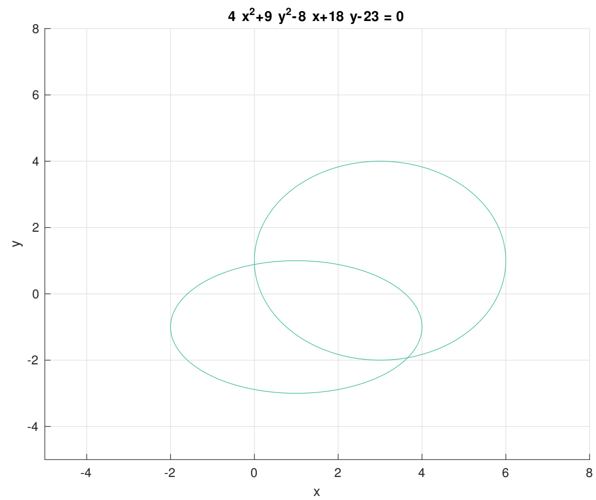
Question 3: Newton's method for a system of nonlinear equations (15%)

3.1)

```
function [F,dF] = funFdF(X)

F = [X(1)^2+X(2)^2-6*X(1)-2*X(2)+1;...
     4*X(1)^2+9*X(2)^2-8*X(1)+18*X(2)-23];

dF = [2*X(1)-6     2*X(2)-2;...
      8*X(1)-8  18*X(2)+18];
```



3.2) We can plot the figure of both equations $f_1(x,y)=0$ and $f_2(x,y)=0$: We can see that there are two intersections, i.e. two solutions for the system. $\boldsymbol{X}^{(0)}=[4;-2]$ can be a good starting point for the solution that is the most distant to the origin.

3.3) The final solution is [3.6584; -1.9269].
The Matlab code:

```
figure,hold on,grid on
ezplot('x^2+y^2-6*x-2*y+1',[-5,8]);
ezplot('4*x^2+9*y^2-8*x+18*y-23',[-5,8])
hold off

X0=[4;-2];
Xiterations = Newtonsys(@funFdF,X0,10);
Xiterations(:,end)
```

Question 4: Approximation of a function (10%)

4.1) Based on the second interpolation error theorem, we have

$$|f(x)-p_{16}(x)| \leq \frac{1}{4(n+1)}Mh^{n+1} = \frac{1}{4\times 17}\cdot 26e^6 \cdot \left(\frac{6-0}{16}\right)^{17} \approx 8.846\times 10^{-6}$$

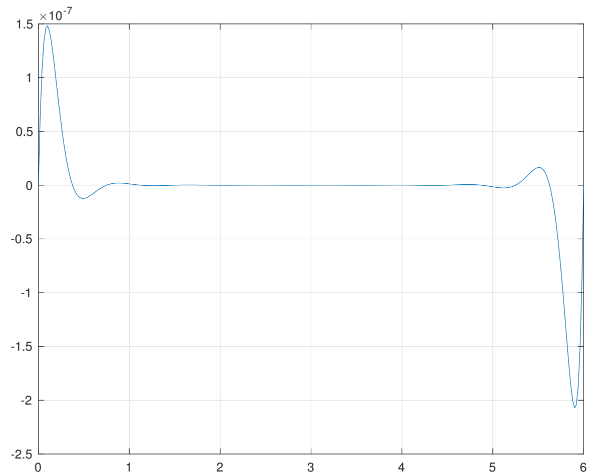
with M as the upper bound of $|f^{(n+1)}(x)|$.

4.2) The Matlab code:

```
close all, clear all
format compact, format long

x=linspace(0,6,300);
knuder=linspace(0,6,17);
f=(x+3).*exp(x);
fknuder=(knuder+3).*exp(knuder);
```

```
fejl=f-LagrangeFormInterpolation(knuder,fknuder,x);
figure, plot(x,fejl), grid on
```



Question 5: Initial-value problem (20%)

5.1) Set

$$Z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} x \\ y \\ x' \\ y' \end{bmatrix},$$

then we have

$$Z' = \begin{bmatrix} x' \\ y' \\ x'' \\ y'' \end{bmatrix} = \begin{bmatrix} z_3 \\ z_4 \\ z_1^2 - z_2 + e^t \\ z_1 - z_2^2 - e^t \end{bmatrix},$$

and the initial condition is

$$Z(0) = \begin{bmatrix} 0 \\ 1 \\ 0 \\ -2 \end{bmatrix}.$$

5.2) The Matlab code

```
function dzdt = odefun(t,Z)
dzdt=[Z(3,1);Z(4,1);Z(1,1)^2-Z(2,1)+exp(t);...
        Z(1,1)-Z(2,1)^2-exp(t)];
```

5.3) We have $n = 2^5 = 32$ and 64, respectively. In addition, $n_f = 2^7 = 128$ and 256, respectively.
The final results are $(10.08777, -21.06047)$ and $(10.08808, -21.06223)$.
The Matlab code:

```
tspan=[0 2];
Z0=[0;1;0;-2];
n1=32;
n2=64;

[tRK1,ZRK1] = MyRK4System(@odefun,tspan,Z0,n1);
[tRK2,ZRK2] = MyRK4System(@odefun,tspan,Z0,n2);

disp('5.3) Z(2) equals')
ZRK1(:,end)
ZRK2(:,end)
```

End