HACETTEPE UNIVERSITY
ENGINEERING FACULTY
ELECTRICAL AND ELECTRONICS ENGINEERING PROGRAM

2023-2024
SPRING SEMESTER

ELE708
NUMERICAL METHODS IN ELECTRICAL ENGINEERING

HW2

N23239410 – Ali Bölücü

1) Exercises

1.a) 2.11

2.11 Write out detailed algorithm for solving lower triangular linear system $Lx = b$ by forward substitution.

$$L = \begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ \vdots & & & \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{12} \\ \vdots \end{bmatrix} = \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \\ \vdots \end{bmatrix}$$

- for $j = 1$ to n
  - if $l_{jj} = 0$ then stop
  - $x_j = b_j / l_{jj}$
  - for $i = j+1$ to n
    - $b_i = b_i - l_{ij} \cdot x_j$
  - end
- end

⇓

- loop over columns
- stop if pivot is zero
- compute x
- substruct found x value from next rows

- for $j = 1$ to n
  - if $l_{jj} = 0$, then stop
  - for $i = 1$ to j
    - $b_j = b_j - x_i \cdot l_{ji}$
  - end
  - $x_j = b_j / l_{jj}$
- end

⇓

- loop over colums
- stop if pivot is zer
- substruct previously founded x value from b
- calculate x

1.b) 2.17

2.17. Write out the LU factorization of the following matrix

$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \; S_1 + S_2 \rightarrow S_2 \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix} \; S_2 + S_3 \rightarrow S_3 \underbrace{\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}}_{U}$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{M_1} \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}}_{M_2}$$

$$L = M_1^{-1} \cdot M_2^{-1} \Rightarrow \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}}_{L}$$

1.c) 2.31

2.31. Let A be a symmetric positive matrix.

$$||x||_A = (x^T . Ax)^{1/2}$$

satisfies the three properties of a vector norm.

1. $||x|| > 0$ if $x \neq 0$
2. $||yx|| = |y| . ||x||$ for any scalar y
3. $||x+y|| \leq ||x|| + ||y||$

1. A is positive definite so $\Rightarrow x^T A.x \geq 0$ — So its true

but $x^T.A.x = 0$ if $x = 0$ $\rightarrow x^T.A.x > 0$ if $x \neq 0$

---

2. $||yx|| = (yx^T A yx)^{1/2} = (y^2 . x^T . A.x)^{1/2}$

$= |y| (x^T A.x)^{1/2}$

$= |y| . ||x||$ $\rightarrow$ its true

---

3. $||x+y||^2 = (x+y)^T . A . (x+y)$

$= (x^T . A.x) + (x^T.A.y) + (y^T.A.x) + (y^T A.y)$

$= ||x||^2 + ||y||^2 + (x^T.A.y) + (\underline{y^T.A.x})$

$= ||x||^2 + ||y||^2 + 2 . x^T.A.y$

$\leq ||x||^2 + ||y||^2 + 2 . (x^T.Ax) . (y^T.A.y)$

$\leq ||x||^2 + ||y||^2 + 2 . ||x|| . ||w||$ — Cauchy - Schwarz

$||x+y||^2 \leq (||x|| + ||y||)^2$

$||x+y|| \leq ||x|| + ||y||$

$x^T A.y = (x^T A.y)^T$

$= y^T . A^T . x$

$= y^T . A . x$

$\Downarrow$

$x^T A.y = y^T . A . x$

1.d) 2.32

2.32. Show that the following functions of an $m \times n$ matrix A satisfy the matrix norm properties

a-) $\|A\|_{max} = \max_{i,j} |a_{ij}|$

b-) $\|A\|_F = \left(\sum_{i,j} |a_{ij}|^2\right)^{1/2}$

1. $\|A\| > 0$ if $A \neq 0$
2. $\|y A\| = |y| \cdot \|A\|$
3. $\|A+B\| \leq \|A\| + \|B\|$

a 1-) if $|a_{ij}| \neq 0$, $|a_{ij}| > 0$ $\Rightarrow$ therefore $\|A\|_{max} > 0$

a.2-) $\|y.A\|_{max} = \max_{ij} |y. a_{ij}|$ $\Rightarrow$ $\|y.A_{max}\| = |y|.\|A\|$

$= |y| . \max |a_{ij}|$

a 3-) $\|A+B\|_{max} = \max_{ij} |a_{ij} + b_{ij}|$

$\leq \max(|a_{jj}| + |b_{jj}|)$ $\Rightarrow$ $\|A+B\|_{max} \leq \|A\| + \|B\|$

$\leq \max |a_{ij}| + \max |b_{ij}|$

---

b 1-) $|a_{ij}|^2 > 0$ if $a_{ij} \neq 0$

Therefore $\left(\sum_{ij} |a_{ij}|^2\right) > 0$ $\rangle$ $\|A\|_F > 0$

b2-) $\|y A\|_F = \left(\sum |a_{ij}.y|^2\right)^{1/2}$

$= \left(\sum |y|^2 . |a_{ij}|^2\right)^{1/2} = |y| . \left(\sum |a_{ij}|^2\right)^{1/2}$ $\rangle$ $\|y.A\| = |y|.\|A\|$

b3-) $\|A+B\|^2 = \left(\sum_i |A+B|^2\right)$

$= \sum_i |A|^2 + 2.\sum_i |A.B| + \sum |B|^2$

$\leq \sum_i |a_i|^2 + 2.\left(\sum_i |a_i|^2\right)^{1/2} . \left(\sum |b_i|^2\right)^{1/2} + \sum |b_i|^2$

$\leq \left(\left(\sum |a_i|^2\right)^{1/2} + \left(\sum |b_i|^2\right)^{1/2}\right)^2$

$\leq (\|A\| + \|B\|)^2$

$\|A+B\|^2 \leq \|A\| + \|B\|$

2) Computer Problems

2.a) 2.3 Resolving the member forces into horizontal and vertical components and defining α =√ 2/2, we obtain the following system of equations for the member forces fi:

$$
\begin{aligned}
&\text{Joint } 2: \begin{cases} f_2 = f_6 \\ f_3 = 10 \end{cases} \\
&\text{Joint } 3: \begin{cases} \alpha f_1 = f_4 + \alpha f_5 \\ \alpha f_1 + f_3 + \alpha f_5 = 0 \end{cases} \\
&\text{Joint } 4: \begin{cases} f_4 = f_8 \\ f_7 = 0 \end{cases} \\
&\text{Joint } 5: \begin{cases} \alpha f_5 + f_6 = \alpha f_9 + f_{10} \\ \alpha f_5 + f_7 + \alpha f_9 = 15 \end{cases} \\
&\text{Joint } 6: \begin{cases} f_{10} = f_{13} \\ f_{11} = 20 \end{cases} \\
&\text{Joint } 7: \begin{cases} f_8 + \alpha f_9 = \alpha f_{12} \\ \alpha f_9 + f_{11} + \alpha f_{12} = 0 \end{cases} \\
&\text{Joint } 8: \{\ f_{13} + \alpha f_{12} = 0
\end{aligned}
$$

Solve the linear system.

When we create the matrix according to the giving equations, and solve. We get following answer.

```
Workspace
>> CP2_3
Force  1: -28.284271
Force  2: 20.000000
Force  3: 10.000000
Force  4: -30.000000
Force  5: 14.142136
Force  6: 20.000000
Force  7: 0.000000
Force  8: -30.000000
Force  9: 7.071068
Force 10: 25.000000
Force 11: 20.000000
Force 12: -35.355339
Force 13: 25.000000
fx >>
```

2.b) 2.4 Write a routine for estimating the condition number of a matrix A. You may use either the 1-norm or the ∞-norm (or try both and compare the results). You will need to compute ||A||, which is easy, and estimate ||A^−1||, which is more challenging. As discussed in Section 2.3.3, one way to estimate ||A^−1|| is to choose a vector y such that the ratio ||z||/||y|| is large, where z is the solution to Az = y. Try two different approaches to choosing y:

Method-a gives better results compared to method-b but calculating b is more practical.

Matrix-2 is more ill-conditioned compared to Matrix-1.

Workspace | Command Window

```
>> CP2_4
-----------------------------------
Matrix 1, 1-norm
Condition estimate, method a: 1.120708e+01
Condition estimate, method b: 7.985759e+00
MATLAB condest function: 1.277419e+01
MATLAB cond function: 1.277419e+01
Actual condition number: 1.277419e+01
-----------------------------------

-----------------------------------
Matrix 1, infinity-norm
Condition estimate, method a: 1.457713e+01
Condition estimate, method b: 3.994167e+00
MATLAB condest function: 1.277419e+01
MATLAB cond function: 1.700000e+01
Actual condition number: 1.700000e+01
-----------------------------------

-----------------------------------
Matrix 2, 1-norm
Condition estimate, method a: 3.205084e+06
Condition estimate, method b: 2.575753e+06
MATLAB condest function: 4.016285e+06
MATLAB cond function: 4.016285e+06
Actual condition number: 4.016285e+06
-----------------------------------

-----------------------------------
Matrix 2, infinity-norm
Condition estimate, method a: 3.536601e+06
Condition estimate, method b: 1.477306e+06
MATLAB condest function: 4.016285e+06
MATLAB cond function: 4.431711e+06
Actual condition number: 4.431711e+06
-----------------------------------

fx >>
```

2.c) Codes for 2.3 and 2.4

2.c.i) 2.3

Editor - D:\Ders\M.Sc\M.Sc\ELE708 - Numerical Methods in Electrical Engineering\HW2\CP2_3.m

CP2_3.m | CP2_4.m | +

```matlab
A = zeros(13,13);
alpha = 1/sqrt(2);

% Joint 2
A(1,2) = 1;
A(1,6) = -1;
A(2,3) = 1;
b(2) = 10;

% Joint 3
A(3,1) = alpha;
A(3,4) = -1;
A(3,5) = -alpha;
A(4,1) = alpha;
A(4,3) = 1;
A(4,5) = alpha;

% Joint 4
A(5,4) = 1;
A(5,8) = -1;
A(6,7) = 1;

% Joint 5
A(7,5) = alpha;
A(7,6) = 1;
A(7,9) = -alpha;
A(7,10) = -1;
A(8,5) = alpha;
A(8,7) = 1;
A(8,9) = alpha;


% Joint 6
A(9,10) = 1;
A(9,13) = -1;
A(10,11) = 1;

% Joint 7
A(11,8) = 1;
A(11,9) = alpha;
A(11,12) = -alpha;
A(12,9) = alpha;
A(12,11) = 1;
A(12,12) = alpha;

% Joint 8
A(13,13) = 1;
A(13,12) = alpha;

b = [0;10;0;0;0;0;0;15;0;20;0;0;0;];

f = A\b;

fprintf('Force %2d: %f \n', [1:13;f'])

```

2.c.ii) 2.4

Editor - D:\Ders\M.Sc\M.Sc\ELE708 - Numerical Methods in Electrical Engineering\HW2\CP2_4.m

CP2_3.m | CP2_4.m

```matlab

A1 = [10 -7  0;
      -3  2  6 ;
       5 -1  5 ];

A2 = [-73 78 24;
       92 66 25;
      -80 37 10];
fprintf('-----------------------------------\n');
disp('Matrix 1, 1-norm');
estcond(A1, 1);
disp(' ');

fprintf('-----------------------------------\n');
disp('Matrix 1, infinity-norm');
estcond(A1, inf);
disp(' ');

fprintf('-----------------------------------\n');
disp('Matrix 2, 1-norm');
estcond(A2, 1);
disp(' ')

fprintf('-----------------------------------\n');
disp('Matrix 2, infinity-norm');
estcond(A2, inf);
disp(' ');

function [c] = estcond(A, p)
norm_A = norm(A, p);
[L,U,P] = lu(A);
n = size(A,1);

% part (a), transposed triangular solves with special rhs
v = zeros(n,1);
v(1) = 1/U(1,1);
for i=2:n
    tot = 0;
    for j= 1:i-1;
        tot = tot-U(j,i)*v(j);
    end

    if tot > 0,
        tot = tot+1;
    else
        tot = tot-1;
    end

    v(i) = tot/U(i,i);
end

for i= n:-1:1
    tot = v(i);
    for j=i+1:n
        tot = tot-L(j,i)*v(j);
    end
    v(i) = tot;
end
y = P*v; z = U\(L\(P*y));
fprintf('Condition estimate, method a: %e\n', norm_A*norm(z,p)/norm(y,p));


% part (b), several random choices for y
maxratio = 0;
for k=1:5
    y = rand(n,1); z = A\y; t = norm(z,p)/norm(y,p);
    if t > maxratio, maxratio = t;
    end
end


fprintf('Condition estimate, method b: %e\n', norm_A*maxratio);
fprintf('MATLAB condest function: %e\n', condest(A));
fprintf('MATLAB cond function: %e\n', cond(A,p));
fprintf('Actual condition number: %e\n', norm(A,p)*norm(inv(A),p))
fprintf('-----------------------------------\n');
end
```