Lecture Note-Numerical Analysis (5): Roots of the Polynomial Equation

1. Definition of n-th order polynomials and their computation

- **N-th order polynomials**

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + a_{n-3} x^{n-3} + \cdots + a_1 x + a_0, \quad (a_n \neq 0)$$
$$= a_0 + a_1 x \cdots + a_{n-3} x^{n-3} + a_{n-2} x^{n-2} + a_{n-1} x^{n-1} + a_n x^n$$

- **Computation of n-th order polynomial**

(1) $f(x) = a_0 + a_1 x \cdots + a_{n-3} x^{n-3} + a_{n-2} x^{n-2} + a_{n-1} x^{n-1} + a_n x^n$

Number of multiplications = 1+3+4+5+….. +(n+1) = (n+1)(n+2)/2 = O(n^2)
Number of additions = n

(2) $f(x) = a_0 + x(a_1 + x(a_2 + x(a_3 + x(\cdots + x(a_{n-1} + a_n x)))\cdots)$

Number of multiplications = 1+1+1+1+….. +1 = n = O(n)
Number of additions = n

```
Function PolyVal(n, a, x, p)
! Pseudo code for calculating n-th order polynomial
! input: n(order), a(1:n+1)(coefficients), x(independent variables)
! outout: p(polynomial value)
!-------------------------------------------------------------------------------------------------------------------
 p = a(n)
 do j=n, 1, -1
        p = a(j-1) + x*p
 end do
!-------------------------------------------------------------------------------------------------------------------
End PolyVal
```

2. Polynomial Deflation: Removal of roots from a polynomial

- **Removal of one root $x = \alpha$ from f(x) to get (n-1)-th order polynomial g(x)**

$$f(x) = a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + a_{n-3}x^{n-3} + \cdots + a_1 x + a_0, \quad (a_n \neq 0)$$
$$= (x-\alpha)g(x) + r_0 \qquad (r_0 = 0)$$
$$g(x) = b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + b_{n-3}x^{n-3} + \cdots + b_1 x + b_0, \quad (b_{n-1} \neq 0)$$

If $\alpha = 0$, $b_{j-1} = a_j$, $j = 1,2,\cdots,n$ with $a_0 = 0$

If $\alpha \neq 0$,

$$f(x) = (x-\alpha)g(x)$$
$$= (x-\alpha)(b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + b_{n-3}x^{n-3} + \cdots + b_1 x + b_0) + r_0$$
$$= x(b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + b_{n-3}x^{n-3} + \cdots + b_1 x + b_0) + r_0$$
$$-\alpha(b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + b_{n-3}x^{n-3} + \cdots + b_1 x + b_0) + r_0$$
$$= b_{n-1}x^n + (b_{n-2} - \alpha b_{n-1})x^{n-1} + (b_{n-3} - \alpha b_{n-2})x^{n-2} + \cdots + (b_1 - \alpha b_2)x^2 + (b_0 - \alpha b_1)x - \alpha b_0 + r_0$$

Therefore, we can get the following relations and pseudo code

$$\begin{array}{r} b_{n-1} = a_n \ (\\ b_{n-2} - \alpha b_{n-1}) = a_{n-1} \\ (b_{n-3} - \alpha b_{n-2}) = a_{n-2} \\ \vdots \\ (b_1 - \alpha b_2) = a_2 \\ (b_0 - \alpha b_1) = a_1 \\ -\alpha b_0 + r = a_0 \end{array} \rightarrow \begin{array}{l} b_{n-1} = a_n \\ b_{n-2} = a_{n-1} + \alpha b_{n-1} \\ b_{n-3} = a_{n-2} + \alpha b_{n-2} \\ \vdots \\ b_1 = a_2 + \alpha b_2 \\ b_0 = a_1 + \alpha b_1 \\ r_0 = a_0 + \alpha b_0 \end{array} \rightarrow \boxed{\begin{array}{l} b_{n-1} = a_n \\ b_j = a_{j+1} + \alpha b_{j+1}, \ (j = n-2, \cdots, 0) \end{array}}$$

If $x = \alpha$ **is not the root (not quotient), then there exists a constant residual such as**

$$r_0 = a_0 + \alpha b_0$$

○ **Removal of two roots** $x = \alpha, x = \beta \quad (\alpha, \beta \in C; \text{complex pairs})$ **from f(x) to get (n-2)-th order polynomial g(x) when the order of polynomial n is greater than 2 (n=3,5,…).**

$$\begin{aligned} q(x) &= (x-\alpha)(x-\beta) = x^2 - (\alpha+\beta)x + \alpha\beta \\ &= x^2 + q_1 x + q_0 \end{aligned} \quad \text{,where} \quad \begin{aligned} q_1 &= -(\alpha+\beta) \\ q_0 &= \alpha\beta \end{aligned}$$

$$\begin{aligned} f(x) &= a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + a_{n-3} x^{n-3} + \cdots + a_1 x + a_0, \quad (a_n \neq 0) \\ &= q(x) g(x) + r_1 x + r_0 \qquad (r_1 = 0, r_0 = 0) \\ g(x) &= b_{n-2} x^{n-2} + b_{n-3} x^{n-3} + \cdots + b_1 x + b_0, \quad (b_{n-2} \neq 0) \\ f(x) &= (x^2 + q_1 x + q_0)(b_{n-2} x^{n-2} + b_{n-3} x^{n-3} + \cdots + b_1 x + b_0) \end{aligned} \tag{1}$$

$$
\begin{aligned}
f(x) &= (x^2+q_1x+q_0)(b_{n-2}x^{n-2}+b_{n-3}x^{n-3}+\cdots+b_1x+b_0)+r_1x+r_0\\
&= x^2(b_{n-2}x^{n-2}+b_{n-3}x^{n-3}+\cdots+b_1x+b_0)+r_1x+r_0\\
&\quad + q_1x(b_{n-2}x^{n-2}+b_{n-3}x^{n-3}+\cdots+b_1x+b_0)+r_1x+r_0\\
&\quad + q_0(b_{n-2}x^{n-2}+b_{n-3}x^{n-3}+\cdots+b_1x+b_0)+r_1x+r_0\\
&= b_{n-2}x^n+(b_{n-3}+q_1b_{n-2})x^{n-1}+(b_{n-4}+q_1b_{n-3}+q_0b_{n-2})x^{n-2}\\
&\quad + (b_{n-5}+q_1b_{n-4}+q_0b_{n-3})x^{n-3}+\cdots\\
&\quad + (b_0+q_1b_1+q_0b_2)x^2+(q_1b_0+q_0b_1+r_1)x+q_0b_0+r_0\\
&= a_nx^n+a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+a_{n-3}x^{n-3}+\cdots+a_1x+a_0
\end{aligned}
\tag{2}
$$

Therefore, we can get the following relations and pseudo code

$$
\boxed{
\begin{aligned}
b_{n-2} &= a_n\\
b_{n-3}+q_1b_{n-2} &= a_{n-1}\\
b_{n-4}+q_1b_{n-3}+q_0b_{n-2} &= a_{n-2}\\
b_{n-5}+q_1b_{n-4}+q_0b_{n-3} &= a_{n-3}\\
&\vdots\\
b_1+q_1b_2+q_0b_3 &= a_3\\
b_0+q_1b_1+q_0b_2 &= a_2\\
q_1b_0+q_0b_1+r_1 &= a_1\\
q_0b_0+r_0 &= a_0
\end{aligned}
}
\rightarrow
\boxed{
\begin{aligned}
b_{n-2} &= a_n\\
b_{n-3} &= a_{n-1}-q_1b_{n-2}\\
b_{n-4} &= a_{n-2}-q_1b_{n-3}-q_0b_{n-2}\\
b_{n-5} &= a_{n-3}-q_1b_{n-4}-q_0b_{n-3}\\
&\vdots\\
b_1 &= a_3-q_1b_2-q_0b_3\\
b_0 &= a_2-q_1b_1-q_0b_2\\
r_1 &= a_1-q_1b_0-q_0b_1\\
r_0 &= a_0-q_0b_0
\end{aligned}
}
\qquad \textbf{(3)}
$$

$$\rightarrow \boxed{\begin{aligned} & b_{n-2} = a_n \\ & b_{n-3} = a_{n-1} - q_1 b_{n-2} \\ & b_j = a_{j+2} - q_1 b_{j+1} - q_0 b_{j+2}, \ (j = n-4, n-5, \cdots, 0) \end{aligned}} \tag{4}$$

If $x = \alpha, x = \beta$ **is not the root (not quotient), then there exists a 1st order residual**

$$R(x) = r_1 x + r_0 = (a_1 - q_1 b_0 - q_0 b_1) x + (a_0 - q_0 b_0) \tag{5}$$

On the other hand, if $R(x) = 0$ **with** $r_1 = r_0 = 0$**, the equation becomes**

$$f(x) = (x^2 + q_1 x + q_0)(b_{n-2} x^{n-2} + b_{n-3} x^{n-3} + \cdots + b_1 x + b_0) = 0$$

Therefore, we can obtain two solutions by solving $q(x) = x^2 + q_1 x + q_0 = 0$ **such as**

$$x = \frac{-q_1 \pm \sqrt{q_1^2 - 4q_0}}{2}$$

And, the following equation is left for the next roots corresponding to the following equation

$$g(x) = b_{n-2} x^{n-2} + b_{n-3} x^{n-3} + \cdots + b_1 x + b_0 = 0 \tag{6}$$

Here, Eq (6) can be expressed when $n \le 5$

$$g(x) = \begin{cases} b_1 x + b_0 = 0, & (n=3) \\ b_2 x^2 + b_1 x + b_0 = 0, & (n=4) \\ b_3 x^3 + b_2 x^2 + b_1 x + b_0 = 0, & (n=5) \end{cases} \rightarrow \begin{cases} x = -\dfrac{b_0}{b_1}, & (n=3) \\ x = \dfrac{-b_1 \pm \sqrt{b_1^2 - 4 b_2 b_0}}{2 b_2}, & (n=4) \\ \text{repeat above process} & \end{cases}$$

Therefore, we have an important question how to make $R(x) = 0$ **by adjusting** q_1 and q_0.

- **As shown by Eq (3) and (4), all coefficients of** $b_j, (j = n-2, n-3, \cdots, 0)$, r_1, r_0 **are the functions of** (q_1, q_0)**. And,** $R(x) = 0$ **can be satisfied by the following relations**

$$R(x) = 0 \rightarrow \begin{array}{l} r_1(q_1, q_0) = 0 \\ r_0(q_1, q_0) = 0 \end{array}$$

which is the nonlinear algebraic equation and can be solved using the Newton-Raphson method such as

$$\begin{pmatrix} q_1 \\ q_0 \end{pmatrix}_{k+1} = \begin{pmatrix} q_1 \\ q_0 \end{pmatrix}_k - \alpha \left\{ \frac{\partial(r_1, r_0)}{\partial(q_1, q_0)} \right\}^{-1} \begin{pmatrix} r_1(q_1, q_0) \\ r_0(q_1, q_0) \end{pmatrix}_k$$

Examples of Polynomial Deflation

(a) Formula

$$\boxed{\begin{aligned} b_{n-2} &= a_n \\ b_{n-3} &= a_{n-1} - q_1 b_{n-2} \\ b_j &= a_{j+2} - q_1 b_{j+1} - q_0 b_{j+2},\ (j = n-4, n-5, \cdots, 0) \\ r_1 &= a_1 - q_1 b_0 - q_0 b_1 \\ r_0 &= a_0 - q_0 b_0 \end{aligned}}$$

(b) Example 1: 3rd order polynomial

$$f(x) = x^3 - 1 \rightarrow a_3 = 1,\ a_2 = a_1 = 0,\ a_0 = -1$$

$$\boxed{\begin{aligned} b_1 &= 1 \\ b_0 &= -q_1 \\ r_1 &= q_1^2 - q_0 = 0 && \rightarrow r_1(q_1, q_0) = 0 \\ r_0 &= -1 + q_0 q_1 = 0 && \rightarrow r_2(q_1, q_0) = 0 \end{aligned}} \rightarrow \boxed{\begin{aligned} q_1^3 &= 1 \\ q_1 &= 1 \\ q_0 &= 1 \\ b_1 &= 1 \\ b_0 &= -1 \end{aligned}} \rightarrow f(x) = (x-1)(x^2 + x + 1)$$

(c) Example 2: 4th order polynomial

$$f(x) = x^4 + x^2 + 1 \rightarrow a_4 = 1,\ a_3 = 0,\ a_2 = 1,\ a_1 = 0,\ a_0 = 1$$

$$\boxed{\begin{aligned}
&b_2 = a_4 = 1\\
&b_1 = a_3 - q_1 b_2 = -q_1\\
&b_0 = a_2 - q_1 b_1 - q_0 b_2 = 1 + q_1^2 - q_0\\
&r_1 = a_1 - q_1 b_0 - q_0 b_1 = -q_1(1 + q_1^2 - q_0) + q_0 q_1 = -q_1(1 + q_1^2 - 2q_0) = 0\\
&r_0 = a_0 - q_0 b_0 = 1 - q_0(1 + q_1^2 - q_0) = -q_0 q_1^2 + q_0^2 - q_0 + 1 = 0
\end{aligned}} \rightarrow \boxed{\begin{aligned} r_1(q_1, q_0) = 0\\ r_2(q_1, q_0) = 0 \end{aligned}}$$

$$q_1(q_1^2 - 2q_0 + 1) = 0 \rightarrow q_1 = 0 \text{ or } q_1^2 = 2q_0 - 1$$

$$-q_0 q_1^2 + q_0^2 - q_0 + 1 = 0 \rightarrow q_0^2 - q_0 + 1 = 0 \text{ or } q_0^2 = 1$$

i) $q_0 = 1,\ q_1 = \pm 1 \rightarrow b_2 = 1,\ b_1 = \mp 1,\ b_0 = 1$

$$f(x) = (x^2 \pm x + 1)(x^2 \mp x + 1) = (x^2 + x + 1)(x^2 - x + 1)$$

ii) $q_0 = -1,\ q_1 = \pm\sqrt{3}i$ **,which are not solution since** q_1 **is a complex number**

iii) $q_1 = 0,\ q_0 = \dfrac{1 \pm \sqrt{3}i}{2}$ **,which are not solution since** q_0 **is a complex number**

(c) Example 3: 5th order polynomial

$$f(x) = x^5 - x^4 + x^3 - x^2 - x + 1 \rightarrow a_5 = 1,\ a_4 = -1,\ a_3 = 1,\ a_2 = -1,\ a_1 = 1,\ a_0 = 1$$

$$\boxed{\begin{aligned}
b_3 &= a_5 = 1 \\
b_2 &= a_4 - q_1 b_3 = -1 - q_1 \\
b_1 &= a_3 - q_1 b_2 - q_0 b_3 = 1 + q_1(1 + q_1) - q_0 = q_1^2 + q_1 - q_0 + 1 \\
b_0 &= a_2 - q_1 b_1 - q_0 b_2 = -1 - q_1(q_1^2 + q_1 - q_0 + 1) + q_0(1 + q_1) \\
&= -q_1^3 - q_1^2 + q_1 q_0 - q_1 - 1 \\
r_1 &= a_1 - q_1 b_0 - q_0 b_1 = 1 + q_1(q_1^3 + q_1^2 - q_1 q_0 + q_1 + 1) - q_0(q_1^2 + q_1 - q_0 + 1) \\
&= q_1^4 + q_1^3 - 2q_1^2 q_0 + q_1^2 + q_0^2 + q_1 - q_0 + 1 = 0 \\
r_0 &= q_0 q_1^3 + q_0 q_1^2 - q_1 q_0^2 + q_0 q_1 + q_0 + 1 = 0
\end{aligned}}$$

$$\rightarrow \boxed{\begin{aligned} r_1(q_1, q_0) &= 0 \\ r_2(q_1, q_0) &= 0 \end{aligned}}$$

Highly complex, use a numerical method such as the Newton-Raphson method

3. Muller's Method to find one real root: Local quadratic approximation of function

- **Local quadratic approximation of f(x) with given 3-point data such as**

  (x_0, f_0), (x_1, f_1), (x_2, f_2) **where** $f_0 = f(x_0)$, $f_1 = f(x_1)$, $f_2 = f(x_2)$

$$(x_0, f_0) \rightarrow f(x) = a(x - x_0)^2 + b(x - x_0) + f_0$$

$$(x_1, f_1) \rightarrow f_1 = a(x_1 - x_0)^2 + b(x_1 - x_0) + f_0$$

$$(x_2, f_2) \rightarrow f_2 = a(x_2 - x_0)^2 + b(x_2 - x_0) + f_0$$

Using the last two equations, we can calculate coefficients a and b as follows

Let

$$h_1 = x_1 - x_0 \qquad d_1 = f_1 - f_0$$

$$h_2 = x_2 - x_0 \qquad d_2 = f_2 - f_0$$

Then,

$$\begin{aligned} a h_1^2 + b h_1 &= d_1 \\ a h_2^2 + b h_2 &= d_2 \end{aligned} \rightarrow \boxed{a = \frac{d_1 h_2 - d_2 h_1}{h_1^2 h_2 - h_2^2 h_1} \qquad b = \frac{d_1 h_2^2 - d_2 h_1^2}{h_1 h_2^2 - h_2 h_1^2}}$$

- **Approximated roots of local quadratic approximation of f(x)**

$$f(x) = a(x - x_0)^2 + b(x - x_0) + f_0$$

$$= ax^2 + (b - 2ax_0)x + ax_0^2 - bx_0 + f_0$$

$$\rightarrow \boxed{x = \frac{-(b - 2ax_0) \pm \sqrt{(b - 2ax_0)^2 - 4a(x_0^2 - bx_0 + f_0)}}{2a}}$$

Choose the nearest x to x_2 as the approximated solution of the root x_R

Then, repeat the above procedure after the following shifting in three points

$(x_1, f_1) \rightarrow (x_0, f_0)$

$(x_2, f_2) \rightarrow (x_1, f_1)$

$(x_R, f_R) \rightarrow (x_2, f_2)$

4. Bairstow's Method: Newton like method to find 1st or 2nd order quotient of the given polynomial.

(If we find a 2nd order quotient, we can calculate complex pair of roots or two real roots.)

○ **Background Rationale**

- Assume an approximated 2nd order quotient $q(x)$ as the form

$$f(x) = q(x)g(x)$$
$$q(x) = x^2 - rx - s$$

with two roots of $x = \dfrac{r \pm \sqrt{r^2 + 4s}}{2}$ (reals or complex pair)

- Since $q(x) = x^2 - rx - s$ approximates the quotient, there exists the residual such as

$f(x) = q(x)g(x) = (x^2 - rx - s)g(x) + R(x)$,where the residual R(x) is the 1st order and can be calculate using a method similar to Polynomial Deflation.

$$\begin{aligned} R(x) &= (a_1 - q_1 b_0 - q_0 b_1)x + (a_0 - q_0 b_0) \\ &= (a_1 + rb_0 + sb_1)x + (a_0 + sb_0) \end{aligned}$$

- If $R(x) = (a_1 + rb_0 + sb_1)x + (a_0 + sb_0) = 0$, then $q(x)$ is the true quotient and we can find two roots by solving $q(x) = 0$

(Question) How to define r, s in $q(x) = x^2 - rx - s$ to meet $R(x) = 0$ for all x.

Answer: By solving for r, s

$$R_1(r,s) = a_1 + rb_0 + sb_1 = 0$$
$$R_2(r,s) = a_0 + sb_0 = 0$$

However, b_0, b_1 are the functions of r, s , Therefore, we should solve the nonlinear system of equations

$$R_1(r,s)=0$$
$$R_2(r,s)=0$$

(Question) How to solve above nonlinear algebraic equation? Answer: Newton Raphson Method

○ **Newton Raphson Method revisited**

- Definition of the system of nonlinear equations

$\boxed{\mathbf{f}(\mathbf{x})=0, \quad \mathbf{f}\in R^n, \quad \mathbf{x}\in R^n}$, which has n unknowns $\mathbf{x}\in R^n$ and n nonlinear equations $\mathbf{f}\in R^n$

- **Newton Raphson Method**

$$\boxed{\mathbf{x}_{j+1}=\mathbf{x}_j-\mathbf{G}^{-1}\mathbf{f}(\mathbf{x}_j)}$$

$$\frac{d\mathbf{f}}{d\mathbf{x}}=\mathbf{G}=\begin{pmatrix} \frac{df_1}{dx_1} & \frac{df_1}{dx_2} & \cdots & \frac{df_1}{dx_n} \\ \frac{df_2}{dx_1} & \frac{df_2}{dx_2} & \cdots & \frac{df_2}{dx_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{df_3}{dx_1} & \frac{df_n}{dx_2} & \cdots & \frac{df_n}{dx_n} \end{pmatrix}\in R^{n\times n}$$

○ Bairstow's Method

$$R_1(r,s) = a_1 + rb_0 + sb_1 = 0$$
$$R_2(r,s) = a_0 + sb_0 = 0$$

$$\boxed{\begin{aligned} \frac{\partial R_1(r,s)}{\partial r} &= b_0 + r\frac{\partial b_0}{\partial r} + s\frac{\partial b_1}{\partial r} \\ \frac{\partial R_2(r,s)}{\partial r} &= s\frac{\partial b_0}{\partial r} \end{aligned}} \qquad \boxed{\begin{aligned} \frac{\partial R_1(r,s)}{\partial s} &= r\frac{\partial b_0}{\partial s} + b_1 + s\frac{\partial b_1}{\partial s} \\ \frac{\partial R_1(r,s)}{s} &= b_0 + s\frac{\partial b_0}{\partial s} \end{aligned}}$$

Using the Newton-Raphson method, $\boxed{\mathbf{x}_{j+1} = \mathbf{x}_j - \mathbf{G}^{-1}\mathbf{f}(\mathbf{x}_j)}$

$$\boxed{\begin{pmatrix} r_{j+1} \\ s_{j+1} \end{pmatrix} = \begin{pmatrix} r_j \\ s_j \end{pmatrix} - \begin{pmatrix} b_0 + r\frac{\partial b_0}{\partial r} + s\frac{\partial b_1}{\partial r} & r\frac{\partial b_0}{\partial s} + b_1 + s\frac{\partial b_1}{\partial s} \\ s\frac{\partial b_0}{\partial r} & b_0 + s\frac{\partial b_0}{\partial s} \end{pmatrix}^{-1} \begin{pmatrix} a_1 + rb_0 + sb_1 \\ a_0 + sb_0 \end{pmatrix}}$$

For RHS of the equation, use $r = r_j,\ s = s_j,\ b_0 = b_0(r_j, s_j),\ b_1 = b_1(r_j, s_j)$

(Question) How to estimate $\begin{pmatrix} b_0 + r\frac{\partial b_0}{\partial r} + s\frac{\partial b_1}{\partial r} & r\frac{\partial b_0}{\partial s} + b_1 + s\frac{\partial b_1}{\partial s} \\ s\frac{\partial b_0}{\partial r} & b_0 + s\frac{\partial b_0}{\partial s} \end{pmatrix}$**?**

Answer: use the central difference formula

5. **Pseudo code for Bairstow's Method to fine two roots for the polynomial f(x) = 0 and the quotient polynomial after the polynomial deflation using the quadratic quotient q(x) = x*x – r*x –s.**

```
Function Bairstow(n, a, IT_max, epsilon, b, rr, ir, res)
!------------------------------------------------------------------------------------------------------------------
!n:              (input) order of the polynomial (n>2)
!a(0:n):         (input) coefficient of the polynomial
!IT_max:         (input) maximum allowed iteration number
!epsilon:        (input) tolerance in function residual
!b(0:n-1):       (output) coefficient of quotient polynomial after Polynomial Deflation
!rr(1:2):        (input/output) estimation of real part of two roots  (inpu
!ir(1:2) ):      t/output) estimation of imaginary part of two roots
!res(1:2):       (output) residual polynomial coefficient as the form R(x)=r(2)x+r(1)
!
!res0(1:2):      (local) residual due to zero perturbation  (lo
!resp(1:2):      cal) residual due to positive perturbation  (lo
!resm(1:2):      cal) residual due to negative perturbation
!grad(1:2,1:2) (local) gradient estimation using central difference
!------------------------------------------------------------------------------------------------------------------
 if a(n)=0, exit with notice of "polynomial order is less than n"
 if (imag_root(1)+ imag_root(2)) != 0, exit with notice of "roots are not complex pair"
 ! define quadratic quotient of the form  q(x) = x^2 - rx - s
        r =    rr(1) + rr(2)
        s = - rr(1)*rr(2) + ir(1)*ir(2)
 !define small perturbation for central difference formula
       dr= 0.01;
       ds= 0.01;
 !iteration of Newton-Raphson to find the  q(x) = x^2 - rx - s  which reduces the residual near to zero.
        do iter =1, IT_max
            call FUNCTION Poly_Defl_two(n, a, -r,-s, b, res0)
 !central difference formula to calculate gradient for the residual function
            rp = r+dr; call FUNCTION Poly_Defl_two(n, a, -rp,-s, b, resp)
```

```
        rm = r-dr; call FUNCTION Poly_Defl_two(n, a, -rm,-s, b, resm)
        grad(1:2,1) = 0.5*(resp(1:2) - resm(1:2))/dr

        sp = s+ds; call FUNCTION Poly_Defl_two(n, a, -r,-sp, b, resp)  s
        m = s-ds; call FUNCTION Poly_Defl_two(n, a, -r,-sm, b, resm)
        grad(1:2,2) = 0.5*(resp(1:2) - resm(1:2))/ds
!update the quotient polynomial
        (r0;s0)← (r;s)
        (r;s)   ← (r;s) – inv(grad)*(res0(1), res0(2))
!termination condition
        if norm(res0) < epsilon, exit
        if sqrt((r-r0)*(r-r0)+ (s-s0)*(s-s0)) < epsilon, exit
!
      end do
!
      call FUNCTION Poly_Defl_two(n, a, -r,-s, b, res)
      call Function Quadroot(1,-r,-s,r1,r2,i1,i2,nr)
      rr(1) = r1; rr(2)= r2;ir(1)= i1, ir(2)=i2;
!---------------------------------------------------------------------------------------------------------------
End Bairstow
```

6. Pseudo code to find N-roots of f(x) = 0 using Bairstow's Method

---

i) check N (N=1 or N=2)

if N=1, return after finding one real root

if N=2, returen after finding two roots

root_real(1)	**← real part of one root**
root_imag(1)	**← imaginary part of one root**
root_real(2)	**← real part of the other root**
root_imag(2)	**← imaginary part of the other root**

ii) set M= int(N/2)+1, k=0, NR=0

iii) repeat util k=M

k=k+1

(a) calculate the quotients of $f(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + a_{n-3} x^{n-3} + \cdots + a_1 x + a_0$

$$f(x) = q(x)g(x)$$

$$q(x) = x^2 + q_1 x + q_0$$

$$g(x) = b_{n-2} x^{n-2} + b_{n-3} x^{n-3} + \cdots + b_1 x + b_0$$

(b) calculate two roots of $q(x) = x^2 + q_1 x + q_0$

root_real(NR+1)	**← real part of one root**
root_imag(NR+1)	**← imaginary part of one root**
root_real(NR+2)	**← real part of the other root**
root_imag(NR+2)	**← imaginary part of the other root**
NR=NR+2	**← number of roots found**

(c) Check the order of $g(x) = b_{n-2} x^{n-2} + b_{n-3} x^{n-3} + \cdots + b_1 x + b_0$

NR_remained = N-NR

check NR_remained NR_remained=1 f or NR_remained=2)

ind a root when NR_remained=1

find two roots when NR_remained=2

return

end if

(d) redefine $f(x) = a_{n-2}x^{n-2} + a_{n-3}x^{n-3} + \cdots + a_1 x + a_0$

$a_j \leftarrow b_j, \quad j = n-2, n-3, \cdots, 0$

(e) repeat step iii)

7. Library and Packages for root location

- **Excel**
- **Matlab**
- **IMSL**
- **Matlib Libraries**