On Efficient Iterative Numerical Methods for Simultaneous Determination of all Roots of Non-Linear Function

[1]Mudassir Shams, [1]Nazir Ahmad Mir, [2]Naila Rafiq

[1]Department of Mathematics and Statistics, Riphah International University I-14, Islamabad 44000, Pakistan

[2]Department of Mathematics, NUML, Islamabad, Pakistan

shams6262@gmail.com, nazir.ahmad@riphah.edu.pk, nrafique@numl.edu.pk

Abstract: We construct a family of two-step optimal fourth order iterative methods for finding single root of non-linear equations. We generalize these methods to simultaneous iterative methods for determining all the distinct as well as multiple roots of single variable non-linear equations. Convergence analysis is present for both cases to show that the order of convergence is four in case of single root finding method and is twelve for simultaneous determination of all roots of non-linear equation. The computational cost, Basin of attraction, efficiency, log of residual and numerical test examples shows, the newly constructed methods are more efficient as compared to the existing methods in literature.

Keywords: Single roots; Distinct roots; Multiple roots; Optimal order; Non-Linear equation; Iterative methods; Simultaneous Methods; Basin of attraction; Computational Efficiency

Introduction

To solve non-linear equation

$$f(x) = 0 \tag{1}$$

is the oldest problem of science in general and in mathematics in particular. These non-linear equations have a diverse application in many areas of science and engineering. Several iterative methods have been used to find the roots of non-linear equation (1), using different techniques such as Decomposition methods, Homotopy analysis method, Variation iteration methods and modification in Newton Raphson method etc. All these methods are used to approximate one root at a time. But mathematician are also interested in simultaneous finding of all roots of non-linear equation because simultaneous iterative methods are very popular due to their wider region of convergence, are more stable as compared to single root finding methods and implemented for parallel computing as well. More detailed on single as well as simultaneous determination of all roots can be found in [1-1011-13, 15, 22, 24-30] and reference cited there in. The most famous of single root finding method is the classical Newton -Raphson method:

$$y_i = x_i - \frac{f(x_i)}{f^{'}(x_i)}, (i = 1, 2, ...) \tag{2}$$

Methods (2) is optimal having efficiency 1.43. Using Weierstrass' Correction [11]

$$\frac{f(x)}{f^{'}(x)} = w(x_i) = \frac{f(x_i)}{\prod\limits_{\substack{j \neq i \\ j=1}}^{n}(x_i - x_j)}, \tag{3}$$

in (2) then, we get classical Weierstrass - Dochive methods to approximate all roots of (1) is:

$$y_i = x_i - \frac{f(x_i)}{\prod\limits_{\substack{j \neq i \\ j=1}}^{n}(x_i - x_j)}, \tag{4}$$

The main aim of this paper is to construct family of optimal fourth order method and then generalize it into simultaneous iterative methods for finding all roots of non-linear equation (1).

Constructions of Method and Convergence Analysis

Here, we present some optimal fourth iterative methods for finding roots of non-linear equation (1) are:

Jarrat et al. [16] suggest the following optimal fourth order methods (abbreviated as JM):

$$\begin{cases} y_i = x_i - \frac{2}{3}\frac{f(x_i)}{f'(x_i)}, \\ z_i = x_i - \frac{f(x_i)}{f'(x_i)}\left(1 - \frac{2}{3}\frac{f'(y_i)-f'(x_i)}{3f'(y_i)-f'(x_i)}\right). \end{cases} \tag{5}$$

Siyyam et al. [17] present the folloeing optimal fourth order iterative method (abbreviated as SM):

$$\begin{cases} y_i = x_i - \frac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \frac{2f(x_i)G(u)+(y_i-x_i)(1+G(u))f'(x_i)}{2f'(x_i)}. \end{cases} \tag{6}$$

where $u = \frac{f(y_i)}{f(x_i)}$ and $G(u) = 1 + 2u + 4u^2 + u^3$.

O. Y. Ababneh et al. [18] present the following fourth order iterative method (abbreviated as YM):

$$\begin{cases} y_i = x_i - \frac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \frac{2f(y_i)}{f'(x_i)} + \left(\frac{f(y_i)(f(x_i))^2}{f'(x_i)(f(x_i+f(y_i))^2}\right) + \left(-\frac{f(y_i)f(x_i)(2f(x_i+f(y_i))}{2f(x_i)(f(x_i+f(y_i))^2}\right)\left(\frac{f(y_i)}{f'(x_i)}\right)^2. \end{cases} \tag{7}$$

Here, we propose the following iteration scheme

$$\begin{cases} y_i = x_i - \frac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \frac{f(y_i)}{f'(x_i)+A\Psi(u)}, \end{cases} \tag{8}$$

where $u = \frac{f(y_i)}{f(x_i)}$ and $\Psi(u)$ is real valued function and find later.

For the iteration scheme (8), we have the following convergence theorem as using CAS Maple 18, we find the error equation of the iteration scheme defined by (8).

Theorem *1: Let $\zeta \in I$ be a simple root of a sufficiently differential function $f : I \subseteq R \rightarrow R$ is an open interval I. If x_0 is sufficiently close to ζ and $\Psi(u)$ be a real valued function satisfying $\Psi(0) = 0$, $\Psi'(0) = -\frac{1}{2}$ and A=4 then the convergence order of the family of iterative method (8) is four and satisfying the error equation*

$$e_{i+1} = (10c_2^3 - 8c_2c_3 + 3c_4)e_i^4 + O(e_i^5) \tag{9}$$

where $\quad c_m = \dfrac{f^m(\zeta)}{m! f'(\zeta)}; m \geq 2.$

Proof *Let* $\quad \zeta \quad$ *be a simple root of* $f \quad$ *and* $\quad x_i = \zeta + e_i \quad$ *By Taylor's series expansion of*

$f(x_i) \quad around \quad x \ \blacksquare \ \zeta, \quad taking \quad f(\zeta) = 0, \quad we \ get:$

$$f(x_i) = f'(\zeta)(e_i + c_2 e_i^2 + c_3 e_i^3 + c_4 e_i^4 + O(e_i^5)) \tag{10}$$

and

$$f'(x_i) = f'(\zeta)(1 + 2c_2 e_i + 3c_3 e_i^2 + 4c_4 e_i^3 + O(e_i^4)). \tag{11}$$

Dividing (10) by (11), we have:

$$\frac{f(x_i)}{f'(x_i)} = e_i + c_2 e_i^2 + (2c_2^2 - 2c_3)e_i^3 + (7c_2 c_3 - 3c_4 - 4c_2^3)e_i^4 + O(e_i^5) \tag{12}$$

using (12) in first step of (8), we have:

$$y_i = \zeta + c_2 e_i^2 + (2c_3 - 2c_2^2)e_i^3 + O(e_i^4). \tag{13}$$

Thus, using Taylor series, we have

$$f(y_i) = f'(\zeta)(c_2 e_i^2 + 2(c_3 - c_2^2)e_i^3 + (3c_4 - 7c_2 c_3 + 5c_2^3)e_i^4 + O(e_i^5)). \tag{14}$$

Dividing (14) by (11), we have:

$$u = \frac{f(y_i)}{f(x_i)} = c_2 e_i + (-3c_2^2 + 2c_3)e_i^2 + (3c_2^3 - 3c_2 c_3)e_i^3. \tag{15}$$

We expand $\quad \Psi(u) \quad$ *by Taylor series about 0 to obtain:*

$$\Psi(u) = \Psi(0) + \Psi'(0)u + \frac{\Psi''(0)u^2}{2!} + ... \tag{16}$$

$$\Psi(u) = \Psi(0) + \Psi'(0)c_2 e_i + (-3\Psi'(0)c_2^2 + 2\Psi'(0)c_3)e_i^2 + ... \tag{17}$$

$$f'(x_i) + A\Psi(u) = 1 + A\Psi(0) + (A\Psi(0)c_2 + 2c_2)e_i + D_1 e_i^2 + D_2 e_i^3 + O(e_i^4). \tag{18}$$

where $D_1 = -3A\Psi'(0)c_2^2 + 2\Psi'(0)c_3 + 3c_3$ *and* $D_2 = 3A\Psi'(0)c_2^3 - 3A\Psi'(0)c_2 c_3 + 4c_4$

$$h_1 = \frac{f(y_i)}{f'(x_i) + A\Psi(u)},$$

$$= \frac{c_2 e_i^2}{D_3} + \frac{1}{D_3^2}(-2c_2^2 + 2D_3 c_2^2 + 2D_3 c_3 - A\Psi'(0)c_2^2)e_i^3 + O(e_i^4). \tag{19}$$

where $D_3 = A\Psi(0) + 1.$

$$e_{i+1} = y_i - h_1 = (c_2 - \frac{c_2}{D_3})e_i^2 +$$

$$(-2c_2^2 + 2c_3 + \frac{2c_2^2}{D_3^2} + \frac{2c_2^2}{D_3} - \frac{2c_3}{D_3} + \frac{2c_2^2 \Psi'(0)}{D_3})e_i^3 + O(e_i^4). \tag{20}$$

Putting $\Psi(0) = 0, \quad \Psi'(0) = -\frac{1}{2} \quad$ *and A=4 in (20), we have:*

$$e_{i+1} = (10c_2^3 - 8c_2 c_3 + 3c_4)e_i^4 + O(e_i^5). \tag{21}$$

Hence proves fourth order convergence.

The Concrete Fourth order Methods

We now construct some concrete forms of the family of methods describe by algorithm (8). Let us take the function $\Psi(u)$ defined by

Table 1: Concrete forms of methods		
S.No	☺	$\Psi(u) = \left(\frac{f(y_i,u)}{f(x_i,u)}\right),$
1	▲	$\Psi(u) = -\frac{1}{2}u$
2	▲	$\Psi(u) = 1 - \frac{1-\frac{1}{2}u}{1+u^2};$
3	▲	$\Psi(u) = -\frac{1}{2}u - u^{☺},$
4	▲	$\Psi(u) = -\frac{1}{2}u - \frac{u^{☺}}{1-u},$

where $u = \frac{f(y_i)}{f(x_i)}$ satisfying the condition $\Psi(0) = 0$, $\Psi'(0) = -\frac{1}{2}$ of the theorem 1 and choose $\alpha = 2$. Therefore, we get following four new two-step fourth order method:

Method -1(NMS-1)

$$\begin{cases} y_i = x_i - \dfrac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \dfrac{f(y_i)}{f'(x_i)-2u}. \end{cases} \tag{22}$$

Method-2(NMS-2)

$$\begin{cases} y_i = x_i - \dfrac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \dfrac{f(y_i)}{f'(x_i)+4\left(1+\dfrac{-1-\frac{1}{2}u}{1+u^2}\right)}. \end{cases} \tag{23}$$

Method-3(NMS-3)

$$\begin{cases} y_i = x_i - \dfrac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \dfrac{f(y_i)}{f'(x_i)+4\left(u^2-\frac{1}{2}u\right)}. \end{cases} \tag{24}$$

Method-4(MNS-4)

$$\begin{cases} y_i = x_i - \dfrac{f(x_i)}{f'(x_i)}, \\ z_i = y_i - \dfrac{f(y_i)}{f'(x_i)+4(-\frac{1}{2}u+\frac{u^2}{1+u})}, \end{cases} \tag{25}$$

where $u = \frac{f(y_i)}{f(x_i)}$ in NMS1-NMS-4.

Complex Dynamical Study of Iterative Methods

Here, we discuss the dynamical study of iterative methods (MNS-1-MNS-4, JM, SM, YM). By choosing suitable initial guess we observe all iterative methods converge. Here, we investigate the region from where we take the initial estimates to achieve the root of non-linear equation. Actually we numerically approximate the domain of attractions of the roots as a qualitative measure, how the iterative methods depend on the choice of initial estimations. To answer these questions on the dynamical behavior of the iterative methods, we investigate the dynamics of the methods (MNS-1-MNS-4) and compare it JM, SM, YM. Let us recall some basic concepts of this study in the background contexture of complex dynamics. For more details on the dynamical behavior of the iterative methods one can consult [19-21].Taking a rational function $\Re_f : C \rightarrow C$, where C denotes the complex plane, the orbit $t_0 \in C$ is defines a set such as $orb(t) = \{t_0, \Re_f(t_0), \Re_f^2(t_0), ..., \Re_f^m(t_0), ...\}$ .The convergence $orb(z) \rightarrow z*$ is understood in the sense if $\lim_{k \rightarrow \infty} R^k(z) = z*$ exist. A point $z_0 \in \mathbb{C}$ is known as periodic with minimal period m if $\Re^m(t_0) = t_0$ hold where m is smallest positive integer. A periodic point for m=1 is known as fixed, attracting if $\left|R^{k'}(z)\right| < 1$, repelling if $\left|R^{k'}(z)\right| > 1$ and neutral otherwise. An attracting point $t^* \in C$ defines basin of attraction, $\Re(t^*)$, as the set of starting points whose orbit tends to s^*. The closure of the set of its repelling periodic points of a rational map is known as the Julia set denoted by J(R) and its complement is the Fatou set denoted by F(R). The iterative methods when they applied to find the roots of (1), provides the rational map $\Re_f$. But we are interested in the basins of attraction of the roots of the non-linear function (1). Fatou set F(Q) contains the basins of attraction of different roots is a well-known fact. In general the Julia set is a fractal and rational map behave as unstable in this region. For the dynamical and graphically point of view, we take 2000×2000 grid of square $[-2.5, 2.5]^2 \in \mathbb{C}$. To each root of (1), we assign a color to which the corresponding orbit of the iterative methods starts and converges to a fixed point. Take color map as lines. We use $\left|f(s_i)\right| < 10^{-3}$ as a stopping criteria and maximum number of iteration is taken as 25. We mark a dark blue point if the orbit of the iterative methods does not converges to root after 25 iterations which means it has a distance greater than 10^{-3} to any root. Different color is used for different roots. Iterative methods have different basins of attraction distinguished by their colors. In basins, brightness in color represents the number of iterations to achieve the root of (1). Brightness in color means less number of iterations steps. Note that darkest blue regions denote the lake of convergence to any root of (1). Finally, in table 1, we present Elapsed time of basins of attraction correspond to iterative maps (MNS-1-MNS-4, JM, SM, YM) using tic-toc command in code using MATLAB (R2011b). Figure 1 and 2 shows the basins of attraction of iterative methods (MNS-1-MNS-4, JM, SM and YM) for non-linear function $f_1(x), f_2(x), f_3(x)$ and $f_4(x)$ respectively. By observing the basins of attraction, we can

easily judge the stability of iterative methods (MNS-1-MNS-4, JM, SM, YM). Elapsed time, divergent regions and brightness in color presents that MNS-1-MNS-4 is better than JM, SM, and YM.
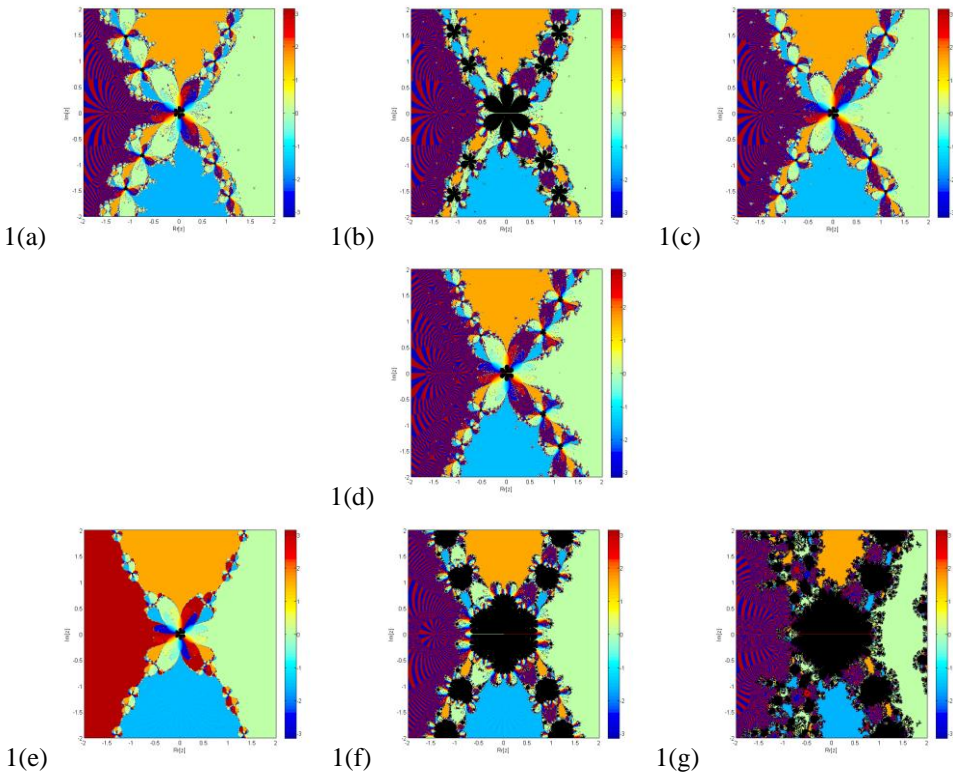

1(a)   1(b)   1(c)

1(d)

1(e)   1(f)   1(g)

Figure 1(a-g), presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear function $f_1(x) = (\sin(x))^2 - x^2 + 1$.

Table 2: Elapsed Time for Basin of attraction						
f_1 $(x) = (\sin(x))^2 - x^2 + 1$						
NSM-1	NSM-2	NSM-3	NSM-4	JM	SM	YM
19.880778	27.543028	28.366321	35.043862	21.593686	62.496179	61.235432

Figure 1, presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear function $f_1(x)$. In figure 1(a-g), brightness of color in basins shows less number of iterations for convergence of methods NSM-1-NSM-4 as compared to JM, SM, YM. From table 2, it is clear that method NSM-1-NSM-4 take less elapsed time as compared to JM, SM and YM, shows clearly the dominance efficiency of the iterative methods NSM-1-NSM-4 over JM, SM, YM.
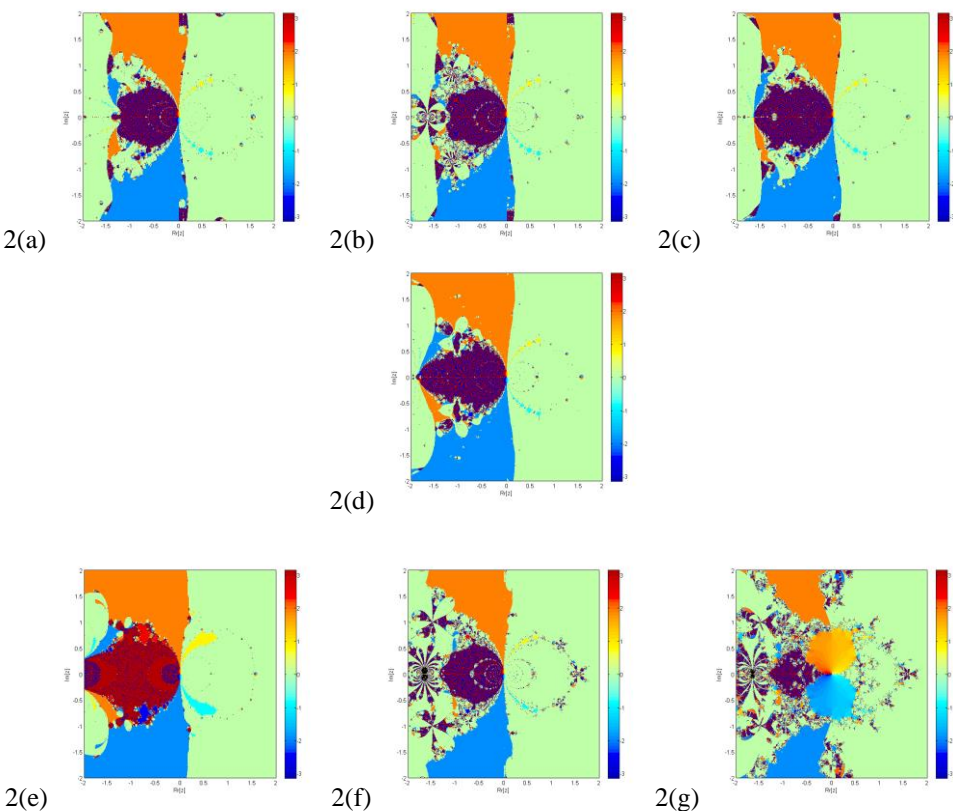
2(a)　　　　　　　　2(b)　　　　　　　　2(c)

2(d)

2(e)　　　　　　　　2(f)　　　　　　　　2(g)

Figure 2(a-g), presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear

function $f_2(x) = \sqrt{x^2 + 2} - \sin\left(\dfrac{\pi}{x}\right) + \dfrac{1}{x^4 + 1} - \sqrt{3} - \dfrac{1}{17}$.

Table 3: Elapsed Time for Basin of attraction						
$f_2(x) = \sqrt{x^2 + 2} - \sin\left(\dfrac{\pi}{x}\right) + \dfrac{1}{x^4+1} - \sqrt{3} - \dfrac{1}{17}$						
NSM-1	NSM-2	NSM-3	NSM-4	JM	SM	YM
48.159561	66.811359	68.643910	86.366701	59.531932	146.227106	153.185194

Figure 2, presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear function $f_2(x)$. In figure 1(a-g), brightness of color in basins shows less number of iterations for convergence of methods NSM-1-NSM-4 as compared to JM, SM, YM. From table 3, it is clear that method NSM-1-NSM-4 take less elapsed time as compared to JM, SM and YM, shows clearly the dominance efficiency of the iterative methods NSM-1-NSM-4 over JM, SM, YM.
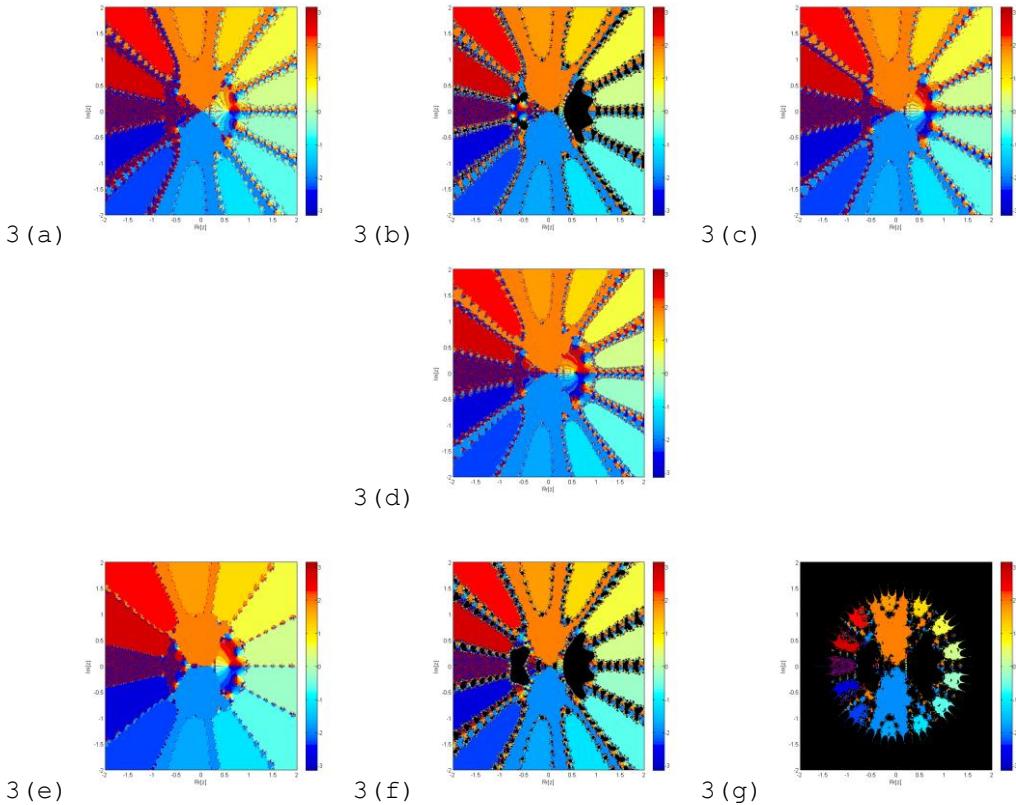
3(a)   3(b)   3(c)

3(d)

3(e)   3(f)   3(g)

Figure3 (a-g), presents the Basins of attraction of Methods (NSM-1-NSM-4, JM,SM,YM) for non-linear function

$$f_3(x) = \cos(x) + \sin(2x)\sqrt{1 - x^2} + \sin(x^2) + x^{14} + x^3 + \frac{1}{2x}.$$

Table 4: Elapsed Time for Basin of attraction						
$f_3(x) = \cos(x) + \sin(2x)\sqrt{1 - x^2} + \sin(x^2) + x^{14} + x^3 + \frac{1}{2x}$						
NSM-1	NSM-2	NSM-3	NSM-4	JM	SM	YM
99.456842	139.463391	138.463391	177.559344	107.150882	302.821147	276.286735

Figure 3, presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear function $f_3(x)$. In figure 1(a-g), brightness of color in basins shows less number of iterations for convergence of methods NSM-1-NSM-4 as compared to JM, SM, YM. From table 4, it is clear that method NSM-1-NSM-4 take less elapsed time as compared to JM, SM and YM, shows clearly the dominance efficiency of the iterative methods NSM-1-NSM-4 over JM, SM, YM.
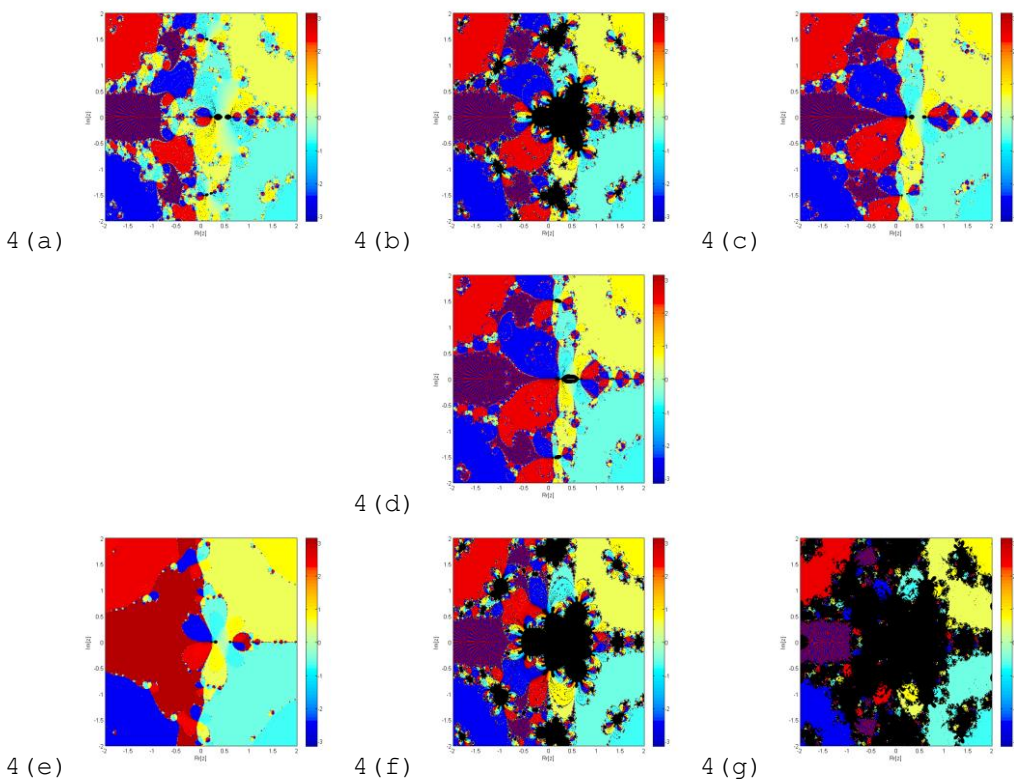
4(a)　　　　　　4(b)　　　　　　4(c)

4(d)

4(e)　　　　　　4(f)　　　　　　4(g)

Figure4 (a-g), presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear function

$$f_4(x) = xe^{x^2} - (\sin(x))^2 + 3\cos(x) + 5.$$

Table 5: Elapsed Time for Basin of attraction						
f_4 ⋂⋃⊟xe^{x^2} ⊘⋂sin ⋂⋃³ ⊟cos ⋂⋃⊟5						
NSM-1	NSM-2	NSM-3	NSM-4	JM	SM	YM
53.410032	70.954946	72.913443	92.048316	58.813235	154.527156	138.266051

Figure 4, presents the Basins of attraction of Methods (NSM-1-NSM-4, JM, SM, YM) for non-linear function $f_4(x)$. In figure 1(a-g), brightness of color in basins shows less number of iterations for convergence of methods NSM-1-NSM-4 as compared to JM, SM, YM. From table 5, it is clear that method NSM-1-NSM-4 take less elapsed time as compared to JM, SM and YM, shows clearly the dominance efficiency of the iterative methods NSM-1-NSM-4 over JM, SM, YM.

Generalization to Simultaneous Iterative Methods

Suppose, the non-linear equation (1) has n roots. Then $f(x)$ and $f'(x)$ can be approximated as:

$$f(x) = \prod_{j=1}^{n}\left(x - x_j\right) \text{ and } f'(x) = \sum_{k=1}^{n}\prod_{\substack{j \neq k \\ j=1}}^{n}\left(x - x_j\right). \tag{26}$$

This implies,

$$\frac{f'(x)}{f(x)} = \sum_{j=1}^{n}\frac{1}{(x - x_j)} = \frac{1}{\frac{1}{x - x_i} - \sum_{\substack{j=1 \\ j \neq i}}^{n}\frac{1}{(x - x_j)}}. \tag{27}$$

This gives,

$$y(x_i) = x_i - \frac{1}{\frac{1}{N(x_i)} - \sum_{\substack{j=1 \\ j \neq i}}^{n}\frac{1}{(x - x_j)}}, \text{ where } N(x_i) = \frac{f(x_i)}{f'(x_i)}. \tag{28}$$

Now from (27), an approximation of $\frac{f(x_i)}{f'(x_i)}$ is formed by replacing x_j with $k_t(x_j^{**})$ as follows:

$$\frac{f(x_i)}{f'(x_i)} = \frac{1}{\frac{1}{N(x_i)} - \sum_{\substack{j=1 \\ j \neq i}}^{n}\frac{1}{(x_i - k_t(x_j^{**}))}}, \text{ where } k_t(x_j^{**}) = z_i, (t = 1,...,4). \tag{29}$$

Using (29) in (2), we have:

$$y_i = x_i - \frac{1}{\frac{1}{N(x_i)} - \sum_{\substack{j=1 \\ j \neq i}}^{n}\frac{1}{(x_i - k_t(x_j^{**}))}}.(t = 1,...4) \tag{30}$$

In case of multiple roots

$$y_i = x_i - \frac{\sigma_i}{\frac{1}{N(x_i)} - \sum_{\substack{j=1 \\ j \neq i}}^{n}\frac{\sigma_j}{(x_i - k_t(x_j^{**}))}}.(t = 1,...4) \tag{31}$$

where

$$k_1(x_j^{**}) = z_j = y_j - \frac{f(y_j)}{f'(x_j) - 2u1}, \tag{32}$$

$$k_2(x_j^{**}) = z_j = y_j - \frac{f(y_j)\left(1 + (u1)^2\right)}{f'(x_j)\left(1 + (u1)^2\right) + 4\left((u1)^2 - \frac{1}{2}(u1)\right)}, \tag{33}$$

$$k_3(x_j^{**}) = z_j = y_j - \frac{f(y_j)}{f'(x_j) + 4\left((u1)^2 - \frac{1}{2}u1\right)}, \tag{34}$$

$$k_4(x_j^{**}) = z_j = y_j - \frac{f(y_j)}{f'(x_j) + 2\left(\frac{u1^2 - u1}{(1+(u1))}\right)}, \tag{35}$$

where $\quad y_i = x_i - \frac{f(x_i)}{f'(x_i)}$ and $u1 = \frac{f(y_j)}{f(x_j)}$.

Using Correction $\quad k_t\mathbf{\Omega}_j^{**}$ where t=1,...,4 in (31), we get the following four simultaneous iterative methods for extracting all distinct as well as multiple roots of non-linear equation (1).

$$\begin{cases} y_i = x_i - \dfrac{\sigma_i}{\frac{1}{N(x_i)} - \displaystyle\sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{\sigma_j}{(x_i - k_t(x_j^{**}))}}, \\[2em] z_i = y_i - \dfrac{\sigma_i}{\frac{1}{N(y_i)} - \displaystyle\sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{\sigma_j}{(y_i - y_j)}}, \end{cases} \tag{36}$$

where $\quad t = 1,...4 \quad$ and $\quad j = 1,...n.$

abbreviated as NMSM-1,NMSM-2,NMSM-3 and NMSM-4.

Convergence Analysis

In this section, the convergence analysis of a family of simultaneous methods (NMSM-1, NMSM-2, NMSM-3 and NMSM-4) given in form of the following theorem. Obviously, convergence for the method (NMSM-1, NMSM-2, NMSM-3 and NMSM-4) will follow from the convergence of the method (NMSM-1, NMSM-2, NMSM-3 and NMSM-4) from theorem (2) when the multiplicities of the roots are simple.

Theorem: *Let $\quad \gamma_1, \gamma_2, \ldots, \gamma_n$ be the n number of simple roots of non-linear equation (1). If $x_1^{**}, \quad x_2^{**}, \quad x_3^{**}, \ldots, \quad x_n^{**}$ be the initial approximations of the roots respectively and sufficiently close to actual roots, the order of convergence of methods (*NMSM-1, NMSM-2, NMSM-3 and NMSM-4*) equals twelve.*

Proof Let

$$\varepsilon_i = x_i - \zeta_i, \; \varepsilon_i' = y_i - \zeta_i \text{ and } \varepsilon_i'' = z_i - \zeta_i \tag{37}$$

be the errors in x_i and y_i approximations respectively. Considering (NMSM-1, NMSM-2, NMSM-3 and NMSM-4), which is

$$y_i = x_i - \frac{\sigma_i}{\frac{\sigma_i}{N(x_i)} - \displaystyle\sum_{\substack{j \neq i \\ j=1}}^{n} \frac{\sigma_j}{(x_i - k_t(x_j^{**}))}}, (t = 1,...,4). \tag{38}$$

where

$$N(x_i) = \frac{f(x_i)}{f'(x_i)}.$$

Then, obviously for distinct roots:

$$\frac{1}{N(x_i)} = \frac{f'(x_i)}{f(x_i)} = \sum_{j=1}^{n} \frac{1}{(x_i - \zeta_j)} = \frac{1}{(x_i - \zeta_i)} + \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{1}{(x_i - \zeta_j)}. \tag{39}$$

Thus, for multiple roots we have from (NMSM-1, NMSM-2, NMSM-3 and NMSM-4):

$$y_i = x_i - \frac{\sigma_i}{\frac{\sigma_i}{(x_i - \zeta_i)} + \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{\sigma_j}{(x_i - \zeta_j)} - \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{\sigma_j}{(x_i - k_t(x_j^{**}))}}, \tag{40}$$

$$y_i - \zeta_i = x_i - \zeta_i - \frac{\sigma_i}{\frac{\sigma_i}{(x_i - \zeta_i)} + \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{\sigma_j (x_i - k_t(x_j^{**}) - x_i + \zeta_j)}{(x_i - \zeta_j)(x_i - k_t(x_j^{**}))}}, \tag{41}$$

$$\varepsilon_i' = \varepsilon_i - \frac{\sigma_i}{\frac{\sigma_i}{\varepsilon_i} + \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{-\sigma_j(x_j^{*} - \zeta_j)}{(x_i - \zeta_j)(x_i - k_t(x_j^{**}))}}, \tag{42}$$

$$= \varepsilon_i - \frac{\sigma_i \varepsilon_i}{\sigma_i + \varepsilon_i \sum_{\substack{j \neq i \\ j=1}}^{n} \frac{-\sigma_j \left(k_t(x_j^{**}) - \zeta_j\right)}{\left(x_i - \zeta_j\right)\left(x_i - k_t(x_j^{**})\right)}}, \tag{43}$$

$$= \varepsilon_i - \frac{\sigma_i . \varepsilon_i}{\sigma_i + \varepsilon_i \sum_{\substack{j \neq i \\ j=1}}^{n} E_i \varepsilon_j^2}, \tag{44}$$

where $k_t(x_j^{**}) - \zeta_j = \varepsilon_j^4$ from (9) and $E_i = \frac{-\sigma_j}{(x_i - \zeta_j)(x_i - k_t(x_j^{**}))}$.

Thus,

$$\varepsilon_i' = \frac{\varepsilon_i^2 \sum_{\substack{j \neq i \\ j=1}}^{n} E_i \varepsilon_j^4}{\sigma_i + \varepsilon_i \sum_{\substack{j \neq i \\ j=1}}^{n} E_i \varepsilon_j^2}. \tag{45}$$

If it is assumed that absolute values of all errors ε_j $(j = 1, 2, 3, ...)$ are of the same order as, say $|\varepsilon_j| = O|\varepsilon|$, then from (45), we have:

$$\varepsilon_i' = O(\varepsilon)^6. \tag{46}$$

Now consider second step of method (NMSM-1, NMSM-2, NMSM-3 and NMSM-4) and using $z_i \approx \zeta_i + \varepsilon_i^{**}$, $y_i \approx \zeta_i + \varepsilon_i'$ in the following form:

$$\varepsilon_i^{''} = \varepsilon_i^{'} - \left(\frac{\alpha_i}{\frac{\alpha_i}{N(y_i)} - \sum\limits_{\substack{j \neq i \\ j=1}}^{n} \frac{\alpha_j}{(y_i - y_j)}} \right). \tag{47}$$

Now, from 2nd-step of (NMSM-1, NMSM-2, NMSM-3 and NMSM-4) , we have:

$$\varepsilon_i^{''} = \varepsilon_i^{'} - \frac{\alpha_i}{\frac{\alpha_i}{\varepsilon_i^{'}} + \sum\limits_{\substack{j \neq i \\ j=1}}^{n} \frac{\alpha_j}{(y_i - \zeta_j)} - \sum\limits_{\substack{j \neq i \\ j=1}}^{n} \frac{\alpha_j}{(y_i - y_j)}} \tag{48}$$

$$= \varepsilon_i^{'} - \frac{\alpha_i \varepsilon_i^{'}}{\alpha_i + \varepsilon_i^{'} \left(\sum\limits_{\substack{j \neq i \\ j=1}}^{n} \frac{-\alpha_j (y_j - \zeta_j)}{(y_i - \zeta_j)(y_i - y_j)} \right)}, \tag{49}$$

$$= \frac{(\varepsilon_i^{'})^2 \left(\sum\limits_{\substack{j \neq i \\ j=1}}^{n} \varepsilon_j^{'} S_i \right)}{\alpha_i + \varepsilon_i^{'} \sum\limits_{\substack{j \neq i \\ j=1}}^{n} \varepsilon_j^{'} S_i}, \quad \text{where } S_i = \frac{-\sigma_j}{(y_i - \zeta_j)(y_i - y_j)}.$$

Since, $\varepsilon_i^{'} = O(\varepsilon)^6$ from (46), thus,

$$\varepsilon_i^{''} = O\left(\left(\varepsilon \right)^6 \right)^2 = O\left(\varepsilon \right)^{12}. \tag{50}$$

Thus from (50), $\varepsilon_i^{'} = O(\varepsilon)^{12}$, shows convergence order of method (NMSM-1, NMSM-2, NMSM-3 and NMSM-4) are twelve. Hence prove the theorem.

Computational Aspect

Here, we compare the computational efficiency of the Midrog Petkovic method [8] and the new method(NMSM-1, NMSM-2, NMSM-3, NMSM-4). As presented in [8], the efficiency of an iterative method can be estimated using the efficiency index given by

$$EL(m) = \frac{\log \mathbf{r}}{\mathbf{D}}, \tag{51}$$

where **D** is the computational cost and **r** is the order of convergence of the iterative method. Using arithmetic operation per iteration with certain weight depending on the execution time of operation to evaluate the computational cost **D** . The weights used for division, multiplication and addition plus subtraction are w $_d$, w $_m$, w $_{AS}$ respectively. For a given polynomial of degree m and n roots, the number of division, multiplication addition and

subtraction per iteration for all roots are denoted by D_m, M_m and AS_m. The cost of computation can be calculated as:

$$\mathbf{D} = \mathbf{D}(m) = w_{as}AS_m + w_m M_m + w_d D_m, \tag{52}$$

thus (51) becomes:

$$EL(m) = \frac{\log \mathbf{r}}{w_{as}AS_m + w_m M_m + w_d D_m} \tag{53}$$

Considering the number of operations of a complex polynomial with real and complex roots reduce to operation of real arithmetic, given in Table 3.1 as polynomial degree m taking the dominants term of order $O(m^2)$ Apply (4.3) and data given in Table 6, we calculate the percentage ratio $\rho((NMSM-1- NMSM-4),(X))$ [8 given by:

$$\rho((NMSM-1- NMSM-4),(X)) = \left(\frac{EL(NMSM-1- NMSM-4)}{EL(X)} - 1 \right) \times 100 , \tag{54}$$

where X is one of the methods, namely Elirch-Aberth and Petkovic methods. Figure 4.1 graphically illustrates these percentage ratioes. It is evident from figure 4.1 that the newly constructed simultaneous methods (3.11,3.12,3.13,3.14) is more efficient as compared to the Elirch [14] and Petkovic methods [8,13].

Table 6: **The number of basic operations**				
Methods	CO	AS_m	M_m	D_m
NMSM-1	12	$5m^2 O(m)$	$2m^2 O(m)$	$2m^2 O(m)$
NMSM-2	12	$8m^2 O(m)$	$3m^2 O(m)$	$2m^2 O(m)$
NMSM-3	12	$7m^2 O(m)$	$2m^2 O(m)$	$2m^2 O(m)$
NMSM-4	12	$5m^2 O(m)$	$2m^2 O(m)$	$2m^2 O(m)$
PJ10D	10	$22m^2 O(m)$	$18m^2 O(m)$	$2m^2 O(m)$

We also calculate the CPU execution time, as all the calculations are done using maple 18 on (Processor Intel(R) Core(TM) i3-3110m CPU@2.4GHz with 64-bit Operating System. We observe that CPU time of the method MMN8M is less than M. S. Petkovic methods [8], showing the domminace efficiency of our method 3.11,3.12,3.13,3.14 as compared to them.

5(a)   5(b)   5(c)   5(d)
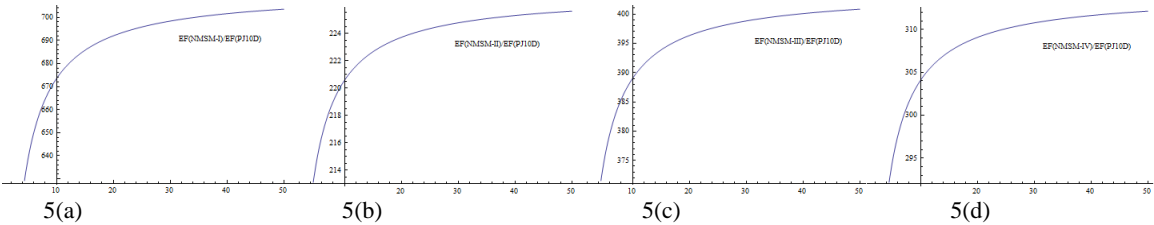
Numerical Results

Here, some numerical examples are considered in order to demonstrate the performance of our family of two-step fourth order single root finding methods (NMS-1,NMS-2,NMS-3,NMS-4) and two-step twelfth order simultaneous methods (NMSM-2,NMSM-2,NMSM-3,NMSM-4) respectively. We compare our family of single root finding methods with optimal fourth order method JM,YM and SM. Family of simultaneous methods of order twelve are compare with J. Džunic, M. S. Petkovic and L. D. Petkovic [8] method of order ten (abbreviated as PJ10D method). All the computations are performed using *Maple* 18 with 2500 (64 digits floating point arithmetic in case of simultaneous methods) significant digit with stopping criteria is as follow.

$$(i)\qquad e_i = \left| f\left(x_i^{(n+1)}\right) \right| \leq \varepsilon, \qquad (ii)\qquad e_i = \left\| x_i^{(n+1)} - r_i \right\|_2 \leq \varepsilon,$$

where e_i represents the absolute error of function values in (i) and norm-2 in (ii) [8]. We take $\varepsilon = 10^{-200}$ for single root finding method and $\varepsilon = 10^{-30}$ for simultaneous determination of all roots of non-linear equation ().

Numerical tests examples from [10, 14, 15, 17, 23] are provided in Tables 7(a, b) and 8-13 .In Table 8-13 the stopping criteria (*i*) is used while in Table 7(a, b) the stopping criteria (i) and (ii) both are used. In all Tables CO represents the convergence order, n represents the number of iterations, ℓ =1 represents all distinct root, $\otimes$ ⊛ 1 represents multiple roots and CPU represents computational time in seconds. We observe that numerical results of the methods (in case of single (NMS-1, NMS-2, NMS-3, NMS-4) as well as simultaneous determination (NMSM-2, NMSM-2, NMSM-3, NMSM-4) of all roots) are better than JM, YM SM and PJ10D respectively on same number of iterations. The Figure 7(a-d) and 8(a, b)-13(a, b), shows the residual fall of different methods for the non-linear function $f_1(x), f_2(x), f_3(x), f_4(x)$ and examples 4.1,4.2,4.3,4.4,4.5,4.6 (in case of simultaneous methods), shows that methods (NMS-1,NMS-2,NMS-3,NMS-4) and (NMSM-2,NMSM-2,NMSM-3,NMSM-4) are more efficient as compared than JM, YM SM and PJ10D respectively.

$(i)......f_1(x)[23] = (\sin(x))^2 - x^2 + 1, \alpha = 1.4044916482.$

$(ii)......f_2(x)[17] = \sqrt{x^2 + 2} - \sin\left(\dfrac{\pi}{x}\right) + \dfrac{1}{x^4 + 1} - \sqrt{3} - \dfrac{1}{17}, \alpha = -2$

$(iii)......f_3(x)[23] = \cos(x) + \sin(2x)\sqrt{1 - x^2} + \sin(x^2) + x^{14} + x^3 + \dfrac{1}{2x}, \alpha = -0.9257722498$

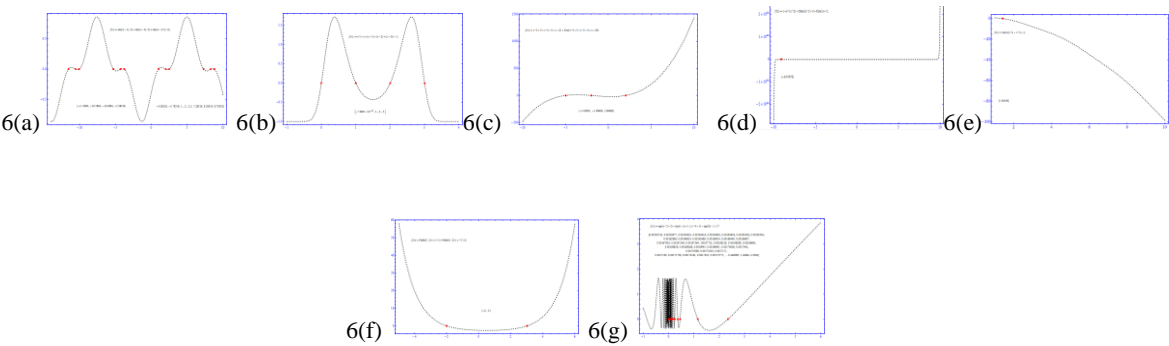$(iv)......f_4(x)[23] = xe^{x^2} - (\sin(x))^2 + 3\cos(x) + 5, \alpha = -1.207647827130919$

6(a)   6(b)   6(c)   6(d)   6(e)

6(f)   6(g)

Table.7(a): Comparsion of optimal Fourth order methods											
Initial guesses	$\|f_i(\Omega_i)\|$	CPU	CO	n	JM	SM	AM	NMS-1	NMS-2	NMS-3	NMS-4
f_1, $x_0 = 1.5$	$\|f_1(\Omega_6)\|$		4	6	4.3e-90	div	1.4e-58	1.3e-697	1.6e-700	2.0e-700	2.9e-700
CPU for f_1					0.125		0.109	.110	0.125	0.125	0.109
f_2, $x_0 = 2.25$	$\|f_2(\Omega_6)\|$		4	6	8.0e-90	div	2.5e-48	1.0e-931	2.1e-886	2.6e-884	1.5e-880
CPU for f_2					0.111		0.125	0.125	0.125	0.125	0.141
f_3, $x_0 = 0.93$	$\|f_3(\Omega_6)\|$		4	6	8.5e-117	div	3.1e-83	8.8e-1011	1.7e-1011	1.7e-1011	1.8e-1011
CPU for f_3					0.250		0.234	0.234	0.250	0.250	0.266
f_4, $x_0 = 1.3$	$\|f_4(\Omega_6)\|$		4	6	4.7e-71	div	2.8e-43	2.5e-541	1.4e-540	1.3e-540	1.1e-540
CPU for f_4					0.140		0.141	0.141	0.141	0.156	0.140

| Table.7(b): Comparsion of optimal Fourth order methods										
Initial guesses	$\left\|x_i^{⚘☰☋} ✍☺\right\|$	CO	n	JM	SM	AM	NMS-1	NMS-2	NMS-3	NMS-4
$f_1,\ x_0$ ⊞1.5	$\left\|x_i^{☾☋} ✍☺\right\|$	4	6	4.4e-90	div	6.0 e-30	3.1e-233	3.4e-234	3.6e-234	4.1e-234
CPU for f_1				0.125		0.109	.110	0.125	0.125	0.109
$f_2,\ x_0$ ⊞✍2.25	$\left\|x_i^{☾☋} ✍☺\right\|$	4	6	8.3e-45	div	2.3e-24	2.0e-310	2.6e-295	1.3e-294	2.3e-293
CPU for f_2				0.111		0.125	0.125	0.125	0.125	0.141
$f_3,\ x_0$ ⊞✍0.93	$\left\|x_i^{☾☋} ✍☺\right\|$	4	6	1.9e-591	div	6.2e-43	2.4e-338	1.4e-338	1.4e-338	1.4e-338
CPU for f_3				0.250		0.234	0.234	0.250	0.250	0.266
$f_4,\ x_0$ ⊞✍1.3	$\left\|x_i^{☾☋} ✍☺\right\|$	4	6	1.6e-36	div	6.8e-23	1.4e-181	2.5e-181	2.5e-181	2.3e-181
CPU for f_4				0.140		0.141	0.141	0.141	0.156	0.140
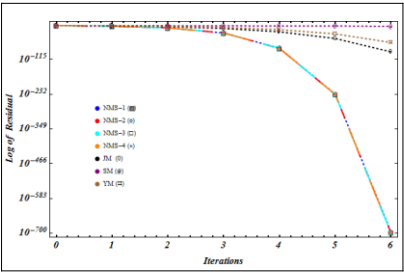
7(a)
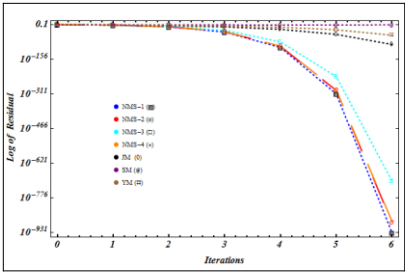
7(b)
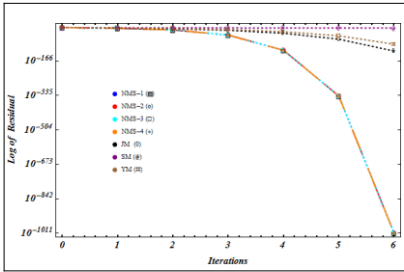
7(c)
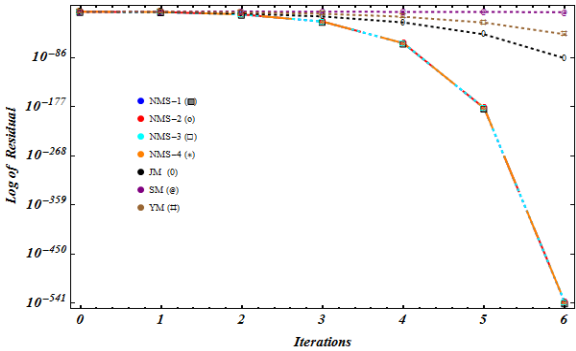
7(d)

Example 4.1[15]: Consider
$$f_5(x) = (x+1)^2 (x+2)^3 (x^2 - 2x + 2)^2 (x^2 + 1)^2 (x-2)^3 (x+2i)^2,$$

with multiple exact roots (⊘ 1):

$$x_1 = -1, \quad x_{2} = -2, \quad x_{3,4} = -2 \pm i, \quad x_{5,6} = \pm i, \quad x_7 = 2, \quad x_8 = -2 \equiv i.$$

The initial approximations have been taken as:

$$x_1 = -1.3 - 2i, \quad x_2 = -2.2 - 3i, \quad x_3 = 1.3 - 1.2i, \quad x_4 = 0.7 - 1.2i, \quad x_5 = -0.2 - 8i,$$

$$x_6 = 0.2 - 1.3i, \quad x_7 = 2.2 - 3i, \quad x_8 = -2.2 - 0.7i.$$

For distinct roots (⊘ 1):
$$f(x) = (x+1)(x+2)(x^2 - 2x + 2)(x^2 + 1)(x-2)(x+2i).$$

Table.8												
Method	CO	CPU	⊘	n	e1	e2	e3	e4	e5	e6	e7	e8
PJ10D	10	0.766		2	1.2e-1	3.0e-2	2.7e-1	7.9e-2	1.8e-1	3.1e-2	1.2e-1	8.2e-3
NMSM(1-4)	12	0.250		2	1.9e-36	1.8e-36	0.0	0.0	1 2e-39	3.8e-37	3.5e-37	0.0
NMSM(1-4)	12	0.328		2	2.0e-58	1.0e-51	0.0	6.4e-81	2.6e-120	1.5e-82	1.6e-51	3.8e-69

8(a)

8(b)
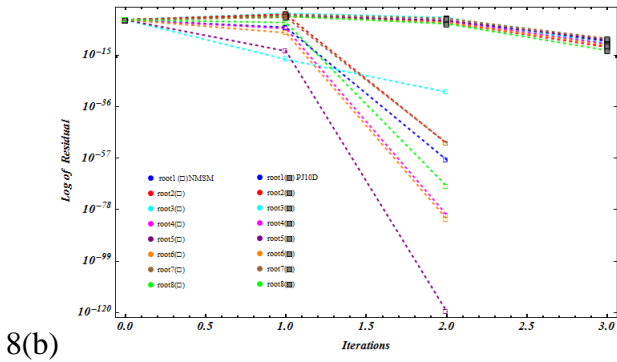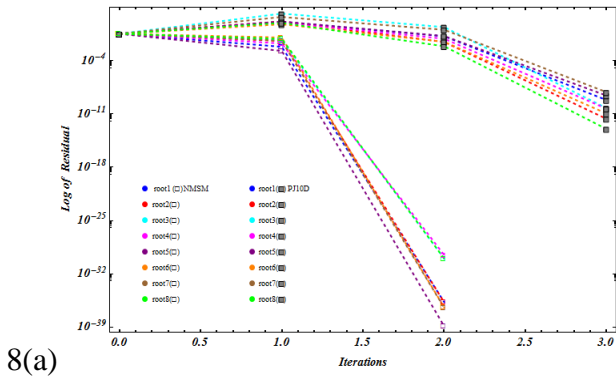
Example4.2[15]: Consider
$$f_6(x) = (x+1)^3 (x+2)^2 (x-1-i)^3 (x-1+i)^3,$$

with multiple exact roots (⊘ 1):

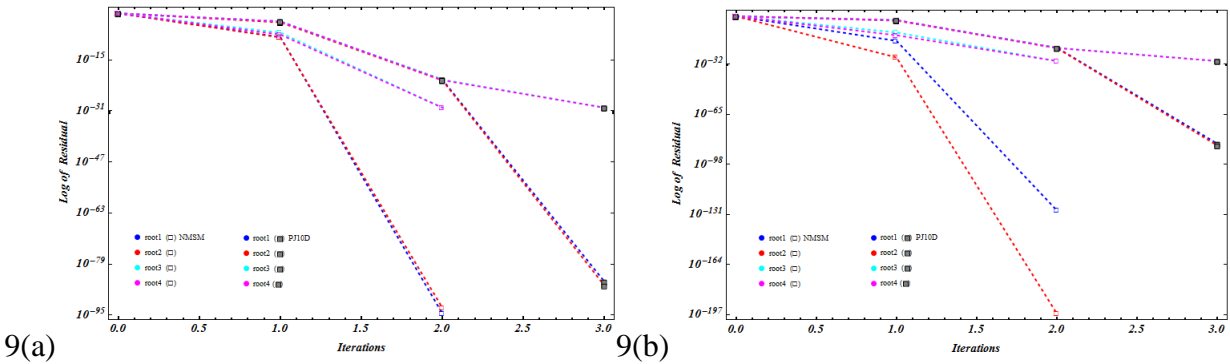$$x_1 = -1, \quad x_{2} = -2, \quad x_3 = 1 - i, \quad x_4 = 1 - i.$$

The initial approximations have been taken as:

$$x_1 = -1.1 - 0.2i, \quad x_2 = -2.1 - 2i, \quad x_3 = 0.8 - 1.2i, \quad x_4 = 0.9 - 1.2i.$$

For distinct roots ($m = 1$):

$$f(x) = (x+1)(x+2)(x-1-i)(x-1+i)$$

Table.9								
Method	CO	CPU	$\otimes$	n	e1	e2	e3	e4
PJ10D	10	0.047	σ	2	5.9e-22	3.4e-22	5.9e-22	4.2e-22
NMSM(1-4)	12	0.032	$\otimes$	2	3.8e-95	1.9e-93	0.0	0.0
NMSM(1-4)	12	0.032	$\otimes$	2	3.0e-129	2.5e-197	0.0	0.0

9(a)   9(b)

Example4.3[14]: Consider

$$f_7(x) = \left(e^{x\,(x-1)\,(x-2)\,(x-3)} - 1\right)^5,$$

with multiple exact roots ($m = 1$):
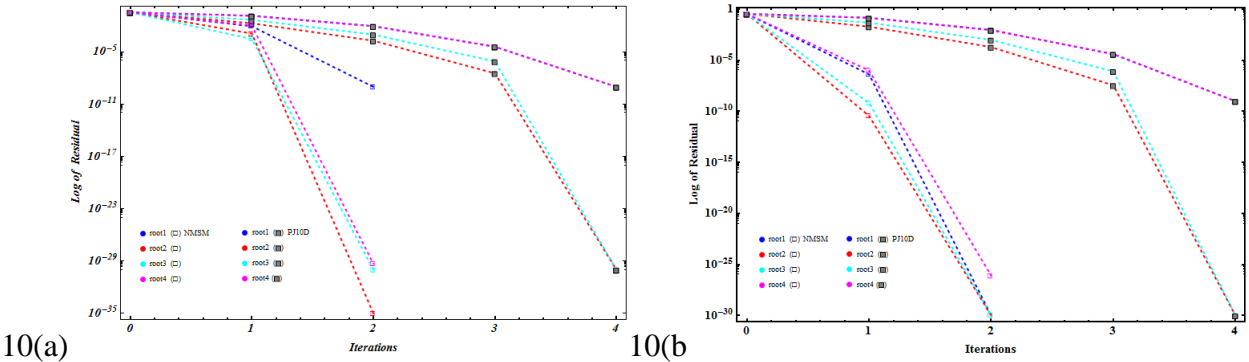
$$x_1 = 0, \quad x_2 = 1, \quad x_3 = 2, \quad x_4 = 3.$$

The initial approximations have been taken as:

$$x_1 = 0.1, \quad x_2 = 0.9, \quad x_3 = 1.8, \quad x_4 = 2.9,$$

For distinct roots ($m = 1$):

$$f(x) = \left(e^{x\,(x-1)\,(x-2)\,(x-3)} - 1\right).$$

Table.10								
Method	CO	CPU	⊘	n	e1	e2	e3	e4
PJ10D	10	0.156		2	9.3e-3	2.7e-4	1.2e-3	9.3e-3
NMSM(1-4)	12	0.078	⊘	2	1.0e-9	0.0	0.0	2.1e-9
NMSM(1-4)	12	0.078	⊘	2	0.0	0.0	0.0	8.1e-27

10(a)   10(b)

Example 4.4[10]:   Consider

$$f_8(x) = \left(x^3 + 5x^2 - 4x - 20 + \cos\left(x^3 + 5x^2 - 4x - 20 \right) - 1 \right)^5,$$

with multiple exact roots (⊘ 1):

$$x_1 = -5, \quad x_2 = -2, \quad x_3 = 2,$$

The initial approximations have been taken as:

$$x_1^{(0)} = -5.1, \quad x_2^{(0)} = -1.8, \quad x_3^{(0)} = 1.9.$$

For distinct roots (⊘1):

$$f^{(0)} = x^3 + 5x^2 - 4x - 20 + \cos\left(x^3 + 5x^2 - 4x - 20 \right) - 1.$$

Table.11							
Method	CO	CPU	⊘	n	e1	e2	e3
PJ10D	10	0.187		2	4.9e-3	6.0e-3	2.5e-1
NMSM(1-4)	12	0.094	⊘	2	7.2e-11	1.7e-10	2.3e-4
NMSM(1-4)	12	0.087	⊘	2	4.9e-51	1.3e-50	5.0 e-19

11(a)   11(b)
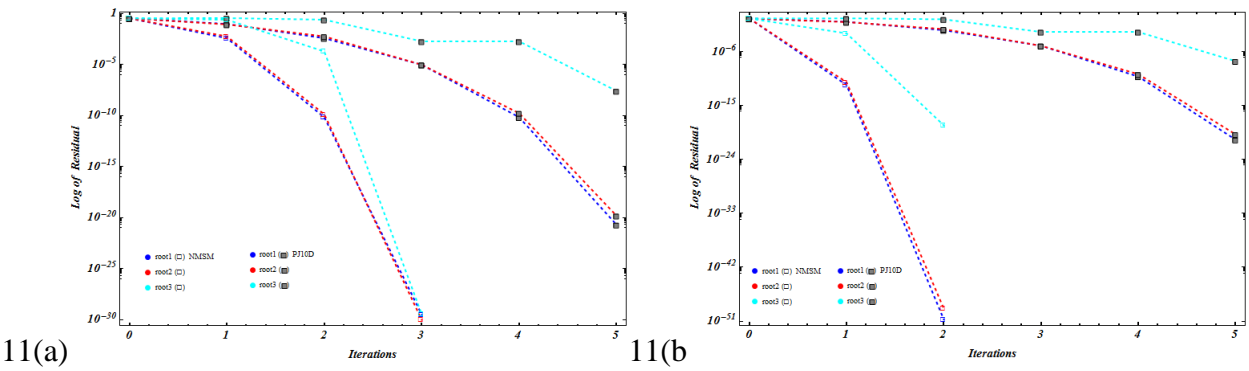
Example 4.5[14]: Consider

$$f_9(x) = \sin^3\left(\frac{x-1}{2}\right)\sin^3\left(\frac{x-2}{2}\right)\sin^3\left(\frac{x-2.5}{2}\right),$$

with multiple exact roots ($\oslash$ 1):
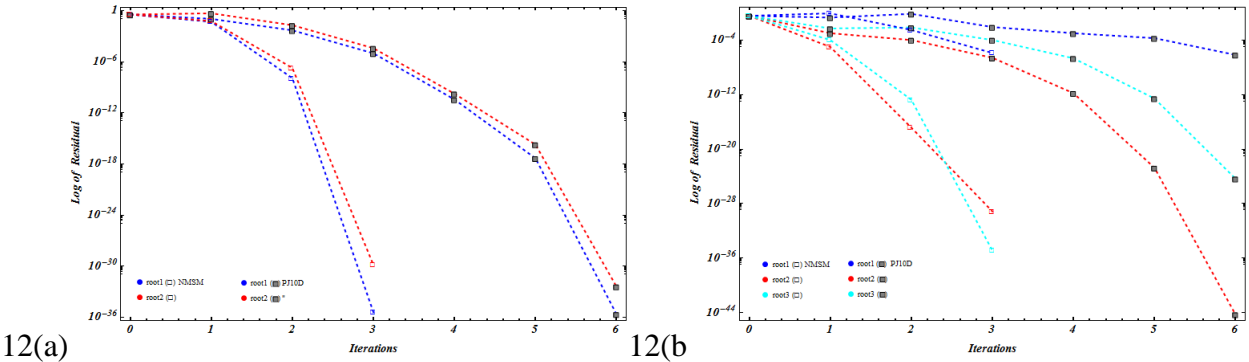
$$x_1 = 0, \quad x_2 = 1, \quad x_3 = 2, \quad x_4 = 3.$$

The initial approximations have been taken as:

$$x_1 = 0.1, \quad x_2 = 0.9, \quad x_3 = 1.8, \quad x_4 = 2.9.$$

For distinct roots ($\oslash = 1$):

$$f(x) = f(x) = \sin\left(\frac{x-1}{2}\right)\sin\left(\frac{x-2}{2}\right)\sin\left(\frac{x-2.5}{2}\right).$$

Table.12							
Method	CO	CPU	$\oslash$	n	e1	e2	e3
PJ10D	10	0.312		2	6.4e-3	1.0e-3	1.6e-4
NMSM(1-4)	12	0.140		2	3.0e-3	1.0e-17	1.3 e-13
NMSM(1-4)	12	0.109		2	1.6e-3	1.6e-34	6.9e-53

12(a)   12(b)

Example 4.6[14]: Consider

$$f_{10}(x) = \sinh^5\left(\frac{x+2}{2}\right)\sinh^5\left(\frac{x-3}{2}\right)$$

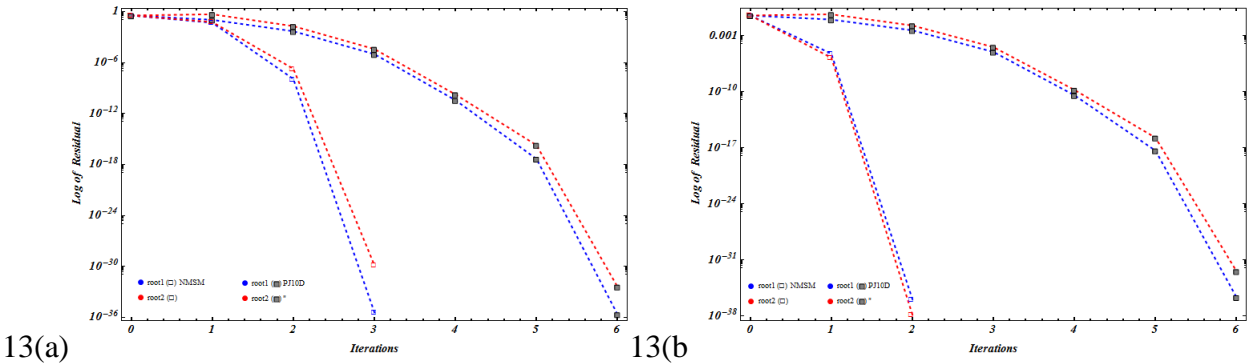with multiple exact roots (☺✴ 1):

⤴ ▣0, ⤴₂ ▣1, ⤴₃ ▣2, ⤴₄ ▣3.

The initial approximations have been taken as:

x_1 ▣0.1, x_2 ▣0.9, x_3 ▣1.8, x_4 ▣2.9.

For distinct roots (☺▣1):

$$f\ \mho\ \Omega\ \boxminus\ \sinh\left(\frac{x\ \boxminus 2}{2}\right)\sinh\left(\frac{x\ \measuredangle 3}{2}\right)$$

Table.13						
Method	CO	CPU	☺	n	e1	e2
PJ10D	10	0.047	☺▣	2	6.4e-3	1.0e-3
NMSM(1-4)	12	0.016	☺▣	2	1.2e-8	1.0e-7
NMSM(1-4)	12	0.030	☺✴	2	1.2e-36	1.4e-38

13(a)   13(b)

Conclusion

We have developed here two-step two iterative methods one for simple determination of single roots (NMS-1,NMS-2,NMS-3,NMS-4)   and other simultaneous determination of all roots (NMSM-2,NMSM-2,NMSM-3,NMSM-4)   of convergence order four and twelve. From Tables 2-5, 7(a, b), 8-13 and Figure 1(a-g)-4(a-g), 5(a-d),7(a-d), 8(a, b)-13(a, b), we observe that our methods (NMS-1,NMS-2,NMS-3,NMS-4), (NMSM-2,NMSM-2,NMSM-3,NMSM-4)   are superior in term of efficiency, CPU time and residual error as compared to the methods PJ6M, PJ10D [8,13] and ELM methods.

References

[1]　M. Cosnard and P. Fraigniaud, Finding the roots of a polynomial on an MIMD multicomputer, Parallel Computing, vol. 15, no. 1--3, (1990), 75--85.

[2]　S. Kanno, N, Kjurkchiev, T. Yamamoto, On some methods for the simultaneous determination of polynomial zeros, Japan J. Apple. Math. 13(1995), 267-288.

[3]　O. Abert, Iteration methods for finding all zeros of a polynomial simultaneously, Math. Comput. 27(1973) 339-344.

[4]　P.D. Proinov, S.I. Cholakov, Semilocal convergence of Chebyshev-like root-finding method for simultaneous approximation of polynomial zeros, Appl. Math. Comput. 236(2014), 669-682.

[5]　Bl. Sendov, A. Andereev and N. Kjurkchiev, Numerical Solutions of polynomial Equations, Elsevier science, New York, 1994.

[6]　X. Wang, L. Liu, Modified Ostrowski's method with eight --order convergence and high efficiency index, Applied Mathematics Letters, 23(2010), 549-554.

[7]　T.-F. Li, D.-S. Li, Z.-D.Xu, and Y.-I. Fang, New iterative methods for non-linear equations, Applied Mathematics and Computations,.(2008), 755-759.

[8]　M.S.Petkovi´c, L.D.Petkovic, J.Džunic, On an efficient simultaneous method for finding polynomial zeros, Appl. Math. Lett. 28(2014), 60-65.

[9]　P.D. Proinov, M.T. Vasileva, On the convergance of family of Weierstrass-type root-finding methods, C. R. Acad. Bulg. Sci, 68(2015), 697-704.

[10]　N. A. Mir, R. Muneer, I. Jabeen, Some families of two-step simultaneous methods for determining zeros of non-linear equations,ISRN Applied Mathematics, (2011), 1-11.

[11]　P.D. Proinov, S.I. Cholakov, Semilocal convergence of Chebyshev-like root-finding method for simultaneous approximation of polynomial zeros, Appl. Math. Comput. 236(2014), 669--682.

[12]　M.S.Petkovic, L.D.Petkovic, J.Džunic, On an efficient simultaneous method for finding polynomial zeros, Applied Mathematics Letters, 28(2014), 60-65.

[13]　O. Aberth, Iteration methods for finding all zeros of a polynomial simultaneously, Math. Comp. 27(1973) 339-344.

[14]　G. H. Nedzhibov, Iterative methods for simultaneous computing arbitrary number of multiple zeros of nonlinear equations, International Journal of Computer Mathematics, (2013), 994-1007.

[15]　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, July 2014.

[16]　I.K. Argyros, D. Chen, Q. Qian. The Jarratt method in Banach space setting. J. Comput. Appl. Math. 51, pp. 103-106, 1994.

[17]　H. I. Siyyam, I.A. Al-Subaihi. A new class of optimal fourth-order iterative methods for solving nonlinear equations, European Journal of Scientific Research. 109, 2, pp. 224-231, 2013.

[18]　Osama Y. Ababneh, New fourth order iterative methods second derivative free, J. Appl. Math. Phy. 2016, 4, 519-523

[19]　F. I. Chicharro, A. Cordero , N. Garrido, J. R. Torregrosa, Generating root-finder iterative methods of second order: convergence and stability, Axioms 2019, 8, 55.

[20]   A.Cordero, J. García-Maimó, J.R. Torregrosa, M.P. Vassileva, P. Vindel, Chaos in King's iterative family, Appl. Math. Lett. 26 (2013), 842--848.

[21]   F. Chicharro, A. Cordero, J.R. Torregrosa, Drawing dynamical and parameters planes of iterative families and methods, Sci. World J. 2013 (2013).

[22]   A. W. M. Nourein, An improvement on two iteration methods for simultaneous determination of the zeros of a polynomial, Inrern. J. Cornput. Math. (1977), 241--252.

[23]   F. Soleymani, Optimal fourth-order iterative methods free from derivatives, Miskolc Mathematical Notes, vol. 12 (2011), no. 2, 255–264.

[24]   P.D. Proinov, A general semilocal convergence theorem for simultaneous methods for polynomial zeros and its applications to Ehrlich's and Dochev-Byrnev's methods, Appl. Math. Comput. 284 (2016), 102--114.

[25]   Jay, L.O. A note on Q-order of convergence. BIT Numer. Math. 2001, 41, 422--429.

[26]   P.D. Proinov, S.I. Ivanov, On the convergence of Halley's method for simultaneous computation of polynomial zeros, J. Numer. Math. 23 (2015), 379--394.

[27]   P.D. Proinov, S.I. Ivanov, Convergence analysis of Sakurai-Torii-Sugiura iterative method for simultaneous approximation of polynomial zeros, J. Comput. Appl. Math. 357 (2019), 56--70.

[28]   S.I. Cholakov, Local and semilocal convergence of Wang-Zheng's method for simultaneous finding polynomial zeros, Symmetry 2019 (2019), Art. 736, 15pp.

[29]   M. Dehghan, M. Hajarian, On derivative free cubic convergence iterative methods for solving nonlinear equations, Comput. Math. Math. Phy. 51(2011), 513-519.

[30]   M. Dehghan, M. Hajarian, On some cubic convergence iterative formulae without derivatives for solving nonlinear equations. Int. J. Numer. Meth. Biomed. Eng. Vol. 27, 27(2011), 722-731.