4. Roots of algebraic and transcendental equations

- Roots of algebraic (polynomial) equations
- User-defined functions
- Roots of transcendental equations
- Symbolic computation
Roots of polynomial equations: roots

• To find the roots of a 2\textsuperscript{nd} order polynomial equation $x^2-x-2=(x-2)(x+1)=0$, type as follows:

  ```matlab
  >> C=[1,-1,-2];
  >> roots(C)
  ans =
      2
     -1
  ```

• Roots of a 3\textsuperscript{rd} order equation $x^3+1=0$ are calculated as follows:

  ```matlab
  >> C=[1,0,0,1];
  >> roots(C)
  ans =
     -1.00000 + 0.00000i
     0.50000 + 0.86603i
     0.50000 - 0.86603i
  ```
User-defined functions

• You can define an arbitrary function by writing a script of the form:

```matlab
function [y1,...,yN] = myfun(x1,...,xM)
    y1 = ...
    ...
end
```

• Save the following script into, say, “myfun.m”

```matlab
%myfun.m
function y = myfun(x)
    y = x^2+sin(x)-1;
end
```

• You can call it as a function in the following ways:

```matlab
>> myfun(0)
ans = -1
>> myfun(1)
ans = 0.84147
```

Remark: These commands must be run in the same directory (folder) as myfun.m was saved. Or you can add the directory where myfun.m exists to Octave’s load path; type “help path” for details.
Anonymous function

- You can use *anonymous function*, which is another way of creating a user-defined function

```
>> myfun1 = @(x) (x^2+sin(x)-1);
>> myfun1(1)
ans = 0.84147
```

- An example of functions with two (and more) variables:

```
>> myfun2 = @(x,y) (x.^2+y.^2+x.*y);
>> [X,Y] = meshgrid(-10:10);
>> mesh(X,Y,myfun2(X,Y))
```

Remark: The use of $x.^2$ instead of $x^2$ above makes it possible to deal with the case when $x$ is a matrix (or a vector or even a tensor).
Roots of transcendental equation: \texttt{fsolve}

- To find roots of $x^2 + \sin(x) - 1 = 0$, type as follows:

  ```octave
  >> \texttt{fsolve(@(x) x^2+sin(x)-1, 1.0)}
  ans = 0.63673
  >> \texttt{fsolve(@(x) x^2+sin(x)-1, -1.0)}
  ans = -1.4096
  ```

- \texttt{fsolve} tries to find a root starting from given initial value
- It can fail to find any root; the success depends on the equation and the provided initial values

**20.1 Solvers**

From https://www.gnu.org/software/octave/doc/

Octave can solve sets of nonlinear equations of the form

\[ F(x) = 0 \]

using the function \texttt{fsolve}, which is based on the MINPACK subroutine \texttt{hybrd}. This is an iterative technique so a starting point must be provided. This also has the consequence that convergence is not guaranteed even if a solution exists.

Function File: \texttt{fsolve (fcn, x0, options)}
Function File: \texttt{[x, fvec, info, output, fjac] = fsolve (fcn, ...)}

\texttt{fsolve} should accept a vector (array) defining the unknown variables, and return a vector of left-hand sides of the equations. Right-hand sides are defined to be zeros. In other words, this function attempts to determine a vector $x$ such that $Fcn(x)$ gives (approximately) all zeros.

$x0$ determines a starting guess. The shape of $x0$ is preserved in all calls to $fcn$, but otherwise it is treated as a column vector.
Symbolic package

• Extends Octave to enable symbolic computation
  • Function `solve` in MATLAB has not been implemented as of today

• To install `symbolic` package, visit [https://github.com/cbm755/octsympy](https://github.com/cbm755/octsympy) and follow the instruction.

• To use this package, type the following in Command Window:

  ```
  >> pkg load symbolic
  ```

• To start symbolic computation, you must first declare a symbolic variable by `syms`

  ```
  >> syms x
  ```

• A symbolic representation of a function:

  ```
  >> x^2+sin(x)-1
  ans = (sym)
    2
    x + sin(x) - 1
  ```

Note: Besides MATLAB/Octave, there are a lot of symbolic computation software, or computer algebra systems; Wolfram Mathematica is a popular one.

http://www.wolframalpha.com
Symbolic package: factorization

- Factorization of a polynomial: factor

```matlab
>> syms x
>> f=x^3+13*x^2-105*x+171;
>> factor(f)
ans = (sym)
    2
   (x - 3) *(x + 19)
```

```matlab
>> syms x y
>> f=x^3*y-3*x^3-4*x^2*y+12*x^2-3*x*y+9*x+18*y-54;
>> factor(f)
ans = (sym)
    2
   (x - 3) *(x + 2)*(y - 3)
```
Symbolic package: differential

• Symbolic differential: diff

$$\text{diff}(x^2 + \sin(x) - 1)$$
$$\text{ans} = (\text{sym}) \ 2x + \cos(x)$$

$$\text{diff}(\exp(-x\sin(x)))$$
$$\text{ans} = (\text{sym}) \ -x\sin(x)$$
$$\quad -x\cos(x) - \sin(x)*e$$
Symbolic package: indefinite integral

• Indefinite integral : int

>> int(x^2+sin(x)-1)
ans = (sym)

3
x
-- - x - cos(x)
3

>> int(sin(log(x)))
ans = (sym)

x*sin(log(x))  x*cos(log(x))
------------  ------------
2             2
Exercises 4.1

• Find all the roots to the following equation

\[(\sin x)^2 \exp \left( -\frac{x}{2} \right) - 0.1 = 0, \quad (x \geq 0)\]

• Hint: You must specify good initial values to use \texttt{fsolve}. To do so, plot the function \( y = f(x) \) in the interval \([0,10]\) as follows and make guesses of possible roots.

\[
\begin{align*}
\texttt{x} &= 0:0.01:10; \\
\texttt{y} &= \texttt{sin(x).}^\texttt{2.*exp(-x/2)} - 0.1; \\
\texttt{y0} &= \texttt{zeros(1,length(x))}; \\
\texttt{plot(x,y,x,y0)}
\end{align*}
\]