



## An optimal fourth order method for solving nonlinear equations



M. A. Hafiz<sup>a</sup>, M. Q. Khirallah<sup>a,b,\*</sup>

<sup>a</sup>Department of Mathematics, Faculty of Science and Arts, Najran University, Najran 1988, Saudi Arabia.

<sup>b</sup>Department of Mathematics and Computer Science, Faculty of Science, Ibb University, Yemen.

### Abstract

In this paper, we use both weight functions and composition techniques together for solving non-linear equations. We designed a new fourth order iterative method to increase the order of convergence without increasing the functional evaluations in a drastic way. This method uses one evaluation of the function and two evaluations of the first derivative. The new method attains the optimality with efficiency index 1.587. The convergence analysis of our new methods is discussed. Furthermore, the correlations between the attracting domains and the corresponding required number of iterations have also been illustrated and discussed. The comparison with several numerical methods and the use of complex dynamics and basins of attraction show that the new method gives good results.

**Keywords:** Nonlinear equations, basins of attraction, iterative methods, optimal methods, complex dynamics.

**2020 MSC:** 41A25, 65H05, 65K05.

©2021 All rights reserved.

### 1. Introduction

Numerical analysis has interesting applications in several branches of pure and applied science that can be studied in the general framework of the non-linear equations [2, 12, 17, 26, 32]. Searching out a solution for non-linear equations is highly significant. Due to their importance, several numerical methods have been suggested and analyzed under different conditions [1, 6, 9, 12, 21, 23, 25, 30]. These numerical methods have been constructed by using different techniques for solving the non-linear equations such as Taylor series, quadrature formula, the variational iteration method, and the decomposition method [3, 7, 11, 15, 18, 20, 21, 33].

In this research, we describe a new iterative method from the first derivative to find a simple root  $r$  of a non-linear equation by using both weight functions and composition techniques together. We designed a new iterative method to increase the order of convergence without increasing the functional evaluations in a drastic way. We can achieve the optimal order without difficulties, as we will show, the obtained two-step method is of fourth-order of convergence and requires three evaluations of the function  $f(x)$ . Commonly in literature, the efficiency of an iterative method is measured by the efficiency

\*Corresponding author

Email address: [mqm73@yahoo.com](mailto:mqm73@yahoo.com) (M. Q. Khirallah)

doi: [10.22436/jmcs.023.02.02](https://doi.org/10.22436/jmcs.023.02.02)

Received: 2020-06-29 Revised: 2020-07-16 Accepted: 2020-08-03

index defined as  $I \approx p^{1/d}$ , where  $p$  is the order of convergence and  $d$  is the total number of functional evaluations per iteration. Therefore, this method has efficiency index  $4^{1/3} \approx 1.5874$  that is the new method which reaches the optimal order of convergence four. In this research, we also study and analyze the stability of the designed method by using tools of complex dynamics. Through the use of these tools, it is possible to compare different algorithms in terms of their basins of attraction and the dynamical behavior of the iterative method on the complex plane [19, 28, 29, 31]. Furthermore, comparisons with several numerical methods, and the use of complex dynamics and basins of attraction are done in this paper [4, 5, 8, 10, 13, 14, 24, 27].

## 2. The fourth-order method

To constrict the method of optimal fourth order of convergence, we designed the following iterative methods through the composition of two steps, in which the first step is similar to Jarrett's method [14], and by using a weight function  $W$  depending on which is included in the second step. Provided that the number of functional evaluations per full iteration is three; thus, we consider the iterative expression as follows:

$$y_n = x_n - \frac{2f(x_n)}{3f'(x_n)}, \quad x_{n+1} = y_n - \frac{f(x_n)}{f'(x_n) + f'(y_n)} W(\eta), \quad (2.1)$$

where  $W(\eta) = \frac{1}{6}[a + b\eta + c \ln(2 - \eta)]$ ,  $2 - \eta \geq 1$  and  $\eta = \frac{f'(x)}{f'(y)}$  without the index  $n$ . The following theorem indicates the conditions on the weight functions in (2.1), such that the order of convergence will arrive at the optimal level four.

**Theorem 2.1.** *Let  $e_n = x_n - r$  be a real sufficiently differentiable function in an open interval  $I$  and let  $r \in I$  be a simple root of  $f(x) = 0$ . If  $x_0$  is close enough to  $\alpha$  and  $W(\eta)$  that satisfies  $a = 1$ ,  $b = 3$ , and  $c = -4$ , then iterative family (2.1) converges to  $r$  with order of convergence four, and its error equation is*

$$e_{n+1} = \frac{1}{234}(299c_2^3 - 234c_2c_3 + 27c_4)e_n^4 + O(e_n^5),$$

where  $c_n = \frac{f^{(k)}(r)}{n! f'(r)}$ ,  $n \geq 2$ .

*Proof.* Let  $e_n = x_n - r$  be the error at  $n^{\text{th}}$  computing step. By using Taylor's expansion of  $x_n$  for the root  $r$ , we get

$$f(x_n) = f'(r)[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + O(e_n^5)], \quad (2.2)$$

and

$$f'(x_n) = f'(r)[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + O(e_n^5)]. \quad (2.3)$$

Now from (2.2) and (2.3), we have

$$\frac{f(x_n)}{f'(x_n)} = e_n - c_2e_n^2 + 2(c_2^2 - c_3)e_n^3 + (7c_2c_3 - 4c_2^3 + 3c_4)e_n^4 + O(e_n^5). \quad (2.4)$$

From (2.1) and (2.4), we get

$$y_n = \frac{r}{3} + \frac{2}{3}c_2e_n^2 + \frac{4}{3}(c_3 - c_2^2)e_n^3 + \frac{2}{3}(4c_2^3 - 7c_2c_3 + 3c_4)e_n^4 + O(e_n^5). \quad (2.5)$$

From (2.5), we get

$$\begin{aligned} f'(y_n) &= f'(r)[1 + \frac{2}{3}c_2e_n + \frac{1}{3}(c_3 + 4c_2^2)e_n^2 + 4(c_2c_3 - \frac{2}{3}c_2^3 + \frac{1}{27}c_4)e_n^3 \\ &\quad + (\frac{8}{3}(2c_2^4 + c_3^2 - 4c_2^2c_3) + \frac{44}{9}c_2c_4 + \frac{5}{81}c_5)e_n^4 + O(e_n^5)]. \end{aligned} \quad (2.6)$$

The expansion of the weight function variable  $\eta$  is

$$\eta = \frac{f'(x_n)}{f'(y_n)} = 1 - \frac{4}{3}c_2e_n + \frac{4}{9}(-5c_2^2 + 6c_3)c_2e_n^2 + \frac{8}{27}(8c_2^3 - 21c_2c_3 + 13c_4)e_n^3 - \frac{4}{81}(8c_2^4 - 135c_2^2c_3 + 27c_3^2 + 155c_2c_4 - 100c_5)e_n^4 + O(e_n^5),$$

and therefore, weight function  $W(\eta)$  around zero results in:

$$\begin{aligned} W(\eta) &= \frac{1}{6}(a+b) + \frac{2}{9}(b-c)c_2e_n - \frac{2}{27}(c_2^2(5b-3c) - 6c_3(b-c))e_n^2 \\ &+ \frac{4}{243}[2c_2^3(12b-c) + 9c_2c_3(3c-7b) + 39c_4(b-c)]e_n^3 \\ &+ \frac{2}{243}[2c_2^4(4b+21c) - 3c_2^2c_3(45b+19c) + 72c_3^2b \\ &+ c_2c_4(155b-51c) + 100c_5(c-b)]e_n^4 + O(e_n^5). \end{aligned} \quad (2.7)$$

And so, from (2.2), (2.3), and (2.6) we have

$$\frac{f(x_n)}{f'(x_n) + f'(y_n)} = \frac{1}{2}e_n - \frac{1}{6}c_2e_n^2 - \frac{1}{9}(c_2^2 + 3c_3)e_n^3 + \frac{1}{54}(50c_2^3 - 15c_2c_3 - 29c_4)e_n^4 + O(e_n^5). \quad (2.8)$$

Finally, when using (2.7) and (2.8), the error equation of any method of (2.1) is:

$$\begin{aligned} x_{n+1} - r &= y_n - \frac{f(x_n)}{f'(x_n) + f'(y_n)}W(\eta) - r, \\ e_{n+1} &= \frac{1}{12}(4-a-b)e_n + \frac{1}{36}c_2(24+a-3b+4c)e_n^2 \\ &+ \frac{1}{54}[3c_3(24+a-3b+4c) + c_2^2(a+13b-8c-72)]e_n^3 \\ &+ \frac{1}{972}[(c_2^3(2592-150a-378b+28c) + 9c_2c_3(5a+77b-40c-504) \\ &+ c_4(648+29a-75b+104c)]e_n^4 + O(e_n^5). \end{aligned} \quad (2.9)$$

The conditions on the weight function  $W$  are:

$$4-a-b=0, \quad 24+a-3b+4c=0, \quad a+13b-8c-72=0.$$

The solution of this system is:  $a=1$ ,  $b=3$ , and  $c=-4$ . Upon the input of these values on (2.9), the error equation of any method of (2.1) is:

$$e_{n+1} = \frac{1}{234}(299c_2^3 - 234c_2c_3 + 27c_4)e_n^4 + O(e_n^5),$$

where  $e_{n+1} = x_{n+1} - r$ . Therefore, the typical algorithms for our proposed method can be written in the following explicit form:

$$\begin{aligned} y_n &= x_n - \frac{2f(x_n)}{3f'(x_n)}, \\ x_{n+1} &= y_n - \frac{f(x_n)}{6(f'(x_n) + f'(y_n))} \left[ 1 + 3\frac{f'(x_n)}{f'(y_n)} - 4 \ln \left( 2 - \frac{f'(x_n)}{f'(y_n)} \right) \right]. \end{aligned} \quad (\text{MHK})$$

□

Let us remark that in terms of computational cost, the developed method (MHK) requires only three functional evaluations per step. So, they have efficiency indices, that is, the new form of method (MHK) which reaches the optimal order of convergence four, conjectured by Kung and Traub [16]. Also, it should be remarked that many literatures have used similar techniques; however, they introduced different methods that are less efficient than the method suggested here [4, 5, 8, 13, 24, 27].

### 3. Graphical comparison by means of attraction basins

It is known that a good selection of initial guesses plays an important role in iterative methods; therefore, all methods converge if the initial estimation is chosen suitably. As a result, the use of classical comparison between different numerical methods is unsuitable. Because of that, to show that a new method is better than an established one, the proposer usually applies his/her new algorithm to find a root of a function and compares its convergence rate with that of an established method. Unfortunately, this procedure does not prove anything as the proposer might have randomly picked a starting point that belongs to the convergent regions of the new method. This can raise a contradictory result such as shown in, e.g., [22]. However, the study of convergence and stability of the iterative method based on basins of attractions of the function gives important information and comprehension of the method behavior at a glance. Therefore, it is an appropriate and important tool to do this. The basic strategy to generate a picture illustrating the global behavior of methods for a polynomial  $f(z)$  is as follows. We choose a rectangular region  $D$  of the complex plane only for the domain of convergence which contains all roots of concerned nonlinear equation  $f(z)$ . We finally subdivide the region into rectangles, say  $360 \times 360$  for each of them, corresponding to a complex number. We apply the iterative methods from each point of complex numbers as an initial guess  $z_0$  and color its point according to which root the sequence converges. If the sequence does not converge to a root, we mark it as a black point. Note that the black color denotes the lack of convergence to any of the roots. This happens particularly when the method converges to a fixed point that is not a root or if it ends in a periodic cycle or at infinity. Different colors are used for different roots. In the basins of attraction, the number of iterations needed to achieve the root is shown by the brightness of colors. Brighter color means less iteration steps, and dark color is assigned for showing more iteration. It should be noted that the presence of black dots in the image does not necessarily mean that the method is not convergent at these points or that it is unable to find a solution at these points; however, it means that the method failed to find the solution under the conditions established for convergence, such as the number of steps and the fixed stopping criterion of error allowed in the solution (tolerance).

In order to graphically compare by means of attraction basins, we investigate the dynamics of the new methods with the most recent, most famous, and most effective methods of optimal fourth order.

Because numerical methods for solving nonlinear equations have developed greatly, they are now able to solve any simple equations. Therefore, very complicated equations must be chosen to test the efficiency of the new methods. That is why we also consider the corresponding figure for finding the roots of the function.

For comparisons, in addition to the Newton’s method, we will use the more efficient fourth-order methods as mentioned in Table 1, where  $y_n = x_n - \frac{2}{3} \frac{f(x_n)}{f'(x_n)}$ , and the methods in Table 2 of which the first step is Newton’s method, where  $P_1(x_n, y_n) = 2 \left( \frac{f(y_n) - f(x_n)}{y_n - x_n} \right) - f'(x_n)$ .

Table 1: Some formulas which are used to solve the problems.

Method	Second step of the method
Jarratt (JR) [14]	$x_{n+1} = x_n - \frac{5f(x_n)}{8f'(x_n)} - \frac{3f(x_n)f'(x_n)}{8f'^2(y_n)}$
Jaiswal (JPJ) [13]	$x_{n+1} = x_n - \left( \frac{2f'(y_n)}{3f'(y_n) - f'(x_n)} \right) \left[ \frac{3}{2} - \frac{3f'(y_n)}{4f'(x_n)} + \left( \frac{f'(y_n)}{2f'(x_n)} \right)^2 \right] \frac{f(x_n)}{f'(x_n)}$
Changbum (CMB) [5]	$x_{n+1} = x_n - \frac{16f(x_n)f'(x_n)}{-5f'^2(x_n) + 30f'(x_n)f'(y_n) - 9f'^2(y_n)}$
Sharma (SB) [24, 27]	$x_{n+1} = x_n - \left( \frac{-1}{2} + \frac{9f'(x_n)}{8f'(y_n)} + \frac{3f'(y_n)}{8f'(x_n)} \right) \frac{f(x_n)}{f'(x_n)}$

Table 2: Second step of the method

Method	Second step of the method
Behzad (BG)[8]	$x_{n+1} = y_n - \frac{f(x_n)+f(y_n)}{f(x_n)-f(y_n)} \frac{f(y_n)}{f'(x_n)}$
Chicharro (FC) [4]	$x_{n+1} = x_n - \frac{f^2(x_n)+f(x_n)f(y_n)+2f^2(y_n)}{f(x_n)f'(x_n)}$
HBM [10]	$x_{n+1} = y_n - \frac{4f(y_n)}{f'(x_n)+P_1(x_n,y_n)} - \frac{f(y_n)}{f'(x_n)}$

#### 4. Numerical examples in complex domain

In this section, we present some examples, (6 examples), to illustrate the efficiency of the new proposed iterative method to find complex roots of complex function  $f:C \rightarrow C$ . We compare between the new fourth order iterative method, namely (MHK) and eight different methods as Jaiswal method (JPJ) [13], Sharma Methods (SB) [24, 27], Changbum method (CMB) [5], Jarratt method (JR)[14], Behzad method (BG)[8], Chicharro method (FC)[4], classical Newton’s method (NM), and Bahgat method (HBM) [10]. All these examples are solved in complex domain, and the method proposed used in this paper is by using Mathematica program.

The proposed methods are studied in the complex plane by using basins of attraction. The comparison illustrates the superiority of our proposed method (MHK) in several cases.

Table 3: Some examples and their roots in complex domain.

Functions	Roots
$f_1(z) = z^2 - 1$	$\pm 1$ .
$f_2(z) = z^3 - 1$	$-0.5 \pm 0.866025 I, 1$ .
$f_3(z) = z^4 - 1$	$\pm I, \pm 1$ .
$f_4(z) = z^{11} - 1$	$-0.959493 \pm 0.281733 I, -0.654861 \pm 0.75575 I,$ $-0.142315 \pm 0.989821 I, 0.415415 \pm 0.909632 I, 0.841254 \pm 0.540641 I, 1,$
$f_5(z) = z^2 - z + 1/z$	$0.877439 \pm 0.744862 I, -0.754878$
$f_6(z) = z^6 + 10z^3 - 8$	$0.906359, -0.45318 \pm 0.78493 I, -2.20663, 1.10332 \pm 1.911 I$

Table 4: Examples 1 and 2.

$f_1(z) = z^2 - 1$ Method	Black Points N = 8, step= 0.01	Time	$f_2(z) = z^3 - 1$ Method	Black Points N = 20, step= 0.01	Time
(MHK)	100	8.83	(MHK)	166	7.64
JPJ [13]	4080	10.77	JPJ [13]	8445	10.08
SB [24, 27]	678	9.73	SB [24, 27]	7409	13.84
CMB [5]	2375	10.37	CMB [5]	13517	10.67
JR [14]	3621	15.41	JR [14]	5393	17.43
BG [8]	1126	12.37	BG [8]	9219	12.36
FC [4]	4080	8.86	FC [4]	19188	12.06
NM	797	6.94	NM	12197	6.66
HBM[10]	2835	7.14	HBM[10]	10566	19.2

From Figures 1-6, we illustrate the efficiency of the new proposed iterative method to find complex roots of complex functions, and we compare between the new fourth order iterative method (MHK) and six different methods (JPJ), (SB), (CMB), (JR), (BG), and (HBM). All these examples are solved in complex domain, and the method proposed and used in this paper is by using Mathematica program (11).

From examples 1-6 in Table 4-6 and the Figures 1-6, we see that (MHK) method is better than the other method, and it gives good results.

Table 5: Examples 3 and 4.

$f_3(z) = z^4 - 1$ Method	Black Points N = 8, step= 0.01	Time	$f_4(z) = z^{11} - 1$ Method	Black Points N = 20, step= 0.01	Time
(MHK)	1729	11.39	(MHK)	14082	9.41
JPJ [13]	19446	8.08	JPJ [13]	27587	21.18
SB [24, 27]	23174	11.39	SB [24, 27]	41012	12.22
CMB [5]	27706	16.5	CMB [5]	34524	10.78
JR [14]	15237	10.55	JR [14]	41561	24.31
BG [8]	26904	12.71	BG [8]	45622	14.29
FC [4]	None	-	FC [4]	None	-
NM	30999	9.47	NM	74982	8.95
HBM [10]	10810	31.82	HBM [10]	18978	22.43

Table 6: Examples 5 and 6.

$f_5(z) = z^2 - z + 1/z$ Method	Black Points N = 8, step= 0.01	Time	$f_6(z) = z^6 + 10z^3 - 8$ Method	Black Points N = 20, step= 0.01	Time
(MHK)	4	13.46	(MHK)	692	33.96
JPJ [13]	197	7.6	JPJ [13]	7752	36.97
SB [24, 27]	177	11.92	SB [24, 27]	16998	45.88
CMB [5]	1902	10.33	CMB [5]	15343	46.27
JR [14]	424	23.87	JR [14]	16350	67.22
BG [8]	134	13.45	BG [8]	24417	36.83
FC [4]	846	15.27	FC [4]	--	
NM	494	9.47	NM	24823	37.21
HBM [10]	7703	24.63	HBM [10]	5061	47.7

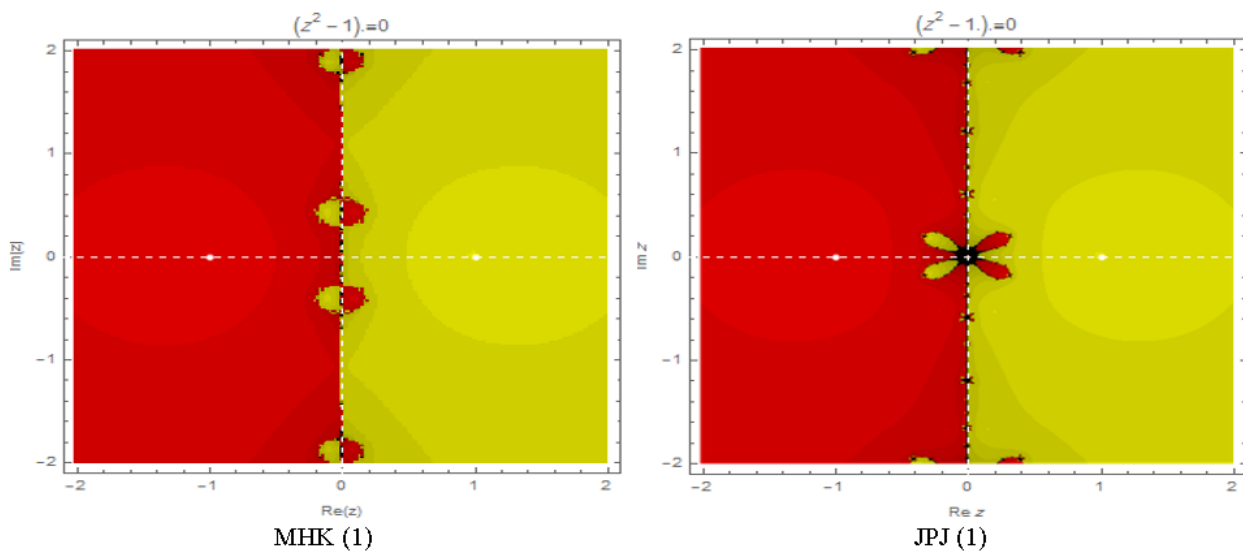


Figure 1: Example 1,  $f_1 = z^2 - 1$ .



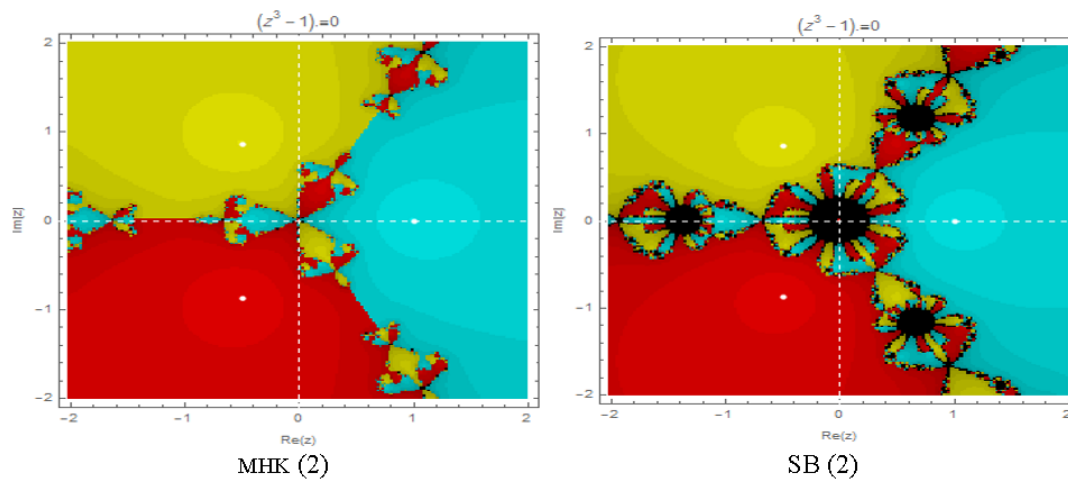


Figure 2: Example 2,  $f_2 = z^3 - 1$ .

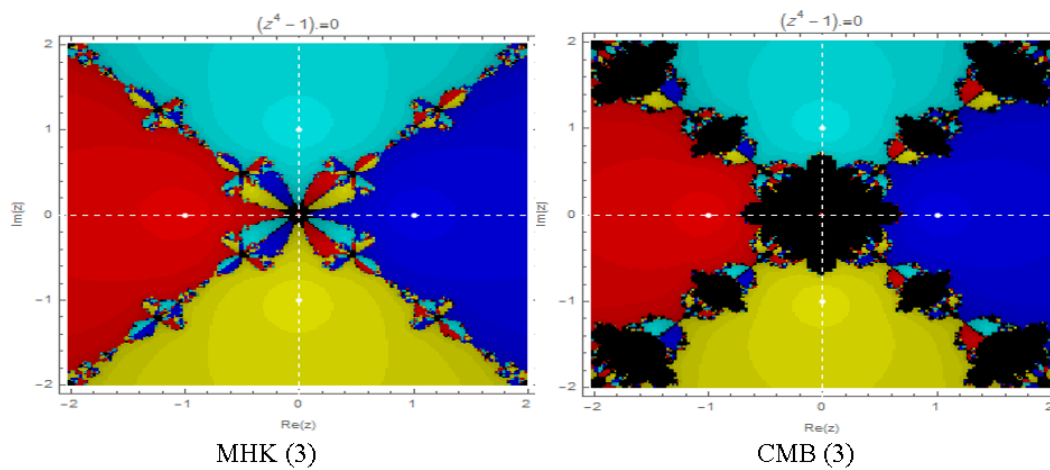


Figure 3: Example 3,  $f_2 = z^4 - 1$ .

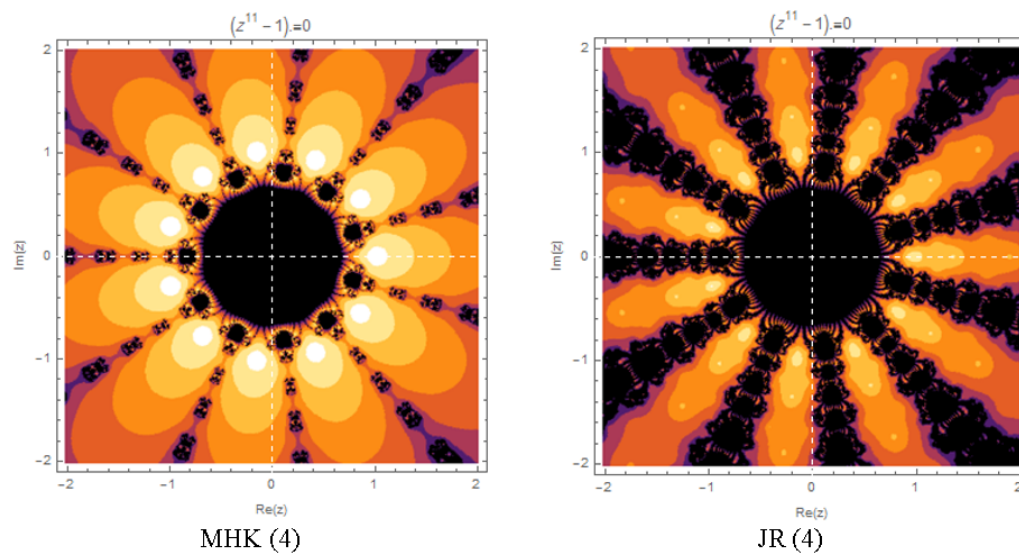


Figure 4: Example 4,  $f_2 = z^{11} - 1$ .

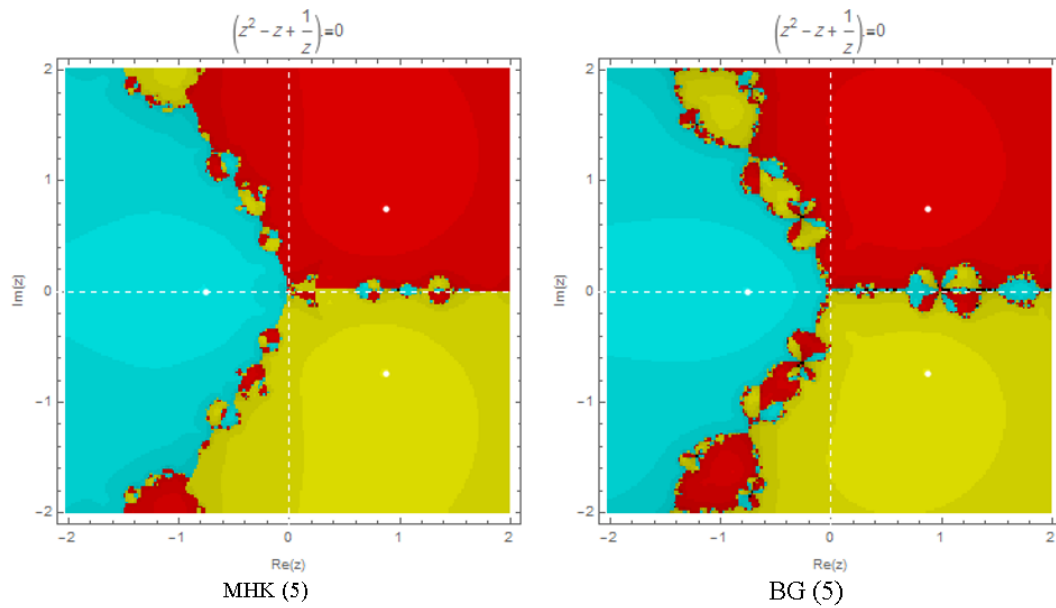


Figure 5: Example 5,  $f_5 = z^2 - z + 1/z$ .

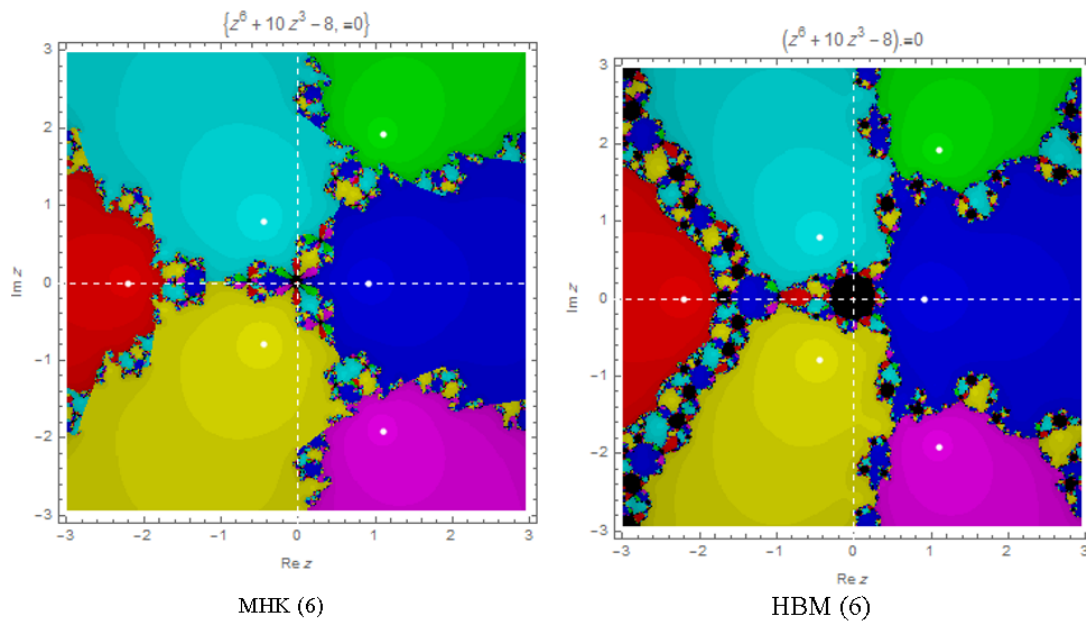


Figure 6: Example 6,  $f_6 = z^6 + 10z^3 - 8$ .

### 5. Numerical examples in real domain

In numerical analysis, many methods produce sequences of real numbers such as the iterative methods for solving nonlinear equations. Sometimes, the convergence of these sequences is slow and their utility in solving practical problems is quite limited. Convergence acceleration methods try to transform a slowly converging sequence into a fast convergent one. Due to this, this paper has aimed to give a rapidly convergent two-point class for approximating simple roots. The highest possible convergence order was attained by using the smallest possible number of evaluations per full cycle. The local order of our class of iterations was established theoretically, and it has been seen that our class supports the optimality conjecture of Kung-Traub [15]. In the sequel, numerical examples have been used in order to show the



efficiency and accuracy of the novel methods from our suggested second derivative-free class.

In this section, we have developed a new optimal fourth-order iterative method for solving non-linear equations. Using both weight functions and composition techniques together, we designed this method to increase the order of convergence without increasing the functional evaluations in a drastic way.

We considered some numerical examples to demonstrate the performance of the new fourth order iterative method, namely (MHK), and we compared this new method with Jaiswal method (JPJ) [13], Sharma Methods (SB) [24, 27], Changbum method (CMB) [5], Jarratt method(JR) [14], Behzad method (BG)[8], Chicharro method (FC)[4], classical Newton’s method (NM), and Bahgat method (HBM) [10]. All these examples were solved in real domain.

In Table 7, our examples are tested with precision  $\epsilon = 10^{-200}$ . The following stopping criterion is used for computer programs:  $|f(x_{n+1})| < \epsilon$ .

And the computational order of convergence (COC) can be approximated using the following formula:

$$COC \approx \frac{\ln |(x_{n+1} - x_n)/(x_n - x_{n-1})|}{\ln |(x_n - x_{n-1})/(x_{n-1} - x_{n-2})|}$$

Table 7 shows the difference of the root  $r$  and the approximation  $x_n$  to  $r$ , where  $r$  is the exact root computed with 2000 significant digits, but only 25 digits are displayed for  $x_n$ . In Table 1, we listed the number of iterations for various methods. The absolute values of the function  $f(x_n)$  and the computational order of convergence (COC) are also shown in Tables 3 and 4. All the computations are performed using Maple 15. The following examples are used for numerical testing:

$$\begin{array}{llll} f_1(x) = x^3 + 4x^2 - 10, & x_0 = 1, & f_2(x) = \sin^2 x - x^2 + 1, & x_0 = 1.3, \\ f_3(x) = x^2 - e^x - 3x + 2, & x_0 = 2, & f_4(x) = \cos x - x, & x_0 = 1.7, \\ f_5(x) = (x - 1)^3 - 1, & x_0 = 2.5, & f_6(x) = x^3 - 10, & x_0 = 2, \\ f_7(x) = e^{x^2+7x-30} - 1, & x_0 = 3.1. & & \end{array}$$

Table 7: Comparisons of the number of iterations with the various methods required, such that  $|f(x_{n+1})| < 10^{-200}$ .

Method	f <sub>1</sub>	f <sub>2</sub>	f <sub>3</sub>	f <sub>4</sub>	f <sub>5</sub>	f <sub>6</sub>	f <sub>7</sub>
Guess	1	1.3	2	1.7	2.5	2	3.1
(MHK)	5	4	5	5	5	4	5
JPJ [13]	5	4	5	5	5	4	5
SB [24, 27]	5	4	5	5	5	4	5
CMB [5]	7	8	8	8	9	8	9
JR [14]	8	8	8	8	9	8	9
BG [8]	5	4	5	5	5	4	5
FC [4]	5	4	5	5	5	4	6
NM	9	8	8	8	9	8	10
HBM [10]	5	4	5	5	5	4	5

Results are summarized in Table 7 as shown, and new algorithms are comparable with all of the methods and in most cases give better or equal results.

### 6. Comparison of different methods

Finally, it should be noted that, like all other iterative methods, the new methods from the classes (HBM1)-(HBM5) have their own domains of validity and in certain circumstances should not be used.

Table 8: Comparison of different methods.

Method	$x_0$	$x_5$	COC	$ x_5 - x_4 $	$\ f(x_5)\ $	Time
$f_1$	1					
(MHK)		1.36523001341409	4	0.2e-173	0.4e-694	0.11
JPJ [13]		1.36523001341409	4	0.2e-152	0.4e-610	0.03
SB [24, 27]		1.36523001341409	4	0.1e-139	0.3e-558	0.05
CMB [5]		1.36523001341409	4	0.7e-144	0.1e-575	0.06
JR [14]		1.36523001341409	4	0.8e-168	0.1e-671	0.06
BG [8]		1.36523001341409	4	0.1e-126	0.3e-506	0.08
FC [4]		1.36523001341409	4	0.1e-98	0.3e-393	0.06
NM		1.365230013414096	2	2.1e-9	3.6e-19	
HBM[10]		1.365230013414096	3.99	1.5e-125	3.1e-505	
$f_2$	1.3					
(MHK)		1.40449164821534	4	0.1e-66	0.2e-267	0.19
JPJ [13]		1.40449164821534	4	0.1e-253	0.1e-1014	0.25
SB [24, 27]		1.40449164821534	4	0.1e-238	0.1e-954	0.23
CMB [5]		1.40449164821534	4	0.1e-244	0.5e-978	0.20
JR [14]		1.40449164821534	4	0.2e-261	0.8e-1046	0.23
BG [8]		1.40449164821534	4	0.6e-227	0.4e-908	0.17
FC [4]		1.40449164821534	4	0.4e-204	0.2e-816	0.22
NM		1.404491648215341	2	1.5e-15	4.8e-33	0.09
HBM[10]		1.404491648215341	4	6.1e-226	4.8e-907	0.21
$f_3$	2					
(MHK)		0.25753028543986	4	0.2e-65	0.5e-384	0.08
JPJ [13]		0.25753028543986	4	0.2e-101	0.6e-408	0.08
SB [24, 27]		0.25753028543986	4	0.4e-133	0.4e-535	0.09
CMB [5]		0.25753028543986	4	0.3e-117	0.4e-471	0.06
JR [14]		0.25753028543986	4	0.1e-108	0.3e-437	0.06
BG [8]		0.73908513321516	4	0.2e-157	0.2e-632	0.06
FC [4]		0.73908513321516	4	0.1e-138	0.2e-557	0.09
NM		0.257530285439860	2	9.8e-12	3.4e-25	0.04
HBM[10]		0.257530285439860	4	2.2e-156	2.9e-231	0.09
$f_4$	1.7					
(MHK)		0.73908513321516	4	0.4e-196	0.1e-786	0.22
JPJ [13]		0.73908513321516	4	0.1e-194	0.1e-780	0.19
SB [24, 27]		0.73908513321516	4	0.5e-192	0.6e-770	0.19
CMB [5]		0.73908513321516	4	0.4e-193	0.1e-774	0.19
JR [14]		0.73908513321516	4	0.3e-195	0.6e-783	0.20
BG [8]		0.73908513321516	4	0.9e-185	0.7e-741	0.17
FC [4]		0.73908513321516	4	0.2e-180	0.2e-723	0.19
NM		0.739085133215160	1.99	2.3e-14	2.0e-30	0.10
HBM[10]		0.739085133215160	3.99	9.7e-184	6.9e-740	0.21
$f_5$	2.5					
(MHK)		2	4	0.1e-116	0.3e-467	0.09
JPJ [13]		2	4	0.4e-110	0.1e-440	0.06
SB [24, 27]		2	4	0.3e-95	0.7e-383	0.06
CMB [5]		2	4	0.1e-108	0.4e-434	0.06
JR [14]		2	4	0.4e-10	0.3e-20	0.06
BG [8]		2	4	0.4e-89	0.3e-356	0.06
FC [4]		2	4	0.3e-77	0.2e-308	0.06
NM		2.000000000000011	2	1.0e-5	3.4e-12	0.03
HBM[10]		2	3.99	4.7e-88	3.9e-355	0.08
(MHK)		2.15443469003188	4	0.3e-72	0.2e-289	0.09
JPJ [13]		2.15443469003188	4	0.3e-257	0.1e-1101	0.06
SB [24, 27]		2.15443469003188	4	0.8e-259	0.1e-1035	0.03
CMB [5]		2.15443469003188	4	0.1e-256	0.1e-1061	0.06
JR [14]		2.15443469003188	4	0.4e-283	0.6e-1133	0.06
BG [8]		2.15443469003188	4	0.3e-247	0.4e-989	0.06
FC [4]		2.15443469003188	4	0.5e-224	0.6e-896	0.08
NM		2.154434690031883	2	2.2e-16	3.2e-33	0.03
HBM[10]		2.154434690031883	4	3.2e-246	4.0e-988	0.07
$f_7$	3.1					
(MHK)		3	4	0.7e-94	0.5e-373	0.14
JPJ [13]		3	4	0.2e-83	0.7e-331	0.09
SB [24, 27]		3	4	0.8e-65	0.3e-256	0.09
CMB [5]		3	4	0.6e-71	0.1e-280	0.09
JR [14]		3	4	0.1e-78	0.2e-311	0.09
BG [8]		3	4	0.1e-58	0.8e-232	0.08
FC [4]		3	4	0.5e-48	0.2e-188	0.09
NM		3.000000000089992	2.03	3.6e-4	1.1e-7	0.04
HBM[10]		3	3.99	1.0e-57	8.7e-231	0.10

## 7. Conclusion

In this work, we have proposed an optimal method of fourth order for finding the roots in complex domain and real domain of non-linear equations. The correlations between the attracting domains and the corresponding required a number of iterations, which also have been illustrated and discussed. Comparisons with several numerical methods and the use of complex dynamics and basins of attraction show that the new method gives good results.

## References

- [1] F. Ahmad, S. Hussain, S. Hussain, A. Rafiq, *New Efficient Fourth Order Method for Solving Nonlinear Equations*, Asia Pacific J. Multidiscip. Res., **1** (2013), 198–207. 1
- [2] N. Ahmad, V. P. Singh, *New Second Derivative Free Iterative Method For Solving Nonlinear Equations Using Halley's Method*, J. Sci. Arts, **1** (2019), 83–90. 1
- [3] M. S. M. Bahgat, *New Two-step Iterative Methods for Solving Nonlinear Equations*, J. Math. Res., **4** (2012), 128–131. 1
- [4] F. I. Chicharro, N. Garrido, J. R. Torregrosa, *Wide stability in a new family of optimal fourth-order iterative methods*, Comput. Math. Methods, **3** (2019), 1–14. 1, 2, 2, 4, 4, 5, 6, 5, 7, 8
- [5] C. B. Chun, M. Y. Lee, B. Neta, J. Dzunić, *On optimal fourth-order iterative methods free from second derivative and their dynamics*, Appl. Math. Comput., **218** (2012), 6427–6438. 1, 2, 1, 4, 4, 5, 6, 5, 7, 8
- [6] A. Cordero, J. L. Huesoa, E. Martinez, J. R. Torregrosa, *Generating optimal derivative free iterative methods for nonlinear equations by using polynomial interpolation*, Math. Comput. Modelling, **57** (2013), 1950–1956. 1
- [7] J. S. Feng, *A New Two-step Method for Solving Nonlinear Equations*, Int. J. Nonlinear Sci., **8** (2009), 40–44. 1
- [8] B. Ghanbari, *A new general fourth-order family of methods for finding simple roots of nonlinear equations*, J. King Saud Univ. Sci., **23** (2011), 395–398. 1, 2, 2, 4, 4, 5, 6, 5, 7, 8
- [9] M. A. Hafiz, *Solving nonlinear equations using Steffensen-type methods with optimal order of convergence*, Palest. J. Math., **3** (2014), 113–119. 1
- [10] M. A. Hafiz, M. S. M. Bahgat, *Solving Nonlinear Equations Using Two-Step Optimal Methods*, Ann. Review Chaos Theory Bifurcat. Dyn. Syst., **3** (2013), 1–11. 1, 2, 4, 4, 5, 6, 5, 7, 8
- [11] D. Hercegand, D. Herceg, *Some Fourth-Order Methods For Nonlinear Equations*, Novi Sad J. Math., **37** (2007), 241–247. 1
- [12] S. L. Huang, A. Rafiq, M. Rizwan Shahzad, F. Ali, *New higher order iterative methods for solving nonlinear equations*, Hacet. J. Math. Stat., **47** (2018), 77–91. 1
- [13] J. P. Jaiswal, *Some Class of Third- and Fourth-Order Iterative Methods for Solving Nonlinear Equations*, Hindawi Publishing Corporation, J. Appl. Math., **2014** (2014), 17 pages. 1, 2, 1, 4, 4, 5, 6, 5, 7, 8
- [14] P. Jarratt, *Some fourth order multipoint iterative methods for solving equations*, Math. Comput., **20** (1966), 434–437. 1, 2, 1, 4, 4, 5, 6, 5, 7, 8
- [15] S. K. Khattri, M. A. Noorb, E. Al-Said, *Unifying fourth-order family of iterative methods*, Appl. Math. Lett., **24** (2011), 1295–1300. 1, 5
- [16] H. T. Kung, J. F. Traub, *Optimal order of one-point and multipoint iteration*, J. Assoc. Comput. Mach., **21** (1974), 643–651. 2
- [17] S. F. Li, *Fourth-order iterative method without calculating the higher derivatives for nonlinear equation*, J. Algorithms Comput. Technol., **13** (2019), 8 pages. 1
- [18] B. Q. Liu, X. J. Zhou, *A new family of fourth-order methods for multiple roots of nonlinear equations*, Nonlinear Anal. Model. Control, **18** (2013), 143–152. 1
- [19] T. Lotfi, T. Eftekhari, *A New Optimal Eighth-Order Ostrowski-Type Family of Iterative Methods for Solving Nonlinear Equations*, Chin. J. Math. (N.Y.), **2014** (2014), 7 pages. 1
- [20] A. K. Maheshwari, *A fourth order iterative method for solving nonlinear equations*, Appl. Math. Comput., **211** (2009), 383–391. 1
- [21] M. A. Noor, W. A. Khan, *Fourth-Order Iterative Method Free from Second Derivative for Solving Nonlinear Equations*, Appl. Math. Sci. (Ruse), **6** (2012), 4617–4625. 1
- [22] M. S. Petković, *Comments on the Basto–Semiao–Calheiros root finding method*, Appl. Math. Comput., **184** (2007), 143–148. 3
- [23] R. K. Saeed, K. M. Aziz, *Iterative methods for solving nonlinear equations by using quadratic spline function*, Math. Sci. Lett., **2** (2013), 37–43. 1
- [24] R. Sharma, A. Bahl, *An Optimal Fourth Order Iterative Method for Solving Nonlinear Equations and Its Dynamics*, Hindawi Publishing Corporation, J. Complex Anal., **2015** (2015), 9 pages. 1, 2, 1, 4, 4, 5, 6, 5, 7, 8
- [25] M. K. Siddique, *A New Second-Order Iteration Method For Solving Nonlinear Equations*, Int. J. Sci. Eng. Res., **9** (2018), 1527–1531. 1
- [26] P. Sivakumar, J. Jayaraman, *Some New Higher Order Weighted Newton Methods for Solving Nonlinear Equation with Applications*, Math. Comput. Appl., **24** (2019), 21 pages. 1

- [27] P. Sivakumar, K. Madhu, J. Jayaraman, *Optimal fourth order methods with its multi-step version for nonlinear equation and their Basins of attraction*, SeMA J., **76** (2019), 559–579. 1, 2, 1, 4, 4, 5, 6, 5, 7, 8
- [28] O. S. Solaiman, I. Hashim, *Efficacy of Optimal Methods for Nonlinear Equations with Chemical Engineering Applications*, Math. Probl. Eng., **2019** (2019), 11 pages. 1
- [29] O. S. Solaiman, S. A. A. Karim, I. Hashim, *Optimal fourth- and eighth-order of convergence derivative-free modifications of King's method*, J. King Saud Univer. Sci., **31** (2019), 1499–1504. 1
- [30] R. Thukral, *A New Family of Fourth-Order Iterative Methods for Solving Nonlinear Equations With Multiple Roots*, J. Numer. Math. Stoch., **6** (2014), 37–44. 1
- [31] J. L. Varona, *Graphic And Numerical Comparison Between Iterative Methods*, Math. Intelligencer, **3** (2002), 37–46. 1
- [32] R. Villafuerte, J. Medina, R. A. Villafuerte, *An Iterative Method to Solve Nonlinear Equations*, J. Electr. Electron. Eng., **6** (2019), 14–22. 1
- [33] X. J. Zhou, X. Chen, Y. Z. Song, *Constructing higher-order methods for obtaining the multiple roots of nonlinear equations*, J. Comput. Appl. Math., **235** (2011), 4199–4206. 1