1.6 Calculation on polynomials

First part: Basic elements

1.6 Calculation on polynomials

1.6.1 Operations on polynomials in MATLAB

- **Functions**

conv	polynomial product
residue	decomposition into simple elements
roots	Find the roots of a polynomial
poly	Find the polynomial from its roots
polyval	Evaluates the polynomial

In Matlab, polynomials are represented in the form of line vectors whose components are given in order of decreasing powers. A degree polynomial is represented by a size vector (N+1).

Example

the polynomial $f(x) = 8x^5 + 2x^3 - 3x^2 + 4x - 2$ is represented by :

```
f=[8 0 2 -3 4 -2]
f =8 0 2 -3 4 -2
```

Other functions in MATLAB such as : '***conv***', '***deconv***', '***roots***', etc. can be used in addition to vector-specific operations.

- **Multiplication of polynomials**

The 'conv' function gives the product of convolution of two polynomials.

Example 1

$$\begin{cases} f(x) = -2x^3 + 5x^2 - 3x + 1 \\ g(x) = x^4 - 3x^2 - 5x - 8 \end{cases}$$

The Convolution product g (x) = f (x) .g (x): is given by:

```
f=[-2 5 -3 1];
g=[1 -3 -5  -8];
h=conv(f,g)
h =
    -2     11     -8      1    -28     19     -8
```

Example 2

```
(3x -2 )(2x - 1 ) = ?
p1=[ 3 -2 ];
p2=[ 2 -1 ];
conv( p1 , p2 )
```

```
ans =
      6     -7       2
```

In other words : (3x –2)(2x – 1) = 6x² -7x +2.

- **Polynomial Division**

The '*deconv*' function gives the convolution ratio of two polynomials. The following example shows the use of this function. Let the same previous functions f (x) and g (x):
La division de f(x) par g(x) :

```
clc
f=[ 3 -2 3 5 ];
g=[ 2 -1 2 ];
h=deconv(f,g)
h =
    1.5000   -0.2500
```

And the polynomial obtained is: $h(x) = 1.5x - 0.25$

1.6.2 Handling Polynomial Functions in MATLAB

Let be the following polynomial:$P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots .. a_2 x^2 + a_1 x^1 + a_0$ where n is the degree of the polynomial and $a_i (i = 0,1,2, \ldots n)$ are the coefficients of the polynomial.
This polynomial can be written as:$P(x) = ((\ldots(\ldots(a_n x^n + a_{n-1})x^{n-1} + a_{n-2})x \ldots + a_1)x^1 + a_0)$.

After factorization, we have: $P(x) = a_n(x - r_1)(x - r_2)(x - r_3) \ldots (x - r_n)$ where $r_0, r_1, r_2 \ldots r_n$ are the roots of the polynomial.

Example 1

The polynomial $P(x) = 2x^3 + 10x^2 - 6x + 20$, that is represented in MATLAB by:

```
P=[1 5 -3 10];
```

To find these r_i, roots, one must perform the function ***'roots'***.

```
r=roots(P);
and the result given is:

clc; clear all
P=[2 10 -6 20];
 r=roots(P)
 r =

  -5.8122
   0.4061 + 1.2472i
   0.4061 - 1.2472i
```

The three roots of this polynomial are given as a column vector. When the racinesr_i are known, the coefficients can be recalculated by the order '*poly*'.

```
>>poly(r)
```

```
ans =

    1.0000    5.0000   -3.0000   10.0000
```

Example 2

```
clc; clear all
P =  [ 2 -4 -8 4 -14 16 ]
 r=roots(P)
 poly(r)
format long e
roots(P)

P =
     2    -4    -8     4   -14    16
r =
   3.1942
  -1.9745
  -0.0496 + 1.2000i
  -0.0496 - 1.2000i
   0.8794

ans =
    1.0000   -2.0000   -4.0000    2.0000   -7.0000    8.0000>>
format long e
>> roots(p)
ans =

    3.194190220624354e+000

   -1.974500417245445e+000

   -4.955755139294071e-002 +1.199959128943957e+000i

   -4.955755139294071e-002 -1.199959128943957e+000i

    8.794252994069751e-001
```

Example 3

Complex coefficient polynomial: $(1-i)x^2 + (2-5i)x + 8 = 0$

```
clc; clear all
P =  [ 1-i 2-5i 8]
 r=roots(P)
```

First part: Basic elements

```
format short e

P =

  1.0000e+000    -1.0000e+000i        2.0000e+000    -5.0000e+000i
8.0000e+000
r =
-3.3768e+000 +2.7863e+000i
 -1.2324e-001 -1.2863e+000i
```

Example 4

```
clc; clear all
A=[4 6;1 3]
p=poly(A)
 A =
     4      6

     1      3

p =

     1     -7      6
```

Thus, the characteristic polynomial of the matrix A is: $P(x) = x^2 - 7x + 6$. The roots of this polynomial are the eigenvalues of the matrix A. these roots can be obtained by the function **'*eig*'** :

```
>> Val_prop=eig(A)
Val_prop =
     6
     1
```

1.6.3 Evaluation of a Polynomial

To evaluate the polynomial p(x) at a given point, we must use the 'polyval' function. We evaluate this polynomial for x=1, for example:

```
clc; clear all
P = [ 2 -7 10 ]
 polyval(P,1)
P =
   2   -7   10
ans =
   5
```

1.6 .4 Determining the coefficients of a polynomial from its roots

Example

```
clc; clear all
 a = [ 3 -5 ]
 poly(a)
a =
     3    -5

ans =
     1      2    -15        (that is to say: x² +2x -15).
clc; clear all
a = [ 1+i 2-i 5]
 poly(a)

a =
  1.0000e+000   +1.0000e+000i       2.0000e+000   -1.0000e+000i
5.0000e+000
ans =
  1.0000e+000             -8.0000e+000              1.8000e+001
+1.0000e+000i -1.5000e+001 -5.0000e+000i
Verification
ans =
  1.0000e+000             -8.0000e+000              1.8000e+001
+1.0000e+000i -1.5000e+001 -5.0000e+000i

p =

  1.0000e+000             -8.0000e+000              1.8000e+001
+1.0000e+000i -1.5000e+001 -5.0000e+000i
ans =
  5.0000e+000 +8.8818e-016i
  2.0000e+000 -1.0000e+000i
  1.0000e+000 +1.0000e+000i
```

1.6.5 Graphic Representation

Example

$y = f(x) = x^2 - 3x + 1$

- **Using the function <u>plot</u>**

```
 clc; clear all
p = [ 6 -4 1 ]
 x = 0 : 0.01 : 2;
 y = polyval( p , x)
```

First part: Basic elements

```
plot (x , y)
grid on
```

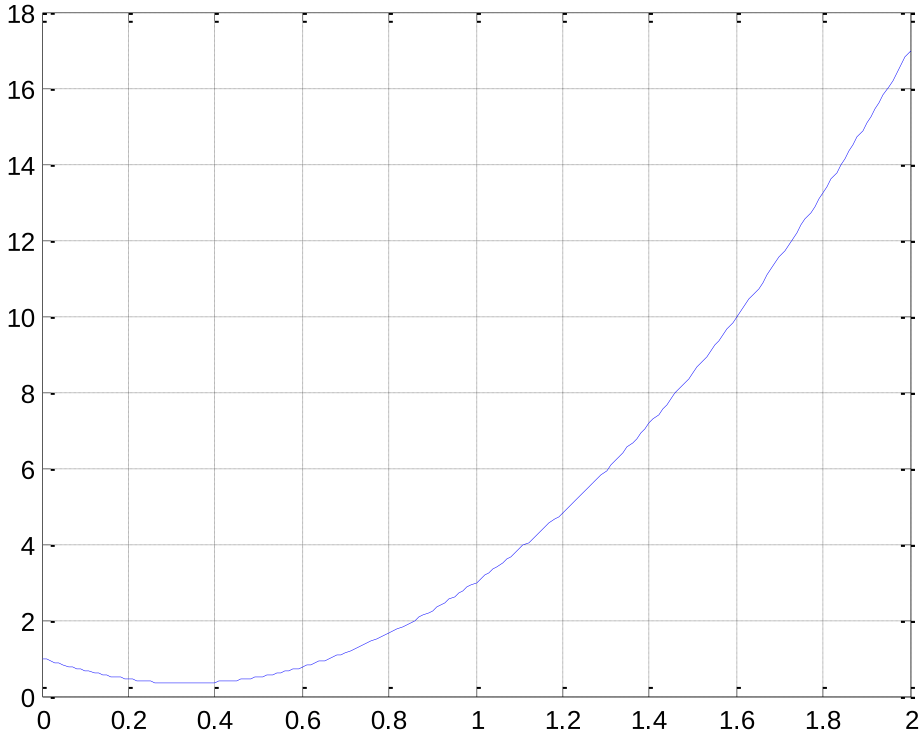
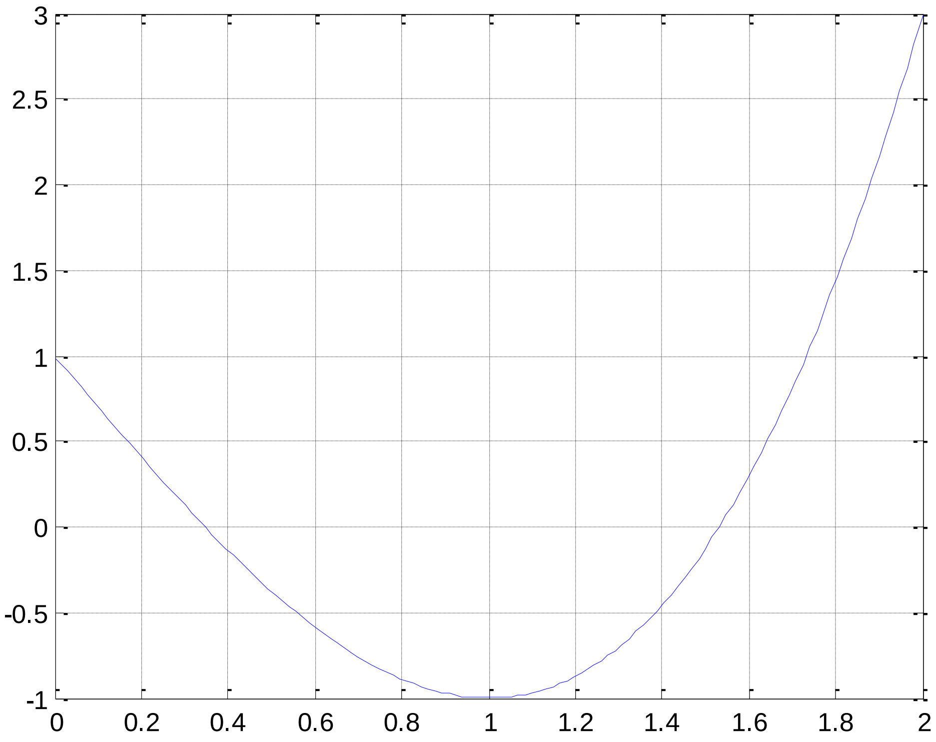


Fig. 1 – Graph of the polynomial p(x) .

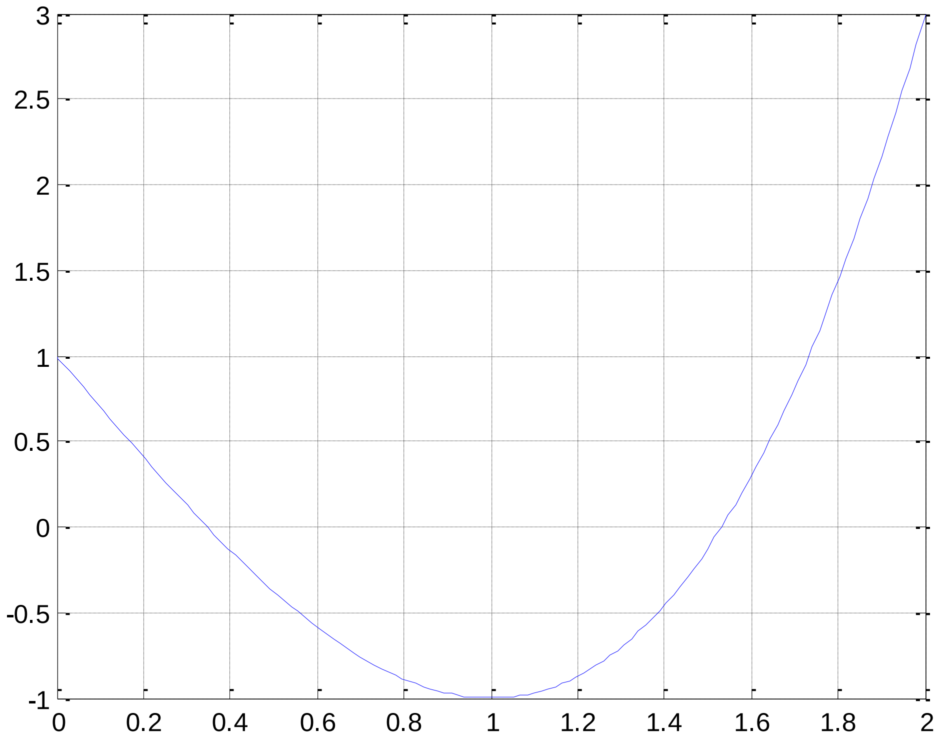
- **Using the function <u>fplot</u>**

```
clc; clear all
fplot ( 'x^3 - 3*x + 1', [ 0 2 ] )
 grid on
```



You must create the .m file of the function:

```
function y=h33(x)
P=[1 0 -3 1];
y=polyval(P,x);
>> h3(5)
ans =
   111
fplot ('h33', [ 0 2] ),grid on
```


3
2.5
2
1.5
1
0.5
0
-0.5
-1
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2