Universitatea din Bucureşti
Facultatea de Matematică şi Informatică

Analiză Numerică
14.10.2020

# LABORATOR#2

## I. ECUAŢII NELINIARE: METODE ITERATIVE DE PUNCT FIX

**ALGORITM** (Metodă iterativă de punct fix)

| |
|---|
| **Date:** $\phi$ `(construită pornind de la` $f$`),` $a$, $b$; |
| $n = 0:\ x_n \in [a, b]$; |
| $n \geq 1:\ x_n = \phi(x_{n-1})$; |
| $\quad n = n + 1$; `repeat step for` $n \geq 1$; |

**OBS:** Metodele iterative de punct fix au **viteza/ordinul de convergenţă cel puţin liniară.**

**EX#1** Fie ecuaţia:

$$x^3 + 4x^2 - 10 = 0\,, \quad x \in [1, 2]\,. \tag{1}$$

(a) Să se construiască în MATLAB® o procedură cu sintaxa $[x_{\text{aprox}}] = \texttt{MetPunctFix}(\phi, x_0, N)$, $x_{\text{aprox}}$ fiind soluţia aproximativă generată de metodă cu datele respective.

(b) Să se implementeze, într-un fişier script, următoarele cerinţe:

(b1) Să se construiască graficul funcţiei $f(x) = x^3 + 4x^2 - 10$ pe intervalul $[1, 2]$.

(b2) Să se afle soluţia exactă $x^\star$ cu ajutorul funcţiei simbolice `solve` predefinită de MATLAB® şi să se aleagă acea soluţie care se află în intervalul $[1, 2]$.

(b3) Considerăm funcţiile $\phi_j : \mathscr{D}_{\phi_j} \cap [1, 2] \longrightarrow \mathbb{R}$, $j = 1, 2, 3, 4$, unde $\mathscr{D}_{\phi_j}$ sunt domeniile de definiţie ale funcţiilor $\phi_j$, $j = \overline{1, 4}$, definite prin:

(i) $\phi_1(x) = -x^3 - 4x^2 + x + 10$;

(ii) $\phi_2(x) = \sqrt{(10/x) - 4x}$;

(iii) $\phi_3(x) = \dfrac{1}{2}\sqrt{10 - x^3}$;

(iv) $\phi_4(x) = \sqrt{10/(x + 4)}$.

Să se construiască graficele funcţiilor $\phi_j$, $|\phi_j'|$, $j = \overline{1, 4}$, pe intervalul maxim $[1, 2]$ şi să se determine care dintre acestea verifică ipotezele Teoremei lui Brouwer.

(b4) Să se construiască aproximările $x_n$, $n = \overline{1, N}$, pentru soluţia ecuaţiei (1), apelând procedura `MetPunctFix` cu $\phi_j$, $j = \overline{1, 4}$, $x_0 = 1$ şi $N = 20$.

(b5) Care dintre funcţiile $\phi_j$, $j = \overline{1, 4}$, generează cea mai rapidă metodă de punct fix în cazul alegerii valorii iniţiale $x_0 = 1$?

(b6) Calculaţi eroarea absolută, $e_a(x_n) = |x^\star - x_n|$, $n = \overline{1, N}$.

## II. ECUAŢII NELINIARE: METODA NEWTON-RAPHSON

**ALGORITM** (Metoda Newton-Raphson)

> **Date:** $f, f', a, b$;
>
> $n = 0:\ x_n \in [a, b]$;
>
> $n \geq 1:\ x_n = x_{n-1} - \dfrac{f(x_{n-1})}{f'(x_{n-1})}$;
>
> $n = n + 1$; `repeat step for` $n \geq 1$;

**OBS:** Metoda Newton-Raphson are **viteza/ordinul de convergenţă cel puţin pătratică.**

**EX#2** Fie $f : [-1, 1] \longrightarrow \mathbb{R}$, $f(x) = x + \mathrm{e}^{-x^2} \cos x$.

(a) Reprezentaţi graficul funcţiei $f$ şi salvaţi imaginea cu numele `Graficf.eps`

(b) Creaţi un fişier funcţie `Functiaf.m` pentru funcţia $f$ şi un fişier funcţie `Derivataf.m` pentru derivata lui $f$.

(c) Creaţi fişierul funcţie `NewtonRaphsonf.m` care determină, folosind structura repetitivă `for`, primele 10 aproximări ale rădăcinii funcţiei $f$ generate de metoda Newton-Raphson cu $x_0 = 0$, apelând în interiorul acestuia funcţiile `Functiaf.m` şi `Derivataf.m`

(d) Creaţi un fişier script prin care se determină, folosind calculul simbolic, rădăcina $x^\star$ a funcţiei $f$ şi să se afişeze graficul funcţiei $\mathrm{err_a}(x_n) = |x^\star - x_n|$, unde $\{x\}_{n\geq 0}$ este şirul de aproximări generat la (c).

**EX#3** (a) Creaţi fişierul funcţie `NewtonRaphson.m` cu datele de intrare $f$, $f'$, prima aproximare $x_0$, `TOL` şi data de ieşire $x_{\mathrm{aprox}}$, generat de metoda Newton-Raphson folosind structura repetitivă `while` şi criteriul de oprire $|f(x_n)| <$ `TOL`.

(b) Fie $f : [0, \pi/2] \longrightarrow \mathbb{R}$, $f(x) = \cos x - x$ şi $x_0 = \pi/4$. Apelaţi fişierul funcţie creat la subpunctul (a) pentru aceste date de intrare.

Afişati, în acelaşi sistem de coordonate $xOy$, graficul funcţiei $f$, dreapta de ecuaţie $y = 0$ şi şirul de aproximări generat.

**OBS:** Apelarea funcţiei se face în fereastra de comandă [de exemplu, `NewtonRaphson`($f$, $f'$, 0.5, 1e-5)], unde $f$ şi $f'$ fie se dau anterior în fereastra de comandă (Command Window) cu ajutorul funcţiei anonime @, fie se prescriu într-un fişier de date `Date.m`.