Numerical Analysis and Computational Mathematics

Fall Semester 2024 – CSE Section

Prof. Laura Grigori

Assistant: Israa Fakih

Session 1 – September 11, 2024

Introduction to Matlab® /Octave

Exercise I (MATLAB)

For a triangle, if the lengths of two sides, a and b, and the angle α between them are known, then the length of the third side c can be calculated by the *law of cosines*:

$$c^2 = a^2 + b^2 - 2ab\cos(\alpha).$$

Define an anonymous function in MATLAB that computes the length of the third side c by using the notation:

```
c = @(a,b,alpha ) ...
```

such that `c` becomes a handle to a function that can be evaluated as `c(a,b,alpha)`. Verify the correct implementation of the function by testing it for some simple triangles (e.g. an equilateral triangle for which we have $a = b = 1$, $\alpha = \pi/3$).

Exercise II (MATLAB)

a) Write a MATLAB file called `logplot.m` containing a script that:

- creates a vector `x` of 100 linearly spaced points between 0 and 1,000 (*hint*: use the MATLAB command `linspace`);
- creates a function handle `f` to an anonymous function that evaluates:

$$f(x) = [x - \log(x+1)]^4;$$

- plots the function `f` at the points `x` using the plotting commands `plot`, `semilogx`, `semilogy` and `loglog` (use `figure(1)`, `figure(2)`, ... to directly plot the different figures).

b) Following point a), indicate which of the plots is the most useful if we want to estimate the order of growth of $f(x)$, i.e. the exponent p in $f(x) = O(x^p)$.

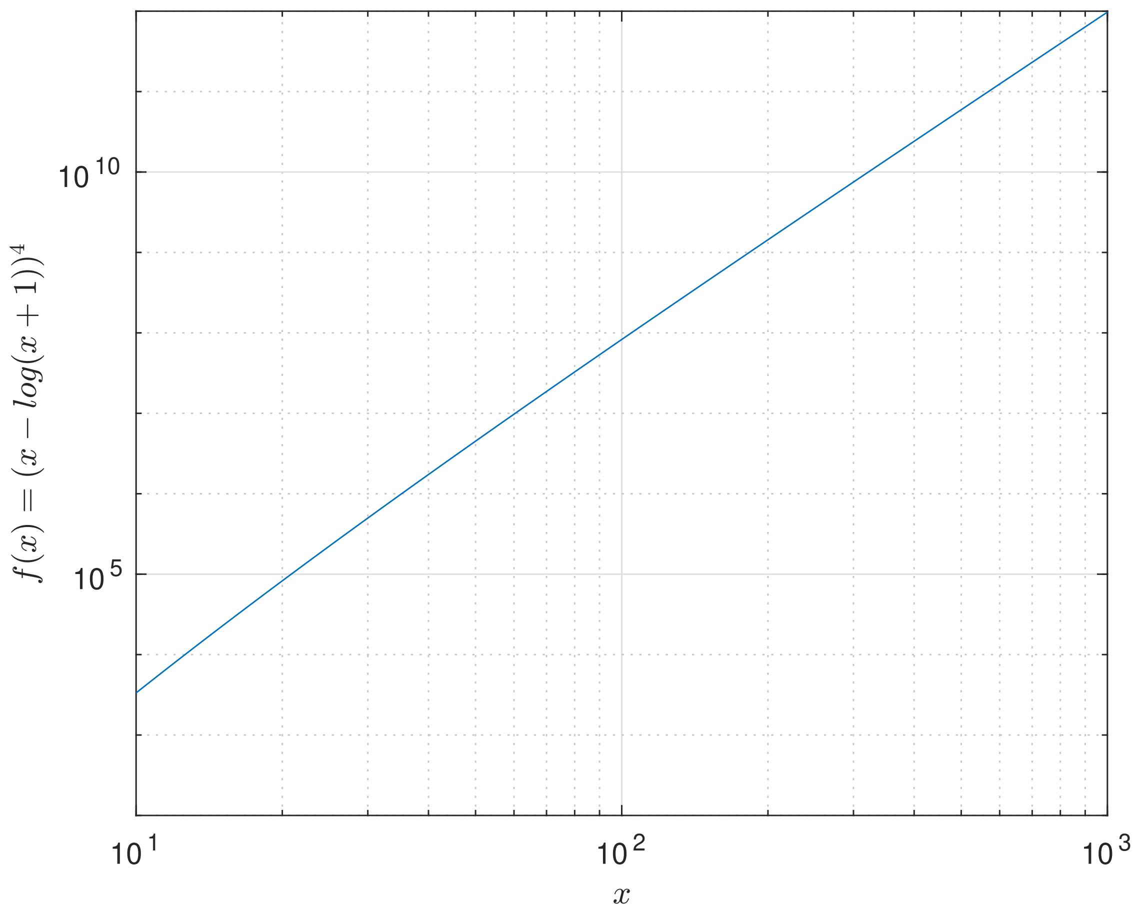

Figure 1: Example of a function that grows very fast.

c) Following point a), experiment with the commands `xlabel`, `ylabel`, `grid`, `title` and `axis` to make the plot more understandable by adding descriptive axis labels, a grid and a title. Reproduce the example reported in Figure 1.

Exercise III (MATLAB)

Let us consider the approximation of the derivative of a regular enough function $f(x)$ in $x = x_0$, say $f'(x_0)$, by means of a centered finite difference scheme for which:

$$f'(x_0) \simeq \delta_{c,h}\, f(x_0) := \frac{f(x_0 + h) - f(x_0 - h)}{2h},$$

for some $h > 0$; the corresponding error is $E_{c,h} = |f'(x_0) - \delta_{c,h} f(x_0)|$. Let us assume that, for a given function $f(x)$ and point x_0, we obtain the following errors E_{c,h_i} associated to different values of h_i for $i = 1, \ldots, n$ with $n = 6$:

h_i	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}
E_{c,h_i}	$1.147 \cdot 10^{-1}$	$2.840 \cdot 10^{-2}$	$7.084 \cdot 10^{-3}$	$1.770 \cdot 10^{-3}$	$4.425 \cdot 10^{-4}$	$1.106 \cdot 10^{-4}$

By conjecturing that the error can be written as $E_{c,h_i} = Ch_i^p$, with C independent of h_i and p, determine the convergence order p of the method for this particular case first *graphically* and then *algebraically*.

Hint: Note that $\frac{E_{c,h_i}}{E_{c,h_j}} = \left(\frac{h_i}{h_j}\right)^p$, so that p can be obtained *algebraically* as: $p = \frac{\log\left(E_{c,h_i}/E_{c,h_j}\right)}{\log(h_i/h_j)}$. *Graphically*, the curves $(h, E_{c,h})$ and (h, h^p) should be parallel in log-log scale. For more details, read remark 1.10 in the lecture notes.

Exercise IV (MATLAB)

Find an efficient way in MATLAB to assign the following matrix:

$$M = \begin{bmatrix} 7 & 8 & 9 & 10 \\ 12 & 10 & 8 & 6 \end{bmatrix},$$

without entering manually each element (*hint*: use the command `a:b:c` to create two row vectors and then combine them to form a matrix).
Suitably use the MATLAB commands to:

a) extract the element in the first row, third column of M;

b) extract the entire second row of M;

c) extract the first two columns of M;

d) extract the vector containing all the elements of the second row of M except for the third element.

Exercise V (MATLAB)

We want to compute the function $f(x) = (\sqrt{1+x} - 1)/x$ for different values of x in a neighborhood of 0. We first notice that $f(x)$ can be equivalently written as $f(x) = 1/(\sqrt{1+x} + 1)$ and also as $f(x) = 1/2 - x/8 + x^2/16 - 5x^3/128 + o(x^4)$.
Create three function handles representing the above definitions of $f(x)$ (*hint*: the term $o(x^4)$ can be neglected in the computation) and, for each function handle,

a) evaluate $f(x)$ at $X = [10^{-10}\, 10^{-12}\, 10^{-14}\, 10^{-16}]$ using a `for` loop;

b) evaluate $f(x)$ at the same points given in a) using MATLAB vector algebra;

c) display the results and comment on the importance of *round-off errors* for this example.