TOPIC 4
Computer application for solving systems of linear equations

Systems of Linear Equations

Mathematical formulations of engineering problems often lead to sets of simultaneous linear equations. Such is the case, for instance, when using the finite element method (FEM).
A general system of linear equations can be expressed in terms of a coefficient matrix A, a right-hand-side (column) vector b and an unknown (column) vector x as
Ax = b
or, component wise, as

$$\begin{aligned} a_{1,1}x_1 + a_{1,2}x_2 + \cdots a_{1,n}x_n &= b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots a_{2,n}x_n &= b_2 \\ &\vdots \\ a_{n,1}x_1 + a_{n,2}x_2 + \cdots a_{n,n}x_n &= b_n \end{aligned}$$

MATLAB has several tool needed for computing a solution of the system of linear equations. Let **A** be an m-by-n matrix and let **b** be an m-dimensional (column) vector. To solve the linear system **Ax = b** one can use the *backslash operator* \ , which is also called the *left division*

1. Case m = n

In this case MATLAB calculates the exact solution (modulo the round off errors) to the system in question.
Let

```
A = [1 2 3;4 5 6;7 8 10]
A =
    1    2    3
    4    5    6
    7    8   10
```

and let `b = ones(3,1);`
Then

```
x = A\b
x =
-1.0000
1.0000
0.0000
```

In order to verify correctness of the computed solution let us compute the *residual vector* **r**

```
r = b - A*x
r =
1.0e-015 *
0.1110
0.6661
0.2220
```

Entries of the computed residual **r** theoretically should all be equal to zero. This example illustrates an effect of the round off errors on the computed solution.

2. Case m > n

If **m** > **n**, then the system **Ax** = **b** is *overdetermined* and in most cases system is inconsistent.
A solution to the system **Ax = b**, obtained with the aid of the backslash operator \ , is the *least squares solution*.
Let now

```
A = [2 -1; 1 10; 1 2];
```

and let the vector of the right-hand sides will be the same as the one in the last example. Then

```
x = A\b
x =
0.5849
0.0491
```

The residual **r** of the computed solution is equal to

```
r = b - A*x
r =
-0.1208
-0.0755
0.3170
```

Theoretically the residual **r** is orthogonal to the *column space* of **A**. We have

```
r'*A
ans =
1.0e-014 *
0.1110
0.6994
```

3. Case m < n

If the number of unknowns exceeds the number of equations, then the linear system is *underdetermined*. In this case MATLAB computes a *particular solution* provided the system is consistent. Let now

```
A = [1 2 3; 4 5 6]; b = ones(2,1);
```

Then

```
x = A\b
x =
-0.5000
0
0.5000
```

A *general solution* to the given system is obtained by forming a linear combination of **x** with the columns of the *null space* of **A**. The latter is computed using MATLAB function **null**

```
z = null(A)
z =
0.4082
-0.8165
0.4082
```

Suppose that one wants to compute a solution being a linear combination of **x** and **z**, with coefficients **1** and **–1**. Using function **lincomb** we obtain:

```
w = lincomb({1,-1},{x,z})
w =
-0.9082
0.8165
0.0918
```

The residual **r** is calculated in a usual way `r = b - A*w`

```
r =
1.0e-015 *
-0.4441
0.1110
```

Exercises:

Find the solution of SLE below:

1. $3x_1 - x_2 + 2x_3 = 10$

$3x_2 - x_3 = 15$

$2x_1 + x_2 - 2x_3 = 0$

2. $-1x + 7y + 5z = 12$

$6x + 3y - 2z = 3$

$8x + z = 10$

$4x - 4y + 2z = -9$

$$F=\begin{bmatrix} -2 & 1 & 5 \\ 0 & 3 & -1 \\ 8 & 2 & 0 \end{bmatrix} \qquad G=\begin{bmatrix} 1 & 3 & -1 \\ 4 & 2 & -2 \\ 5 & 1 & 4 \end{bmatrix} \qquad H=\begin{bmatrix} 0 & -3 & 1 \\ 2 & 1 & -4 \\ 4 & 0 & 1 \\ 2 & 5 & 3 \end{bmatrix}$$

Find the solution of SLE below:

a.Fx=G(:,1) case 1

b.Gx=F(:,1) case 1

c.Hx=H(:,2) case 2(m>n)

d.Fx=G(:,2) case 1

e.Gx=F(:,2) case 1

f.H'(x)=G(:,2) case 3

g. Fx=G(:,3) case 1

h. Gx=F(:,3) case 1 (m=n)

i. H'(x)=F(3,:)' case 3(m<n)

b. x=1 0.8387
4 -0.5806
5 1.0968
r= 1.0e*-015
0.441