Computers, Materials & Continua
DOI:10.32604/cmc.2021.018955
Article

Tech Science Press

Computer Geometries for Finding All Real Zeros of Polynomial Equations Simultaneously

Naila Rafiq[1], Saima Akram[2], Mudassir Shams[3,*] and Nazir Ahmad Mir[1]

[1]Department of Mathematics, NUML, Islamabad, 44000, Pakistan
[2]Centre for Advanced Studies in Pure & Applied Mathematics, Bahauddin Zakariya University, Multan
[3]Department of Mathematics and Statistics, Riphah International University I-14, Islamabad, 44000, Pakistan
[*]Corresponding Author: Mudassir Shams. Email: mudassir4shams@gmail.com
Received: 26 March 2021; Accepted: 27 April 2021

Abstract: In this research article, we construct a family of derivative free simultaneous numerical schemes to approximate all real zero of non-linear polynomial equation. We make a comparative analysis of the newly constructed numerical schemes with a well-known existing simultaneous method for determining all the distinct real zeros of polynomial equations using computer algebra system Mat Lab. Lower bound of convergence of simultaneous schemes is calculated using Mathematica. Global convergence property of the numerical schemes is presented by taking random starting initial approximation and their convergence history are graphically presented. Some real life engineering applications along with some higher degree polynomials are considered as numerical test problems to show performance and efficiency of the derivative free family of numerical methods with comparison of an existing method of same order in literature. Local computational order of convergence, CPU time, graph of computational order of convergence and residual error graphs elaborate efficiency, robustness and authentication of the suggested family of numerical methods in its domain.

Keywords: Polynomials; simultaneous iterative methods; random initial guesses; lower bound; local computational order; CAS-mathematica and mat lab

1 Introduction

One of the most primal problem of science and engineering is locating the zeros of polynomial of degree k with arbitrary real coefficient.

$$f(x) = \sum_{i=1}^{k} a_i x^i, \quad a_k \neq 0. \tag{1}$$

Let $\xi_1, \ldots, \xi_k$ denote all the simple zeros of Eq. (1). According to Abel's impossibility theorem [1] "There is no solution in radical to general polynomial with arbitrary co-efficient of degree five or higher" we therefore look toward numerical schemes to approximate zeros of polynomial Eq. (1).

There exist a lot of numerical schemes in literature which approximate single zeros at a time (see, e.g., [2–4]). Here, we consider the following family of numerical schemes [5]:

$$x^{(n+1)} = y^{(n)} - \frac{f(x^{(n)})}{f'(x^{(n)})}\left(\frac{(f(x^{(n)}))^2 f(y^{(n)}) + 2f(x^{(n)})f(y^{(n)}) + \beta(f(y^{(n)}))^3}{(f(x^{(n)}))^3}\right), \tag{2}$$

where $y^{(n)} = x^{(n)} - \dfrac{f(x^{(n)})}{f'(x^{(n)})}$ and $\beta \in \mathrm{R}$.

The numerical scheme Eq. (2) approximate single zero of Eq. (1) at a time.

Beside these single roots finding methods, mathematicians and engineers are interested in simultaneous numerical schemes which approximate all roots simultaneously. More detail on simultaneous methods, their global convergence and parallel implementation on computer algebra system (CAS) and stability are found in [6,7] and reference cite there in [8–14].

Therefore, the main aim of this research article is to construct a derivative free numerical scheme which approximates all real zero of Eq. (1). Using CAS-Mathematica, we find the lower bound of convergence to verify convergence order theoretically. Computational order of convergence [15] and convergence history for random initial approximations are graphed to show the efficiency and performance of numerical schemes as compared to other existing methods of same order. Log of residual graph, graphs of computational order of convergence and local computational order of convergence [16] support the global convergence behavior of our newly constructed numerical scheme for estimating all real zeros of Eq. (1).

2 Construction of Numerical Scheme

Corresponding to numerical schemes

$$y^{(n)} = x^{(n)} - \frac{f(x^{(n)})}{f'(x^{(n)})}, \tag{3}$$

for approximating all zeros of Eq. (1), the numerical method is

$$x_i^{(n+1)} = x_i^{(n)} - \frac{f(x_i^{(n)})}{\prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - x_j^{(n)})}. \quad (i = 1, \ldots, k) \tag{4}$$

This numerical method is well known Weierstrass method [17] for approximating all zeros of polynomial Eq. (1) having local quadratic convergence. Using in analogical way as above, we convert Eq. (2) into simultaneous method as:

$$x_i^{(n+1)} = y_i^{(n)} - \frac{f(x_i^{(n)})}{\prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - x_j^{(n)})}\left(\frac{(f(x_i^{(n)}))^2 f(y_i^{(n)}) + 2f(x_i^{(n)})f(y_i^{(n)}) + \beta(f(y_i^{(n)}))^3}{(f(x_i^{(n)}))^3}\right), \tag{5}$$

where $y_i^{(n)} = x_i^{(n)} - \dfrac{f(x_i^{(n)})}{\prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - x_j^{(n)})}$ and $\beta \in \mathrm{R}$. Thus, we have constructed here, a new derivative free family of numerical schemes (abbreviated as MD).

Nedzhibov et al. [18] in 2005, present the following cubic convergence derivative free family of simultaneous numerical schemes (abbreviated as ND) as:

$$x_i^{(n+1)} = x_i^{(n)} - \frac{f(x_i^{(n)})}{\prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - x_j^{(n)})} \left(1 + \frac{f(y_i^{(n)})}{f(x_i^{(n)}) - 2\beta f(y_i^{(n)})}\right). \tag{6}$$

2.1 Convergence Analysis

Here, we prove the convergence order of the suggested derivative free family of numerical schemes.

Theorem: *Let algebraic polynomial Eq. (1) has k number of simple zeros* $\xi_1, \ldots, \xi_k$ *and for sufficiently close initial guesses* $x_1^{(0)}, \ldots, x_k^{(0)}$ *of the zeros, then for arbitrary real parameter* β *,the numerical scheme MD has third order convergence.*

Proof: let $\in_i = x_i^{(n)} - \xi_i$, $\in_i' = y_i^{(n)} - \xi_i$, $\in_i'' = x_i^{(n+1)} - \xi_i$ be the errors in $x_i^{(n)}$, $y_i^{(n)}$,$x_i^{(n+1)}$ respectively and

$$Ð_i = \prod_{\substack{j \neq i \\ j=1}}^{k} \left(\frac{x_i^{(n)} - \zeta_j}{x_i^{(n)} - x_j^{(n)}}\right). \tag{7}$$

Iterative schemes Eq. (5) can be written as:

$$x_i^{(n+1)} = y_i^{(n)} - w_i(x_i^{(n)})(Ģ_i + 2Ģ_i^2 + \beta Ģ_i^3), \tag{8}$$

where $Ģ_i = \dfrac{f(y_i^{(n)})}{f(x_i^{(n)})}$ and $w_i(x_i^{(n)}) = \dfrac{f(x_i^{(n)})}{\prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - x_j^{(n)})}$.

If we express Eq. (1) as $f(x_i^{(n)}) = (x_1^{(n)} - \xi_1)\ldots(x_k^{(n)} - \xi_k) = \in_i \prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - \xi_j)$, then we have:

$$f(y_i^{(n)}) = (y_1^{(n)} - \xi_1)\ldots(y_i^{(n)} - \xi_i). \tag{9}$$

Substitution in Eq. (5), we have:

$$f(y_i^{(n)}) = \prod_{j=1}^{k} (x_i^{(n)} - w_i(x_i^{(n)} - \xi_j) = (\xi_i - w_i(x_i^{(n)})) \prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - w_i(x_i^{(n)}) - \xi_j)$$

$$= \in_i (1 - Đ_i) \prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - w_i(x_i^{(n)}) - \xi_j).$$

Then, for $Ģ_i$, we have:

$$Ģ_i = \frac{\in_i (1 - Đ_i) \prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - w_i(x_i^{(n)}) - \xi_j)}{\in_i \prod_{\substack{j \neq i \\ j=1}}^{k} (x_i^{(n)} - w_i(x_i^{(n)}) - \xi_j)} = (1 - Đ_i)\, R_i, \tag{10}$$

where $R_i = \frac{x_i^{(n)} - w_i(x_i^{(n)}) - \xi_j}{x_i^{(n)} - w_i(x_i^{(n)}) - \xi_j}$. Thus, Eq. (5) become:

$$\in_i'' = \in_i' - \in_i Đ_i (Ģ_i + 2Ģ_i^2 + \beta Ģ_i^3) \tag{11}$$

$$\in_i'' = \in_i - \in_i Đ_i - \in_i Đ_i (Ģ_i + 2Ģ_i^2 + \beta Ģ_i^3),$$

$$= \in_i (1 - Đ_i)\, ((1 - Đ_i R_i) - 2Đ_i (1 - Đ_i)\, R_i^2 - \beta (1 - Đ_i)^2 R_i^3).$$

The following relation holds true:

$$\prod_{\substack{j \neq i \\ j=1}}^{k} \frac{(x_i^{(n)} - \zeta_j)}{(x_i^{(n)} - x_j)} - 1 = \sum_{t \neq i}^{k} \frac{\varepsilon_t}{x_i^{(n)} - x_t^{(n)}} \prod_{j \neq i}^{t-1} \frac{(x_i^{(n)} - \zeta_t)}{(x_i^{(n)} - x_j^{(n)})}. \tag{12}$$

If we assume that, absolute values of all errors are of the same order, say $|\varepsilon_i| = |\varepsilon_j| = O(|\varepsilon|)$, then

$$Đ_i - 1 = \sum_{t \neq i}^{k} \frac{\varepsilon_t}{x_i^{(n)} - x_t^{(n)}} \prod_{j \neq i}^{t-1} \frac{(x_i^{(n)} - \zeta_t)}{(x_i^{(n)} - x_j^{(n)})} = O(|\varepsilon|) \tag{13}$$

holds. Using $Đ_i R_i = \prod_{\substack{j \neq i \\ j=1}}^{k} \left(\frac{x_i^{(n)} - w_i(x_i^{(n)})\xi_j}{x_i^{(n)} - x_j^{(n)}} \right)$, we have:

$$Đ_i R_i - 1 = \sum_{t \neq i}^{k} \frac{x_t^{(n)} - w_i(x_i^{(n)}) - \xi_t}{x_i^{(n)} - x_t^{(n)}} \prod_{j \neq i}^{t-1} \frac{x_i^{(n)} - w_i(x_i^{(n)}) - \xi_t}{(x_i^{(n)} - x_j^{(n)})},$$

$$= \sum_{t \neq i}^{k} \frac{\in_t - \in_i Đ_i}{x_i^{(n)} - x_t^{(n)}} \prod_{j \neq i}^{t-1} \frac{x_i^{(n)} - w_i(x_i^{(n)}) - \xi_t}{(x_i^{(n)} - x_j^{(n)})} = O(|\in|).$$

Thus,

$$\in_i'' = \in_i \sum_{t \neq i}^{k} \frac{\in_t}{x_i^{(n)} - x_t} \prod_{j \neq i}^{t-1} \frac{(x_i^{(n)} - \zeta_t)}{(x_i^{(n)} - x_j^{(n)})} (O(|\in|) + 2O(|\in|^2)\,\mathrm{R}_i + O(|\in|^2)\,\mathrm{R}_i^3 \beta) = O(\in^3). \tag{14}$$

Hence, the theorem is proved.

2.2 *Using CAS-Mathematica for Finding Lower Bound of Convergence*

Consider

$$f(x) = (x - \varphi_1)(x - \varphi_2)(x - \varphi_3) \tag{15}$$

and the first component $H(x^{(n)})$ of iterative scheme Eq. (5) to find zeros of Eq. (15), simultaneously. In order to find lower bound of convergence, we have to express the differential of an operator $H(x^{(n)})$ in terms of their partial derivate of its component as $H_i(\mathrm{x})$:

$$\begin{array}{cccccc}
\frac{\partial \mathrm{H}_1(\mathbf{x})}{\partial x_1} & \frac{\partial \mathrm{H}_1(\mathbf{x})}{\partial x_2} & \frac{\partial \mathrm{H}_1(\mathbf{x})}{\partial x_3} & & & \\
\frac{\partial^2 \mathrm{H}_1(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 \mathrm{H}_1(\mathbf{x})}{\partial x_1 \partial x_2} & \frac{\partial^2 \mathrm{H}_1(\mathbf{x})}{\partial x_2^2} & \frac{\partial^2 \mathrm{H}_1(\mathbf{x})}{\partial x_2 \partial x_3} & & \\
\frac{\partial^3 \mathrm{H}_1(\mathbf{x})}{\partial x_1^3} & \frac{\partial^3 \mathrm{H}_1(\mathbf{x})}{\partial x_1^2 \partial x_2} & \frac{\partial^3 \mathrm{H}_1(\mathbf{x})}{\partial x_1 \partial x_2^2} & \frac{\partial^3 \mathrm{H}_1(\mathbf{x})}{\partial x_2^3} & \frac{\partial^3 \mathrm{H}_1(\mathbf{x})}{\partial x_2^2 \partial x_3} & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \cdots
\end{array}$$

and to so on.

The lower bound of the convergence is obtained until the first non-zero element of row is found zero (see [19]). The Mathematica program is given for each of the considered method as:

- MD method

$$\mathrm{H}_1(x_1, x_2, x_3) := \frac{(\mathbf{x})^2 \prod_{\substack{j \neq i \\ j=1}}^{n} (\mathbf{x}_i - \mathbf{x}_j)}{\mathbf{x} \prod_{\substack{j \neq i \\ j=1}}^{n} (x_i - x_j) + f(\mathbf{x})},$$

In[1] := D[H$_1$[x$_1$, x$_2$, x$_3$], x$_1$]/.{x$_1 \to \varphi_1$, x$_2 \to \varphi_2$, x$_3 \to \varphi_3$}
Out[1] := 0
In[2] := D[H$_1$[x$_1$, x$_2$, x$_3$], x$_2$]/.{x$_1 \to \varphi_1$, x$_2 \to \varphi_2$, x$_3 \to \varphi_3$}]
Out[2] := 0
In[2] := D[H$_1$[x$_1$, x$_2$, x$_3$], x$_2$]/.{x$_1 \to \varphi_1$, x$_2 \to \varphi_2$, x$_3 \to \varphi_3$}]
Out[2] := 0
In[34] := Simplify[D[H$_1$[x$_1$, x$_2$, x$_3$], x$_1$, x$_1$, x$_1$, x$_2$]/.{x$_1 \to \varphi_1$, x$_2 \to \varphi_2$, x$_3 \to \varphi_3$}]

$$\text{Out[34]} := \frac{6(-1+\varphi_3)}{\varphi_1^2 \varphi_3^3}.$$

- ND method

$$\mathrm{H}_1(x_1, x_2, x_3) := \mathbf{x} - \frac{f(\mathbf{x})}{\prod_{\substack{j=1 \\ j \neq i}}^{n} (x_i - x_j)} \left(1 + \frac{f(\mathbf{y})}{f(\mathbf{x}) - 2\beta f(\mathbf{y})}\right),$$

where $\mathbf{y} = x - \frac{f(\mathbf{x})}{\prod_{\substack{j=1 \\ j \neq i}}^{n} (x_i - x_j)}$ and $\beta \in \mathrm{R}$.

In[1] := D[H$_1$[x$_1$, x$_2$, x$_3$], x$_1$]/.{x$_1 \to \varphi_1$, x$_2 \to \varphi_2$, x$_3 \to \varphi_3$}]
Out[1] := 0

$\vdots$

In[39] := Simplify[D[H$_1$[r$_1$, r$_2$, r$_3$], r$_1$, r$_3$, r$_1$, r$_2$]/.{x$_1 \to \varphi_1$, x$_2 \to \varphi_2$, x$_3 \to \varphi_3$}]

$$\text{Out[39]}: = -\frac{6(2-2\varphi_3)}{\varphi_1 {\varphi_3}^2}$$

3 Numerical Results

Here, some numerical examples from [20,21] are considered with and without random initial approximation to estimate all real zeros of polynomial equation of higher degree. All the computations are performed using Mat Lab R@2011 with 64 digits floating point arithmetic. We take $\in = 10^{-30}$ as tolerance and use the following stopping criteria

$$e_i = |x_i^{(n+1)} - x_i^{(n)}| < \in,$$

where e_i represents the absolute error. We compare our iterative schemes MD with ND of the same convergence order. In all numerical calculations, we take $\beta = 2$.

Example 1: Consider a stirred tank reactor (SCRT) in which two items A and R are feed in reactor at Q and q-Q rate respectively. A complex reaction

$a + r \to b$; $b + r \to c$; $c + r \to r$; $c + r \to e$, develops the following function

$$h_c \frac{2.98(x + 2.25)}{(x + 1.45)(x + 2.85)^2(x + 4.35)} = 1. \tag{16}$$

We obtained an algebraic polynomial equation of degree 4 by taking $\hbar_c = 0$ in Eq. (16)

$$f_1(x) = x^4 + 11.5x^3 + 47.49x^2 + 83.06325x + 51.32366875 = 0$$

with exact roots:

$$\zeta_1 = -1.45, \quad \zeta_2 = -2.85, \quad \zeta_3 = -2.85, \quad \zeta_4 = -4.45$$

Convergence history and computational order of convergence graph (Figs. 1, 4, 7 and 10) of numerical schemes ND, MD are obtained by taking the following random initial guessed valued in our computer program i.e.,

$$X_1 = [0.032601; 0.5612; 0.88187; 0.66918],$$

where $X_1 = [x_i^{(0)}, i = 1, \ldots, 4]$. Using random initial guessed value X_1, the iterative scheme MD converges to exact zeros after 100 iteration by consuming 16.848368 s CPU time while ND converges after 155 iterations and consumes 37.249147 s.

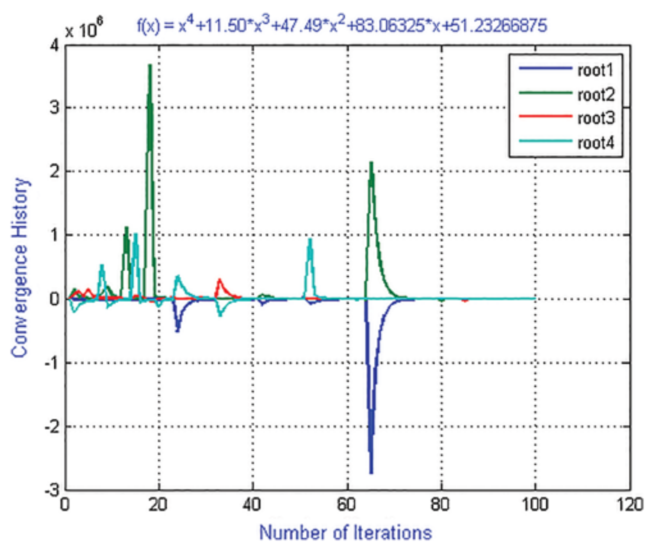

Figure 1: Shows convergence history of numerical scheme MD for polynomial equation $f_1(x)$

Convergence rates increase by taking the following initial guessed value:

$$x_1^{(0)} = -1.0, \quad x_2^{(0)} = -1.1, \quad x_3^{(0)} = -2.2, \quad x_4^{(0)} = -3.9.$$

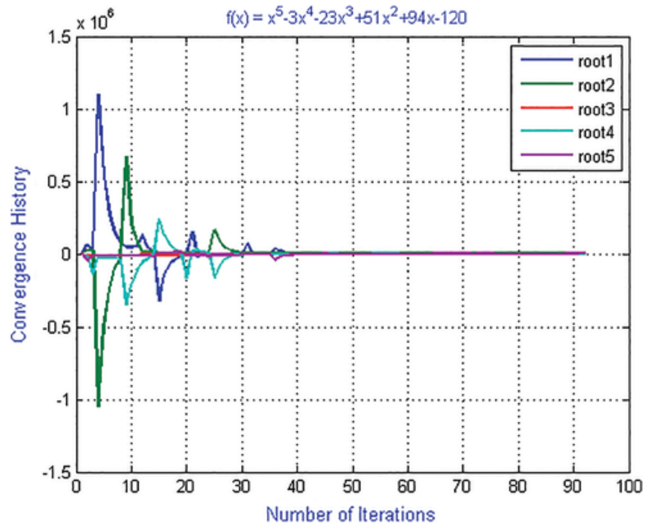

Figure 2: Shows convergence history of numerical scheme MD for polynomial equation $f_2(x)$

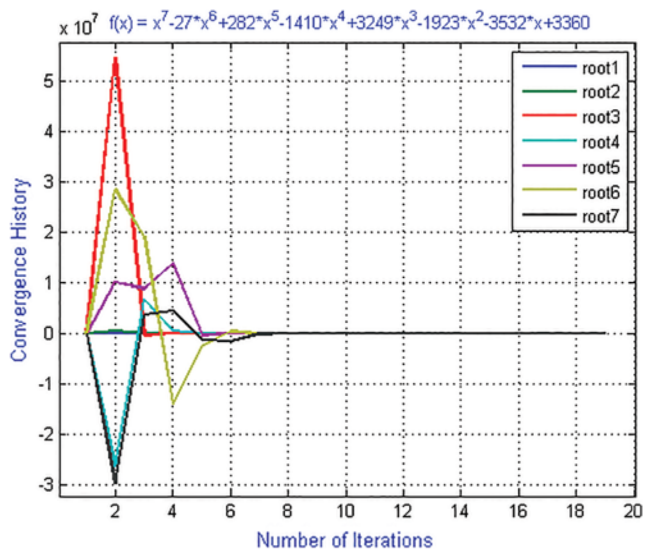

Figure 3: Shows convergence history of numerical scheme MD for polynomial equation $f_3(x)$

The numerical results are presented in Tabs. 1–6. In all Tabs. 1–6, CO presents convergence order, CPU, presents CPU time and local computational order of convergence (LCOC) by σ.

Example 2: Consider

$$f_2(x) = x^5 - 3x^4 - 23x^3 + 51x^2 + 94x - 120 \tag{17}$$

with exact roots:

$\zeta_1 = 1, \quad \zeta_2 = -2, \quad \zeta_3 = 3, \quad \zeta_4 = -4, \quad \zeta_5 = 5.$

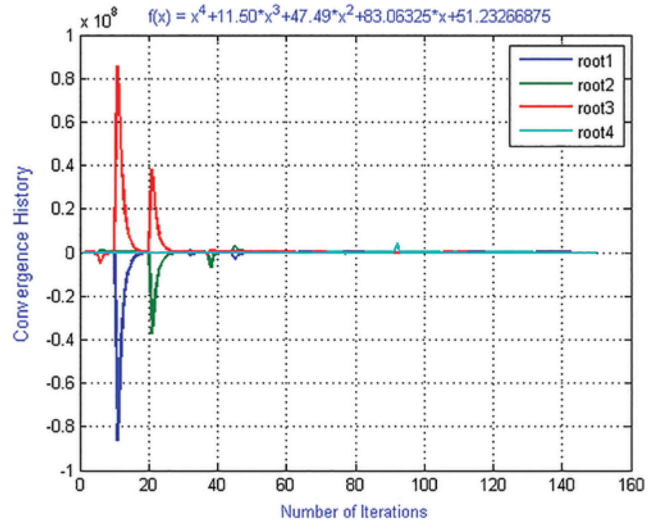

Figure 4: Shows convergence history of numerical scheme ND for polynomial equation $f_1(x)$

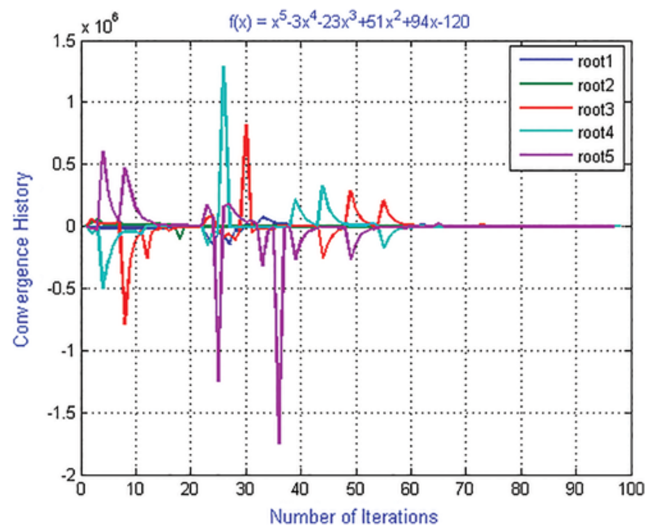

Figure 5: Shows convergence history of numerical scheme ND for polynomial equation $f_2(x)$

For convergence history and computational order of convergence graph (Figs. 2, 5, 8 and 11) of numerical schemes ND, MD, we used the following initial guessed valued in our computer program i.e.,

$X_2 = [0.81472; 0.90579; 0.12699; 0.91338; 0.63236]$

where $X_2 = [x_i^{(0)}, i = 1, \ldots, 5]$. Using random initial guessed values X_2, the iterative scheme MD converges to exact zeros after 92 iteration by consuming 27.056098 s CPU time while ND converges after 98 iterations and consumes 31.444833 s.

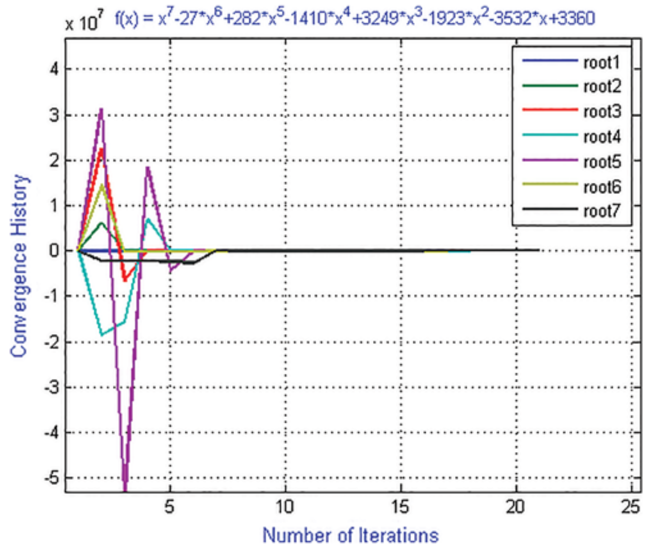

Figure 6: Shows convergence history of numerical scheme ND for polynomial equation $f_3(x)$

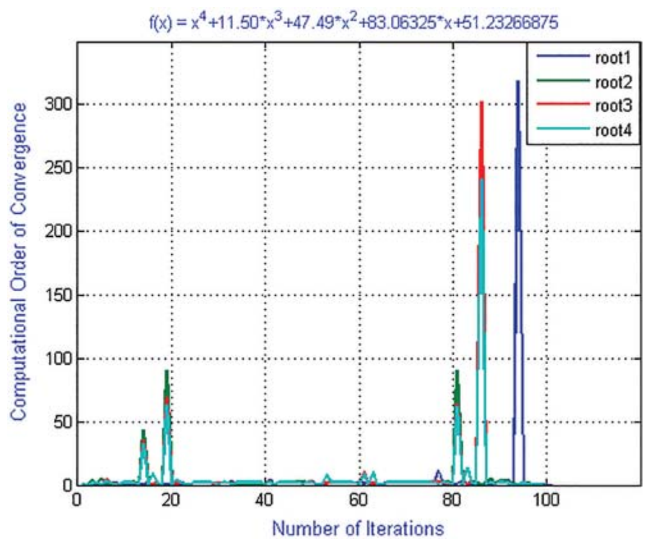

Figure 7: Shows computational order of convergence of numerical scheme MD for polynomial equation $f_1(x)$

Convergence rates increase by taking the following initial guessed value:

$x_1{}^{(0)} = 0.9, \quad x_2{}^{(0)} = -1.9, \quad x_3^{(0)} = 2.9, \quad x_4^{(0)} = -3.9, \quad x_5^{(0)} = 4.9$

Example 3: Consider

$$f_3(x) = x^7 - 27x^6 + 282x^5 - 1410x^4 + 3249x^3 - 1923x^2 - 3532x + 3360 \tag{18}$$

with exact roots:

$\zeta_1 = -1, \quad \zeta_2 = 1, \quad \zeta_3 = 3, \quad \zeta_4 = 4, \quad \zeta_5 = 7, \quad \zeta_6 = 5, \quad \zeta_7 = 8.$

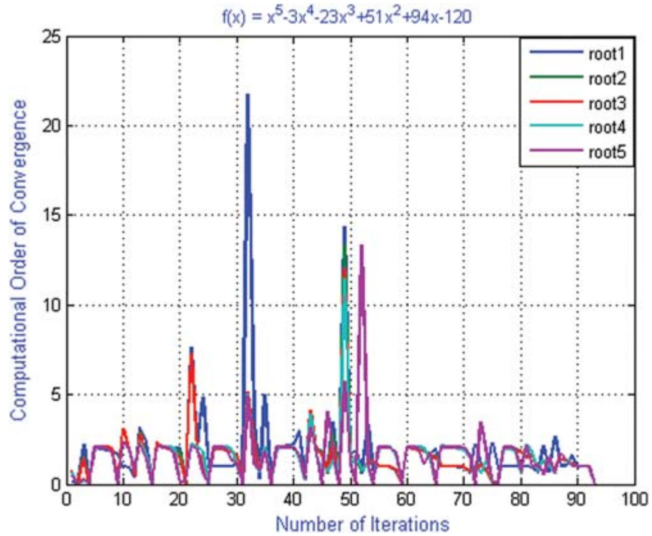

Figure 8: Shows computational order of convergence of numerical scheme MD for polynomial equation $f_2(x)$

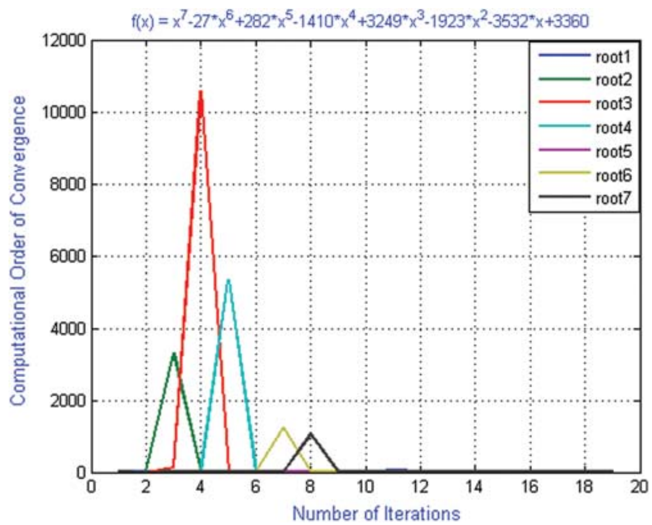

Figure 9: Shows computational order of convergence of numerical scheme MD for polynomial equation $f_3(x)$

For convergence history and computational order of convergence graph (Figs. 3, 6, 9 and 12) of numerical schemes ND, MD are obtained by taking the following initial guessed valued in our computer program i.e.,

$X_3 = [0.77029; 0.35022; 0.66201; 0.41616; 0.84193; 0.83193; 0.83292; 0.25644]$,

where $X_3 = [x_i^{(0)}, i = 1, \ldots, 7]$. Using random initial guessed value X_3, the iterative scheme MD converges to exact zero after 19 iterations by consuming 13.7714 s CPU time while ND converges after 23 iterations and consumes 18.120062 s.

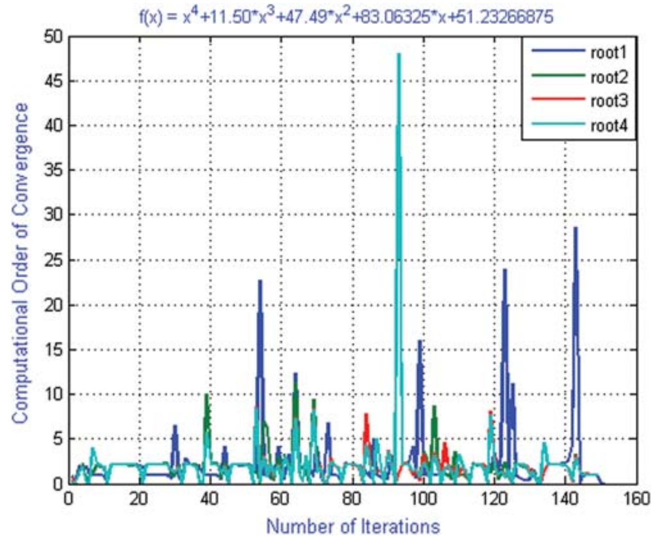

Figure 10: Shows computational order of convergence of numerical scheme ND for polynomial equation $f_1(x)$

Table 1: Simultaneous determination of all zeros of polynomial $f_1(x)$

$x_1{}^{(0)} = -1.0,\ x_2{}^{(0)} = -1.1,\ x_3^{(0)} = -2.2,\ x_4^{(0)} = -3.9$						
Method	CO	PU	$e_1^{(7)}$	$e_2^{(7)}$	$e_3^{(7)}$	$e_4^{(7)}$
ND	3	0.047	8.3e-18	1.8e-17	1.3e-5	1.9e-5
MD	3	0.032	2.5 e-6	2.6e-6	3.0e-32	1.5e-28

Table 2: Local computational order of convergence for polynomial $f_1(x)$

$x_1{}^{(0)} = -1.0, x_2{}^{(0)} = -1.1, x_3^{(0)} = -2.2, x_4^{(0)} = -3.9$					
Method	PU	$\sigma_1^{(6)}$	$\sigma_2^{(6)}$	$\sigma_3^{(6)}$	$\sigma_4^{(6)}$
ND	0.047	3.12	2.93	2.53	2.14
MD	0.032	2.94	2.98	3.01	3.15

Table 3: Simultaneous determination of all zeros of polynomial $f_2(x)$

$x_1{}^{(0)} = 0.9,\ x_2{}^{(0)} = -1.9,\ x_3^{(0)} = 2.9,\ x_4^{(0)} = -3.9,\ x_5^{(0)} = 4.9$							
Method	CO	PU	$e_1^{(3)}$	$e_2^{(3)}$	$e_3^{(3)}$	$e_4^{(3)}$	$e_5^{(3)}$
ND	3	0.057	9.0e-22	1.4e-21	5.4e-24	1.0e-24	1.8e-22
MD	3	0.035	3.1 e-26	3.1e-26	2.0e-29	1.2e-31	2.3e-26

Table 4: Local computational order of convergence for polynomial $f_2(x)$

$x_1{}^{(0)}=0.9,\ x_2{}^{(0)}=-1.9,\ x_3^{(0)}=2.9,\ x_4^{(0)}=-3.9,\ x_5^{(0)}=4.9$						
Method	PU	$\sigma_1^{(2)}$	$\sigma_2^{(2)}$	$\sigma_3^{(2)}$	$\sigma_4^{(2)}$	$\sigma_5^{(2)}$
ND	0.057	2.81	2.41	2.71	2.64	1.95
MD	0.035	2.95	3.15	2.69	3.24	3.06

Table 5: Simultaneous determination of all zeros of polynomial $f_3(x)$

$x_1{}^{(0)}=-0.9,\ x_2{}^{(0)}=0.9,\ x_3^{(0)}=2.5,\ x_4^{(0)}=3.9,\ x_5^{(0)}=6.9,\ x_6^{(0)}=4.5,\ x_7^{(0)}=7.9$									
Method	CO	PU	$e_1^{(5)}$	$e_2^{(5)}$	$e_3^{(5)}$	$e_4^{(5)}$	$e_5^{(5)}$	$e_5^{(5)}$	$e_5^{(5)}$
ND	3	0.761	1.9e-34	6.3e-34	1.2e-33	5.4e-32	4.7e-29	2.2e-31	1.5e-28
MD	3	0.407	0.0	0.0	3.2e-61	0.0	0.0	2.8e-61	0.0

Table 6: Local computational order of convergence for polynomial $f_3(x)$

$x_1{}^{(0)}=-0.9,\ x_2{}^{(0)}=0.9,\ x_3^{(0)}=2.5,\ x_4^{(0)}=3.9,\ x_5^{(0)}=6.9,\ x_6^{(0)}=4.5,\ x_7^{(0)}=7.9$							
Method	PU	$\sigma_1^{(4)}$	$\sigma_2^{(4)}$	$\sigma_3^{(4)}$	$\sigma_4^{(4)}$	$\sigma_5^{(4)}$	$\sigma_5^{(4)}$	$\sigma_5^{(4)}$
ND	0.761	2.91	3.02	3.11	2.67	2.74	2.91	2.85
MD	0.407	3.01	3.21	3.45	2.97	3.00	3.18	3.42

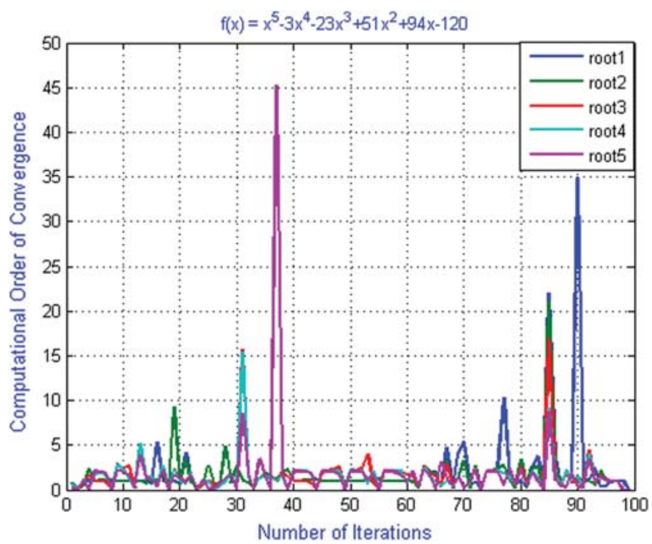

Figure 11: Shows computational order of convergence of numerical scheme ND for polynomial equation $f_2(x)$

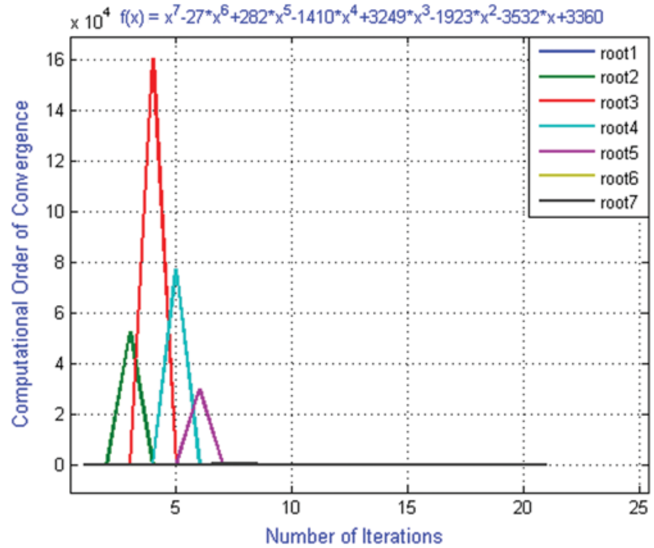

Figure 12: Shows computational order of convergence of numerical scheme ND for polynomial equation $f_3(x)$

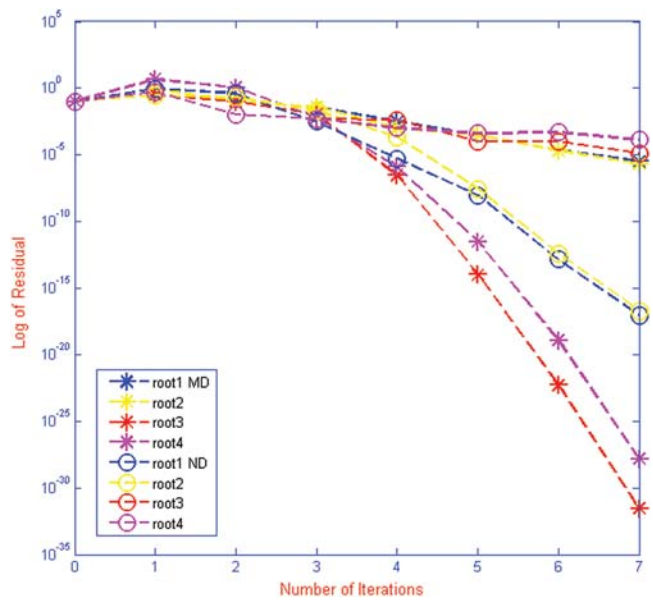

Figure 13: Shows residual fall for iterative method MD and ND for polynomial $f_1(x)$ respectively

Convergence rates increase by taking the following initial guessed value:

$x_1{}^{(0)} = -0.9, \quad x_2{}^{(0)} = 0.9, \quad x_3^{(0)} = 2.5, \quad x_4^{(0)} = 3.9, \quad x_5^{(0)} = 6.9, \quad x_6^{(0)} = 4.5, \quad x_7^{(0)} = 7.9.$

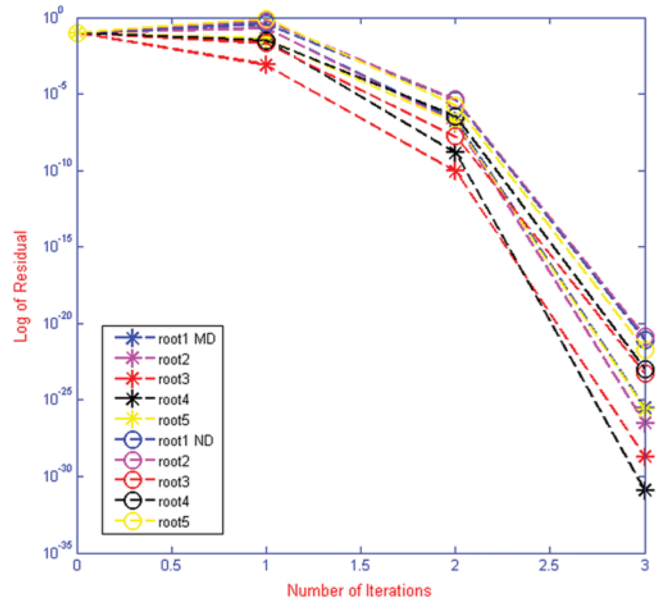

Figure 14: Shows residual fall for iterative method MD and ND for polynomial $f_2(x)$ respectively

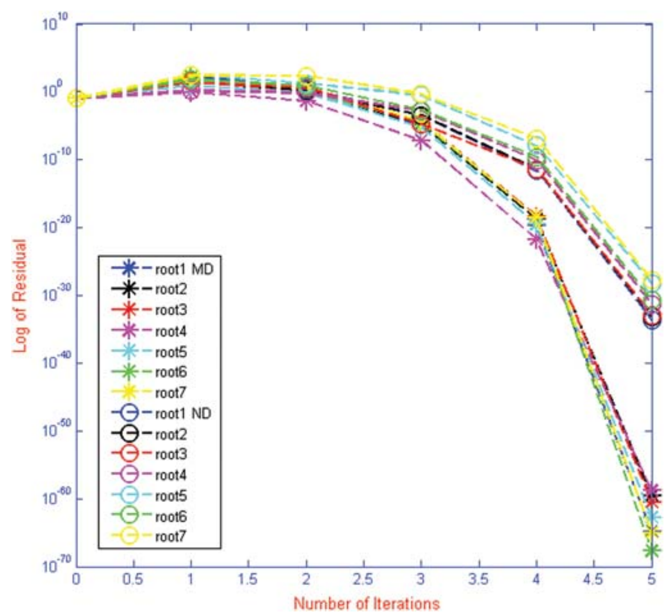

Figure 15: Shows residual fall for iterative method MD and ND for polynomial $f_3(x)$ respectively

4 Conclusions

Here, we have developed a family of derivative free method for approximating all real zeros of polynomial. Lower bound of convergence of iterative methods MD and ND are calculated using CAS-Mathematica. Using Mat Lab, we graph convergence history and computational order of convergence. From Tabs. 1–6 and Figs. 1–15, we observe that our method MD is much better in terms of convergence history, computational order of convergence, numerical results, log of residual and local computational order of convergence as compared to ND method.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. I. Rosen, "Niels hendrik abel and equations of the fifth degree," *American Mathematical Monthly*, vol. 102, no. 6, pp. 495–505, 1995.

[2] O. S. Solaiman and I. Hashim, "An iterative scheme of arbitrary odd order and its basins of attraction for nonlinear systems," *Computers, Materials & Continua*, vol. 66, no. 2, pp. 1427–1444, 2021.

[3] S. A. Sariman and I. Hashim, "New optimal newton-householder methods for solving nonlinear equations and their dynamics," *Computers, Materials & Continua*, vol. 65, no. 1, pp. 69–85, 2020.

[4] C. Chun, "Some variants of king's fourth-order family of methods for nonlinear equations," *Applied Mathematics and Computation*, vol. 190, no. 1, pp. 57–62, 2007.

[5] N. Rafiq, S. Akram, N. A. Mir and M. Shams, "Study of dynamical behavior and stability of iterative methods for nonlinear equation with application in engineering," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–20, 2020.

[6] Y. M. Chu, N. Rafiq, M. Shams, S. Akram, N. A. Mir *et al.*, "Computer methodologies for the comparison of some efficient derivative free simultaneous iterative methods for finding roots of non-linear equations," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 275–290, 2021.

[7] P. D. Proinov and S. I. Ivanov, "On the convergence of halley's method for multiple polynomial zeros," *Mediterranean Journal of Mathematics*, vol. 12, no. 2, pp. 555–572, 2015.

[8] O. S. Solaiman, S. A. A. Karim and I. Hashim, "Dynamical comparison of several third-order iterative methods for nonlinear equations," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 1951–1962, 2021.

[9] N. A. Mir, M. Shams, N. Rafiq, S. Akram and M. Rizwan, "Derivative free iterative simultaneous method for finding distinct roots of polynomial equation," *Alexandria Engineering Journal*, vol. 59, no. 3, pp. 1629–1636, 2020.

[10] P. D. Proinov and M. T. Vasileva, "On a family of weierstrass-type root-finding methods with accelerated convergence," *Applied Mathematics and Computation*, vol. 273, pp. 957–968, 2016.

[11] M. Shams, N. A. Mir, N. Rafiq, A. O. Almatroud and S. A. Akram, "On dynamics of iterative technique for nonlinear equation with application in engineering," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–17, 2020.

[12] S. C. Chapra, *Applied Numerical Methods with MATLAB® for Engineers and Scientists*, 6th ed. New York: McGraw Hill, 2010.

[13] V. K. Kyncheva, V. V. Yotov and S. I. Ivanov, "Convergence of newton, halley and chebyshev iterative methods as methods for simultaneous determination of multiple polynomial zeros," *Applied Numerical Mathematics*, vol. 112, pp. 146–154, 2017.

[14] S. I. Cholakov, "Local and semi local convergence of Wang-Zheng's method for simultaneous finding polynomial zeros," *Symmetry*, vol. 11, no. 6, pp. 1–16, 2019.

[15] F. Soleymani and D. K. R. Babajee, "Computing multiple zeros using a class of quartically convergent methods," *Alexandria Engineering Journal*, vol. 52, no. 3, pp. 531–541, 2013.

[16] S. Akram, M. Shams, N. Rafiq and N. A. Mir, "On the stability of weierstrass type method with king's correction for finding all roots of non-linear function with engineering application," *Applied Mathematical Sciences*, vol. 14, no. 10, pp. 461–473, 2020.

[17] N. A. Mir, M. Shams, S. Akram and R. Ahmed, "On the family of simultaneous method for finding distinct as well as multiple roots of nonlinear equation," *Punjab University Journal of Mathematics*, vol. 52, no. 6, pp. 31–44, 2020.

[18] G. H. Nedzhibov, "Convergence of the modified inverse weierstrass method for simultaneous approximation of polynomial zeros," *Communication in Numerical Analysis*, vol. 16, no.1, pp. 74–80, 2016.
[19] M. Shams, N. Rafiq, B. Ahmad and N. A. Mir, "Inverse numerical iterative technique for finding all roots of non-linear equations with engineering applications," *Journal of Mathematics*, vol. 2021, no. 1, pp. 1–10, 2021.
[20] M. Shams, N. Rafiq, N. A. Mir, B. Ahmad, S. Abbasi *et al.*, "On computer implementation for comparison of inverse numerical schemes," *Computer System Science and Engineering*, vol. 36, no. 3, pp. 493–507, 2021.
[21] M. R. Farmer, "*Computing the zeros of polynomials using the divide and conquer approach*," Ph.D. Thesis, Department of Computer Science and Information Systems, Birkbeck, University of London, 2014.