ESO208A: Computational Methods in Engineering

Richa Ojha

Department of Civil Engineering

IIT Kanpur

Acknowledgements: Profs. Abhas Singh and Shivam Tripathi (CE)

Example Problem

- *Determine the roots of the polynomial*

$$f(x) = x^5 - 3.5x^4 + 2.75x^3 + 2.125x^2 - 3.875x + 1.25$$

Use initial guesses of r = s = -1 and iterate to $\varepsilon_a \leq 0.1\%$

Example Problem

- *Determine the roots of the polynomial*

$$f(x) = x^5 - 3.5x^4 + 2.75x^3 + 2.125x^2 - 3.875x + 1.25$$

Use initial guesses of r = s = -1 and iterate to $\varepsilon_a \leq 0.1\%$

Soln:

Step 1: Input $a_0, a_1, \ldots a_n$ and initialize r and s.

In $p_n(x) = \sum_{k=0}^{n} a_k x^k = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$

Here n = 5;

$p_5(x) = \sum_{k=0}^{5} a_k x^k = a_0 + a_1 x + a_2 x^2 + \cdots + a_5 x^5$

$a_0 = 1.25$; $a_1 = -3.875$; $a_2 = 2.125$; $a_3 = 2.75$;

$a_4 = -3.5$; $a_5 = 1$;

Example Problem

Step 2: compute b_0, b_1, … b_n *using* recursive relations derived $b_n = a_n$;

$$b_{n-1} = a_{n-1} + rb_n \,;$$

$$b_i = a_i + rb_{i+1} + sb_{i+2};\ i = (n-2), \cdots 2, 1, 0$$

Here, n =5

$$b_5 = a_5; b_4 = a_4 + rb_5\,; b_i = a_i + rb_{i+1} + sb_{i+2};\ i = 3, 2, 1, 0$$

$$b_3 = a_3 + rb_4 + sb_5;$$

$$b_2 = a_2 + rb_3 + sb_4$$

$$b_1 = a_1 + rb_2 + sb_3$$

$$b_0 = a_0 + rb_1 + sb_2$$

Example Problem

Step 3: compute c_0, c_1, … c_n *using* recursive relations derived

$$c_n = b_n;$$

$$c_{n-1} = b_{n-1} + rc_n\;;$$

$$c_i = b_i + rc_{i+1} + sc_{i+2}; \qquad i = (n-2), \cdots 2, 1, 0$$

Here, n =5

$c_5 = b_5; c_4 = b_4 + rc_5\;; c_i = b_i + rc_{i+1} + sc_{i+2};\ i = 3, 2, 1, 0$

$$c_3 = b_3 + rc_4 + sc_5$$

$$c_2 = b_2 + rc_3 + sc_4$$

$$c_1 = b_1 + rc_2 + sc_3$$

$$c_0 = b_0 + rc_1 + sc_2$$

Example Problem

Step 4: compute Δr and Δs from $\begin{bmatrix} c_1 & c_2 \\ c_2 & c_3 \end{bmatrix} \begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} -b_0 \\ -b_1 \end{bmatrix}$

Here, $\begin{bmatrix} -16.375 & -4.875 \\ -4.875 & 10.75 \end{bmatrix} \begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} -11.375 \\ 10.5 \end{bmatrix}$

Solving, $\Delta r = 0.3558$ and $\Delta s = 1.1381$

Step 5: compute $r_{new} = r + \Delta r$, $s_{new} = s + \Delta s$

$r_{new} = -1 + 0.3558 = -0.6442$, $s_{new} = -1 + 1.1381 = 0.1381$

Step 6: check for convergence, $\left|\frac{r_{new} - r}{r_{new}}\right|, \left|\frac{s_{new} - s}{s_{new}}\right| \leq \varepsilon;\ b_0, b_1 \leq \varepsilon'$

$|\varepsilon_{a,r}| = \left|\frac{r_{new} - r}{r_{new}}\right| = \left|\frac{-0.6442 \quad (-1)}{-0.6442}\right| 100\%;$ $|\varepsilon_{a,s}| = \left|\frac{s_{new} - s}{s_{new}}\right| = \left|\frac{0.1381 \quad (-1)}{0.1381}\right| 100\%;$

Step 7: Stop if all convergence checks are satisfied. Else, set $r = r_{new}$, $s = s_{new}$ and go to step 2.

Revision of Solution of Non-linear Equations

1. Graphical Method – Provide insights but tedious/subjective
2. Bracketing methods
   1. *Bisection method*
   2. *False position method*
   3. *Modified false position method*

   Guaranteed convergence
   Linear or better convergence
3. Open methods
   1. *Fixed-point iteration*
   2. *Newton-Raphson*
   3. *Secant & Modified Secant*

   May diverge
   FP - linear convergence
   NR – quadratic convergence
   Secant – between linear & quadratic
   NR – problems near zero gradient

Revision of Solution of Non-linear Equations

Hybrid Methods

1. *Dekker method*
2. *Brent method*

Combination
- Bracketing method at the beginning
- Open method near convergence

Multiple roots

1. *Bracketing method* – Only for odd number of roots
2. *Newton-Raphson* - Linear convergence
3. *Modified Newton Raphson* – Quadratic convergence
    a. Known multiplicity
    b. Derivative function

Revision of Solution of Non-linear Equations

Roots of polynomials

1. Evaluation of polynomials

2. Division of polynomials

3. Deflation of polynomials

4. Effective degree of polynomials

Method of finding roots

1. Müller method

2. Bairstow method

Real and complex roots

Revision of Solution of Non-linear Equations

1. Except for rare cases, computers will provide approximate solution.
2. No method is “universally” better than others.
3. Domain knowledge should guide the selection of algorithm and guess value(s).

Comparison of different algorithms

Method	Type	Guesses	Convergence	Stability	Programming	Comments
Direct	Analytical	—	—	—		
Graphical	Visual	—	—	—	—	Imprecise
Bisection	Bracketing	2	Slow	Always	Easy	
False-position	Bracketing	2	Slow/medium	Always	Easy	
Modified FP	Bracketing	2	Medium	Always	Easy	
Fixed-point iteration	Open	1	Slow	Possibly divergent	Easy	
Newton-Raphson	Open	1	Fast	Possibly divergent	Easy	Requires evaluation of $f'(x)$
Modified Newton-Raphson	Open	1	Fast (multiple), medium (single)	Possibly divergent	Easy	Requires evaluation of $f'(x)$ and $f''(x)$
Secant	Open	2	Medium/fast	Possibly divergent	Easy	Initial guesses do not have to bracket the root
Modified secant	Open	1	Medium/fast	Possibly divergent	Easy	
Brent	Hybrid	1 or 2	Medium	Always (for 2 guesses)	Moderate	Robust
Müller	Polynomials	2	Medium/fast	Possibly divergent	Moderate	
Bairstow	Polynomials	2	Fast	Possibly divergent	Moderate	