Topic 3

Matrix Operations

Matrix & Linear Algebra Operations

Element-by-Element(array) Operations

Introduction

- Matlab is designed to carry out advanced array operations that have many applications in science and engineering.
- We have seen scalars operate just like a number.
- Vectors and matrices mathematical operations are more complex.
- We begin with basic mathematics operations of **matrix** and **linear algebra**.

Matrix & Linear Algebra Operation

Add

Subtract

Multiplication

Division

Inverse

Addition & Subtraction

- To add/subtract, arrays must have identical size.
- A scalar can be added to an array.

```
>> va=[3 8 6];vb=[6 3 5];
>> vc=va-vb
vc =
    -3     5     1
```

Both are vectors

```
>> vd=10+vc
vd =
     7    15    11
```

Scalar & vector

Multiplication

- If **A** & **B** are arrays, * is executed according to the ***rules of linear algebra***.
- **A * B** - no. of columns **A** must be equal to no. of rows **B**
- **A*B ≠ B * A** - not commutative

```
>> a=[ 3 6 8;5 2 4;2 7 5];
>> b=[4 3 6 3;1 6 3 9;4 7 9 8];
>> f=a*b
f =
    50   101   108   127
    38    55    72    65
    35    83    78   109
```

3 x 3 matrix

3 x 4 matrix

3 x 4 matrix

```
>>g=b*a?
```

Multiplication - cont'd

```
>> a=[3 6 7];
>> b=[1;5;3];
>> m=a*b
m =
     54
```

Row vector

Column vector

Scalar

Dot product
of 2 vectors

```
>> n=b*a

n =
      3      6      7
     15     30     35
      9     18     21
```

3 x 3 matrix

```
>> z=3*n

z =
      9     18     21
     45     90    105
     27     54     63
```

Scalar * matrix

Division

- Division also is executed according to the rules of linear algebra
- **Identity matrix** - **`eye`** command
- **Inverse matrix** - **`inv( )`** function, or **`^-1`**

```
>> a=[7 4 2;5 2 7;9 5 7];
>> b=inv(a)
b =
    1.0000    0.8571   -1.1429
   -1.3333   -1.4762    1.8571
   -0.3333   -0.0476    0.2857
```

Square matrix

a*b=?

A matrix has an inverse only if it is square and its determinant is not zero

Use `det` command to calculate the determinant of a matrix (square)

Division – cont'd

```
>> a*b
ans =
    1.0000    0.0000   -0.0000
   -0.0000    1.0000         0
   -0.0000    0.0000    1.0000
```

b is the inverse of **a**

```
>> a*a^-1
ans =

    1.0000    0.0000   -0.0000
   -0.0000    1.0000         0
   -0.0000    0.0000    1.0000
```

Both give the same results

Division – cont'd

Two types of array division:

- **Left division, **

  Use to solve matrix equation **AX**=**B**, where **X** and **B** are **column** vectors.

  The solution for **AX**=**B** is $X=A^{-1}B$

In Matlab it can be done by;

i) inverse function $X=A^{-1}B$,or

ii) Left division **X=A\B**

Uses inverse

Uses Gauss elimination

Both of the above give the same result, but for large matrices the \ is more accurate

Division – cont'd

- **Right division, /**

Use to solve matrix equation **XC**=**D**, where **X** and **D** are **row** vectors.

> In Matlab it can be done by
>
> right division **X=D/C**

Division Example

- Solving linear equations

$$3x + 6y + 2z = 0$$
$$4x - 3y + 5z = 9$$
$$2x + 7y + 2z = 4$$

$$\begin{vmatrix} 3 & 6 & 2 \\ 4 & -3 & 5 \\ 2 & 7 & 2 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \end{vmatrix} = \begin{vmatrix} 0 \\ 9 \\ 4 \end{vmatrix}$$

$$\begin{vmatrix} x & y & z \end{vmatrix} \begin{vmatrix} 3 & 4 & 2 \\ 6 & -3 & 7 \\ 2 & 5 & 2 \end{vmatrix} = \begin{vmatrix} 0 & 9 & 4 \end{vmatrix}$$

AX=**B** form

XC=**D** form

Division Example - cont'd

AX=B form

X = A\B

OR

$X=A^{-1}B$

```
>> a=[3 6 2;4 -3 5;2 7 2];
>> b=[0;9;4];
>> v1=a\b
v1 =
    -3.7674
     0.2326
     4.9535
```

x
y
z

```
>> v2=inv(a)*b
v2 =
    -3.7674
     0.2326
     4.9535
```

```
>> v3=a^-1*b ?
```

Division Example – cont'd

XC=D form

X = D/C

```
>> c=[3 4 2;6 -3 7;2 5 2];
>> d=[0 9 4];
>> v4=d/c
v4 =
    -3.7674     0.2326     4.9535
```

x y z

OR

X=DC^{-1}

```
>> v5=d*inv(c)
v5 =
  -3.7674     0.2326     4.9535
```

```
>> x6=d*c^-1 ?
```

Array Operations

(element-by-element operations)

Perform

element-to-element

exponential, multiplication & division

on vectors and matrices

Element-by-Element Operations

When **multiplication** and **division** symbols are used with arrays, the mathematical operations follow the rules of **linear algebra**.

In other situations ***element-by-element*** operations are required. i.e. operations are carried out on each element of an array(s).

Element-by-element array can be done **only with arrays of the same size**.

(By definition addition and subtraction are already element-by-element operation)

Element-by-Element Operations - cont'd

Element-by-element multiplication, division and exponentiation of 2 arrays is entered in Matlab by typing a **period** in front of the arithmetic operator.

.*	**.**	**./**	**.^**

Element-by-element calculation are very useful for calculating the values of a function at many values.

Element-by-Element Operations - cont'd

```
>> A=[2 9 7;8 6 3];
>> B=[8 5 6;5 7 9];
>> C=A.*B
C =
    16    45    42
    40    42    27
```

Element-by-element multiplication

```
>> A*B ?
>> A*B' ?
>> A'*B?
>> A'*B' ?
```

```
>> B./A
ans =
    4.0000    0.5556    0.8571
    0.6250    1.1667    3.0000
```

Element-by-element division

```
>> A3=A.^3
A3 =
     8   729   343
   512   216    27
```

Element-by-element exponentiation

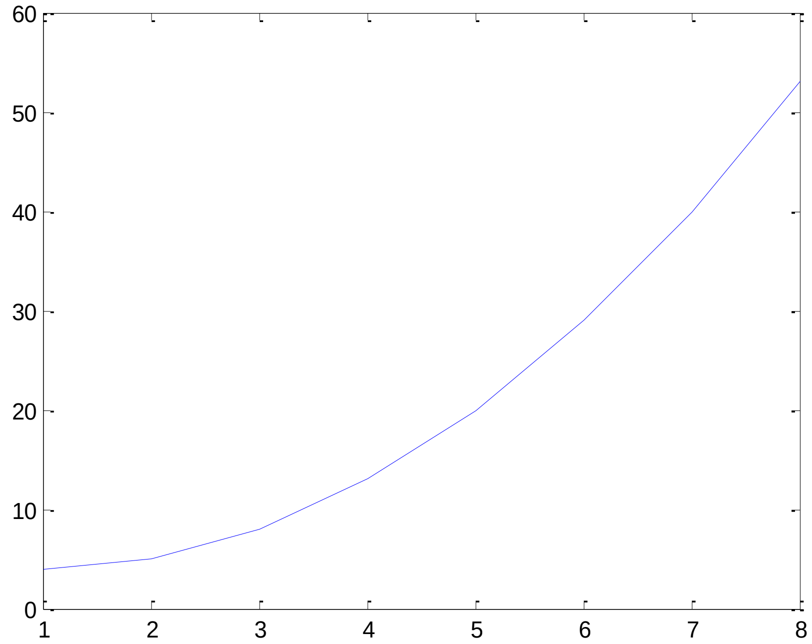
```
>> A^3 ?
```

Element-by-Element Operations - Example

Determine y for the expression, $y = x^2 - 2x + 5$ when x = 1, 2, 3, ...8.

```
>> x=[1:8];
>> y=x.^2-2.*x+5
y =
      4     5     8    13    20    29    40    53
```

x-y plot



Try This !

1. Define the following vectors:

   u = [4 -2 3] ***v*** = [-2 5 1]

   What will be displayed if the following commands are executed.

   a) ***u***.****v*** b) ***u****v***′ c) ***u***′****v***

2. Two vectors are given:

   u = -2**i** +8**j** -3**k** and ***v*** = 7**i** -5**j** – 4**k**

   Calculate the dot product of the vectors in three ways:

   (*a*) Write an expression using element-by-element calculation and the MATLAB built-in function sum.

   (*b*) Define **u** as a row vector and **v** as a column vector, and then use matrix multiplication.

   (c) Use the MATLAB built-in function dot.

3. Determine value of ***y*** for the expression, $y = \dfrac{x^3 + 5x}{4x^2 - 7}$ when ***x*** = 1 , 3, 5, ……..9.

Arrays in Built-in Math Functions

When an array is passed as the argument of a function, it is executed on each element of the array.

Like element-by-element operation.

```
>> x=[0:pi/5:pi]
x =
     0   0.6283   1.2566   1.8850   2.5133   3.1416

>> y=sin(x)
y =
     0   0.5878   0.9511   0.9511   0.5878   0.0000
```

Radian angle

```
>> x1=[0:36:180];
>> y1=sind(x1)
y1 =
     0   0.5878   0.9511   0.9511   0.5878   0
```

Degree angle

Arrays in Built-in Math Functions - cont'd

```
>> m=[1 5 8;45 65 130]
m =
     1     5     8
    45    65   130

>> r=sqrt(m)
r =
    1.0000    2.2361    2.8284
    6.7082    8.0623   11.4018
```

Problem Examples

Problem Example 1

A simply supported beam carries a udl along the whole span. Determine the bending moment at every metre of the beam.

$$r_a = ql/2$$

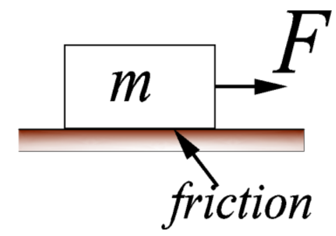
$$m_x = r_a\,x - qx^2/2$$

```
>> q=70;
>> l=6;
>> ra=q*l/2;
>> x=[0:6];
>> mx=ra.*x-q.*x.^2/2
```

```
ra =
   210

x =
     0   1   2   3   4   5   6

mx =
     0 175 280 315 280 175   0
```

Problem Example 2

The coefficient of friction, c, is determined experimentally by measuring force F required to move mass m. The result of the experiment is given below. Determine m for each test, and the average of all the tests.



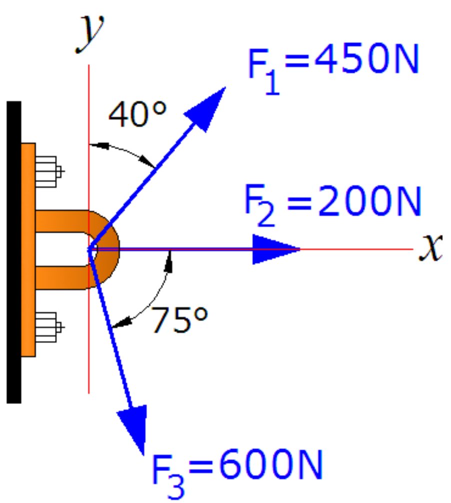
Test	1	2	4	5	6	7
m (kg)	2	4	7	10	15	25
F (N)	12.9	24.1	43.1	61.1	90	152

$\mu = F/mg$

```
>> m=[2 4 7 10 15 25];
>> F=[12.9 24.1 43.1 61.1 90 152];
>> mu=F./(m.*9.81)
mu =
    0.6575   0.6142   0.6276   0.6228   0.6116   0.6198
>> aveMu=mean(mu)
aveMu =
    0.6256
```

Problem Example 3

Determine the resultant of the forces acting on the bracket shown below.



$\mathbf{F} = F_x\mathbf{i} + F_y\mathbf{j} = F(\cos\theta\,\mathbf{i} + \sin\theta\,\mathbf{j})$

$F = S(F_x^2 + F_y^2)$

$\tan\theta = F_y / F_x$

```
>> f1=450*[cosd(50)  sind(50)];
>> f2=200*[cosd(0) sind(0)];
>> f3=600*[cosd(-75) sind(-75)];
>> ft=f1+f2+f3;
>> fr=sqrt(ft(1)^2 + ft(2)^2);
>> th=atand(ft(2)/ft(1));
```

Two element vector:
*1st element **x** compt.*
*2nd element **y** compt.*

Two element vector

scalar

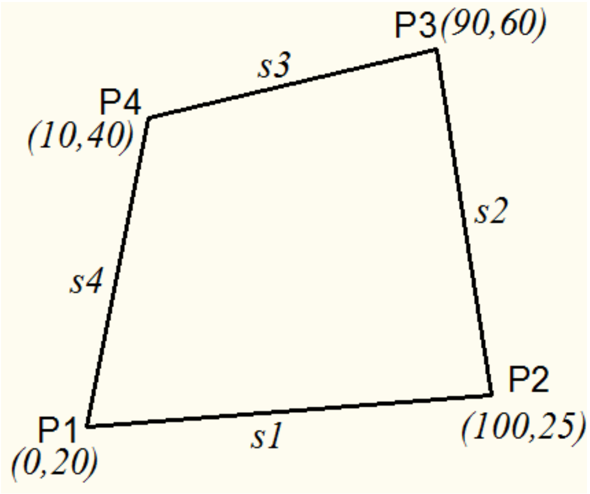
Problem Example 3 - cont'd

```
>> f1=450*[cosd(50)  sind(50)];
>> f2=200*[cosd(0) sind(0)];
>> f3=600*[cosd(-75) sind(-75)];
>> ft=f1+f2+f3;
>> fr=sqrt(ft(1)^2 + ft(2)^2);
>> th=atand(ft(2)/ft(1));
```

```
f1 =
  289.2544  344.7200

f2 =
   200      0

f3 =
  155.2914 -579.5555

ft =
  644.5459 -234.8355

fr =
  685.9935

th =
  -20.0188
```

Problem Example 4

Determine the length of the sides of the polygon shown.



$$S_{mn} = S[(X_m - X_n)^2 + (Y_m - Y_n)^2]$$

scalar

xa *xb* *ya* *yb*

```
S1= S[(x1-x2)² + (y1-y2)²]
S2= S[(x2-x3)² + (y2-y3)²]
S3= S[(x3-x4)² + (y3-y4)²]
S4= S[(x4-x1)² + (y4-y1)²]
```

vector

```
>> xa=[0 100 90 10];
>> xb=[xa(2:end) xa(1)];
>> ya=[20 25 60 40];
>> yb=[ya(2:end) ya(1)];
>> s=sqrt((xa-xb).^2+(ya-yb).^2)
s =
  100.1249   36.4005   82.4621   22.3607
```

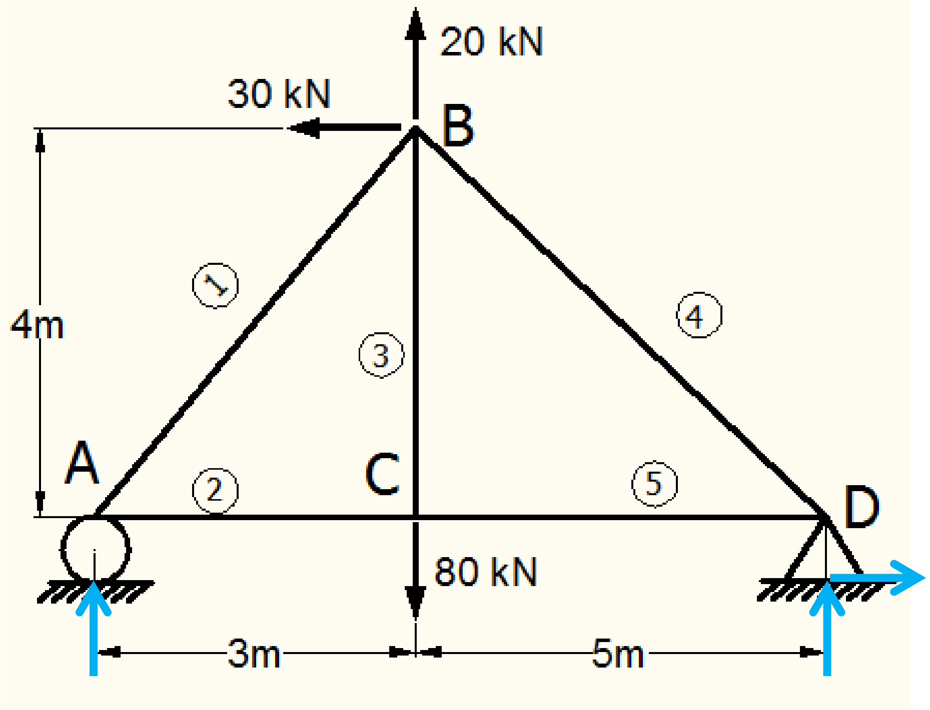
Problem Example 5

For the function $y = \frac{x^2 - 3}{x + 4}$ calculate the value of *y* for the following values of *x* using element-by-element operation: -2, -1, 0, 1, 2, 3.

```
>> x =[-2:3];
>> y=(x.^2-3)./(x+4)
y =
    0.5000   -0.6667   -0.7500   -0.4000    0.1667   0.8571
```

Problem Example 6

Determine the forces in all members of the truss loaded as shown.



1) Determine the support reaction components.
2) Write equilibrium equation for the joints [$\sum Fx = 0, \sum Fy = 0$].
3) Put the equations in matrix form.
4) Solve the simultaneous equations using **AX=B.**

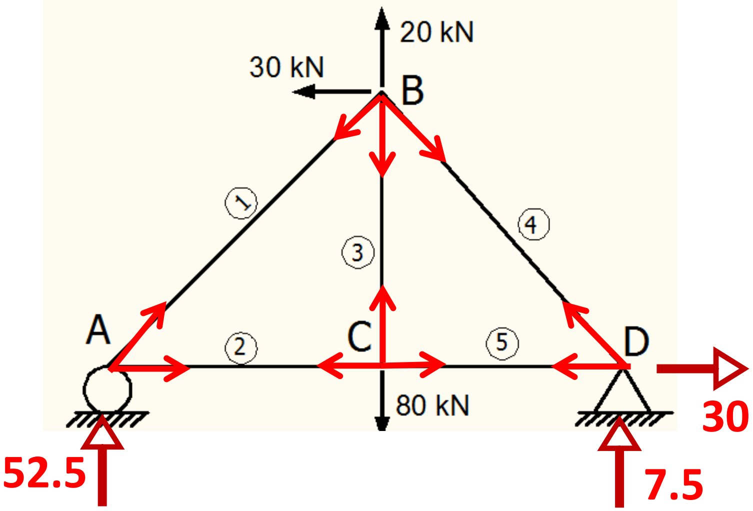
Using equilibrium eqns.;

V_A = 52.5 kN

V_D = 7.5 kN

H_D = 30 kN

Problem Example 6 (cont'd)



Joint A: $F_1 \cos 53.13 + F_2 = 0$

$F_1 \sin 53.13 = - 52.5$

Joint B: $-F_1 \cos 53.13 + F_2 \cos 38.66 = 30$

$-F_1 \sin 53.13 – F_3 \; -F_4 \sin 38.66 = -20$

Joint C: $- F_1 + F_5 = 0$

Putting the eqns. in matrix form

$$\underbrace{\begin{vmatrix} \cos 53.13 & 1 & 0 & 0 & 0 \\ \sin 53.13 & 0 & 0 & 0 & 0 \\ -\cos 53.13 & 0 & 0 & \cos 38.66 & 0 \\ -\sin 53.13 & 0 & -1 & -\sin 38.66 & 0 \\ 0 & -1 & 0 & 0 & 1 \end{vmatrix}}_{C} \underbrace{\begin{vmatrix} F1 \\ F2 \\ F3 \\ F4 \\ F5 \end{vmatrix}}_{F} = \underbrace{\begin{vmatrix} 0 \\ -52.5 \\ 30 \\ -20 \\ 0 \end{vmatrix}}_{P}$$

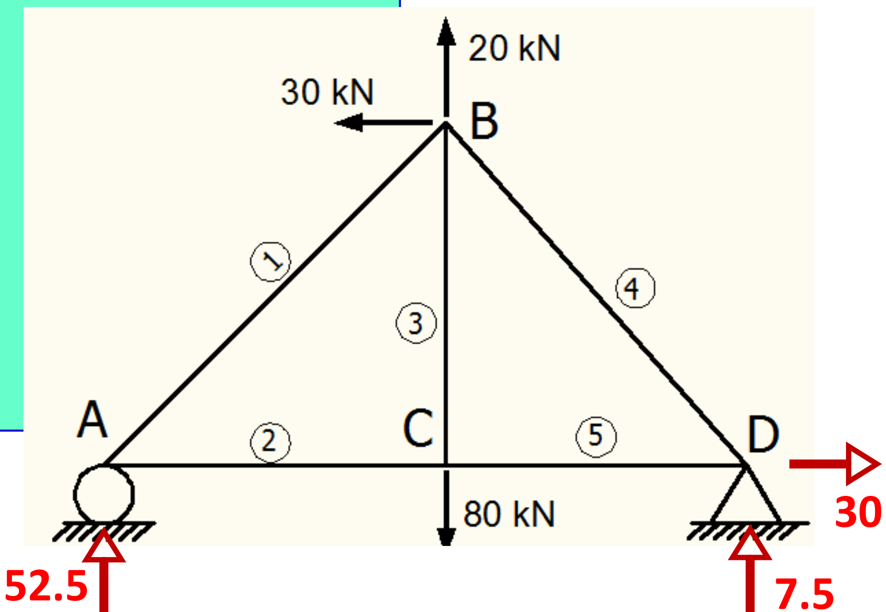
C F = P

$F = C^{-1} P$ ⇐ **Use Matlab to solve**

Problem Example 6 (cont'd)

```
>> ca1=cosd(53.13);sa1=sind(53.13);
>> ca2=cosd(38.66);sa2=sind(38.66);
>> c=[ca1   1   0   0    0
       sa1   0   0   0    0
      -ca1   0   0 ca2    0
      -sa1   0  -1 -sa2  0
        0   -1   0   0    1];
>> p=[0;-52.5;30;-20;0];
>> f=inv(c)*p
f =
  -65.6251
   39.3751
   80.0002
  -12.0061
   39.3751
```

F1
F2
F3
F4
F5



More practice see Problem A3

Thank You