

HIGHER ORDER NONLINEAR EQUATION SOLVERS AND THEIR DYNAMICAL BEHAVIOR

SABIR YASIN¹, AMIR NASEEM

ABSTRACT. In this report we present new sixth order iterative methods for solving non-linear equations. The derivation of these methods is purely based on variational iteration technique. To check the validity and efficiency we compare of methods with Newton's method, Ostrowski's method, Traub's method and modified Halleys's method by solving some test examples. Numerical results shows that our developed methods are more effective. Finally, we compare polynomigraphs of our developed methods with Newton's method, Ostrowski's method, Traub's method and modified Halleys's method.

Mathematics Subject Classification: 37F50.

Key words and phrases: Non-linear equations, Newton's method, polynomiography.

1. Introduction

One of the most important problems is to find the values of x which satisfy the equation

$$f(x) = 0.$$

The solution of these problems has many applications in applied sciences. In order to solve these problems, various numerical methods have been developed using different techniques such as adomian decomposition, Taylor's series, perturbation method, quadrature formulas and variational iteration techniques

Received 03-11-2018. Accepted 05-12-2018.

¹ Corresponding Author

© 2018 Sabir Yasin, Amir Naseem. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22] and the references therein.

One of the most famous and oldest method for solving non linear equations is classical Newton's method which can be written as:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, n = 0, 1, 2, \dots \quad (1)$$

This is an important and basic method, which converges quadratically [12]. Modifications of Newton's method gave various iterative methods with better convergence order. Some of them are given in [3, 8, 9, 10, 11, 18, 19] and the references therein.

In [23], Traub developed following Double Newton's method:

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},$$

$$x_{n+1} = y_n - \frac{f(y_n)}{f'(y_n)}, n = 0, 1, 2, 3, \dots$$

This method is also known as Traub's Method.

Ostrowsk' method (see [24, 25, 26]) is also a well known iterative method which has forth order convergence.

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},$$

$$x_{n+1} = y_n - \frac{f(x_n)f(y_n)}{f'(x_n)f(x_n) - 2f(y_n)}, n = 0, 1, 2, 3, \dots$$

Noor et al. [27] developed modified Halley's method which has fifth-order convergence

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},$$

$$x_{n+1} = y_n - \frac{f(x_n)f(y_n)f'(y_n)}{2f(x_n)f'^2(y_n) - f'^2(x_n)f(y_n) + f'(x_n)f'(y_n)f(y_n)}, n = 0, 1, 2, 3, \dots$$

In this paper, we develop three new iterative methods using variational iteration technique. The variational iteration technique was developed by He [14]. Using this technique, Noor and Shah [17] has suggested and analyzed some iterative methods for solving the nonlinear equations. The purpose of this technique was to solve a variety of diverse problems [14, 15]. Now we have applied the described technique to obtain higher-order iterative methods. We also discuss the convergence criteria of these new iterative methods. Several examples are given to show the performance of our proposed methods as compare to the other similar existing methods. We also compare polynomigraphs of our developed methods with Newton's method, Ostrowski's method, Traub's method and modified Halley's method.

2. Construction OF Iterative Methods using Variational Technique

In this section, we develop some new sixth order iterative methods for solving non linear equations. By using variational iteration technique, we develop the main recurrence relation from which we derive the new iterative methods for solving non linear equations by considering some special cases of the auxiliary functions g . These are multi-step methods consisting of predictor and corrector steps. The convergence of our methods is better than the one-step methods. Now consider the non-linear equation of the form

$$f(x) = 0. \quad (2)$$

Suppose that α is the simple root and γ is the initial guess sufficiently close to α . Let $g(x)$ be any arbitrary function and λ be a parameter which is usually called the Lagranges multiplier and can be identified by the optimality condition. Consider the auxiliary function

$$H(x) = \psi(x) + \lambda[f(\psi(x))g(\psi(x))], \quad (3)$$

where $\psi(x)$ is the arbitrary auxiliary function of order p with $p \geq 1$.

Using the optimality criteria, we can obtain the value of λ from (3) as:

$$\lambda = -\frac{\psi(x)}{g'(\psi(x))f(\psi(x)) + g(\psi(x))f'(\psi(x))}. \quad (4)$$

From (3) and (4), we get

$$H(x) = \psi(x) - \frac{f(\psi(x))g(\psi(x))}{[f'(\psi(x))g(\psi(x)) + f(\psi(x))g'(\psi(x))]} \quad (5)$$

Now we are going to apply (5) for constructing a general iterative scheme for iterative methods. For this, suppose that

$$\psi(x) = y = x - \frac{f(x)}{f'(x)} - \frac{f^2(x)f''(x)}{2f'^3(x)} - \frac{f^3(x)f'''(x)}{6f'^4(x)}. \quad (6)$$

Which is well known Abbasbanday's method of 3rd order of convergence. With the help of (5) and (6), we can write

$$H(x) = y - \frac{f(y)g(y)}{[f'(y)g(y) + f(y)g'(y)]}, \quad (7)$$

if α is the root of $f(x)$, then for $x = \alpha$, we can write:

$$\frac{g(y)}{g'(y)} = \alpha - \frac{f(\alpha)}{f'(\alpha)} - \frac{f^2(\alpha)f''(\alpha)}{2f'^3(\alpha)} - \frac{f^3(\alpha)f'''(\alpha)}{6f'^4(\alpha)} = \frac{g(\alpha)}{g'(\alpha)}. \quad (8)$$

Also,

$$\frac{g(x)}{g'(x)} = \frac{g(\alpha)}{g'(\alpha)}. \quad (9)$$

With the help of (8) and (9), we get

$$\frac{g(y)}{g'(y)} = \frac{g(x)}{g'(x)}. \quad (10)$$

Using (10) in (7), we obtain

$$H(x) = y - \frac{f(y)g(x)}{[f'(y)g(x) + f(y)g'(x)]}. \quad (11)$$

Which enable us to define the following iterative scheme:

$$x_{n+1} = y_n - \frac{f(y_n)g(x_n)}{[f'(y_n)g(x_n) + f(y_n)g'(x_n)]} \quad (12)$$

where $y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}$. Relation(12) is the main and general iterative scheme, which we use to deduce iterative methods for solving non-linear equations by considering some special cases of the auxiliary functions g .

2.1. Case 1. Let $g(x_n) = e^{(\beta x_n)}$, then $g'(x_n) = \beta g(x_n)$. Using these values in (12), we obtain the following algorithm.

Using these values in (12), we obtain the following algorithm.

Algorithm 2.1. For a given x_0 , compute the approximate solution x_{n+1} by the following iterative schemes:

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}, n = 0, 1, 2, \dots,$$

$$x_{n+1} = y_n - \frac{f(y_n)}{[f'(y_n) + \beta f(y_n)]}.$$

2.2. Case 2. Let $g(x_n) = e^{\beta f(x_n)}$, then $g'(x_n) = \beta f'(x_n)g(x_n)$. Using these values in (12), we obtain the following algorithm.

Algorithm 2.2. For a given x_0 , compute the approximate solution x_{n+1} by the following iterative schemes:

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}, n = 0, 1, 2, \dots,$$

$$x_{n+1} = y_n - \frac{f(y_n)}{[f'(y_n) + \beta f(y_n)f'(x_n)]}.$$

2.3. Case 3. Let $g(x_n) = e^{-\frac{\beta}{f(x_n)}}$, then $g'(x_n) = \beta \frac{f'(x_n)}{f^2(x_n)}g(x_n)$. Using these values in (12), we obtain the following algorithm.

Algorithm 2.3. For a given x_0 , compute the approximate solution x_{n+1} by the following iterative schemes:

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}, n = 0, 1, 2, \dots,$$

$$x_{n+1} = y_n - \frac{f^2(x_n)f(y_n)}{[f^2(x_n)f'(y_n) + \beta f'(x_n)f(y_n)]}.$$

By assuming different values of β , we can obtain different iterative methods. To obtain best results in all above algorithms, choose such values of β that make the denominator non-zero and greatest in magnitude.

3. Convergence Analysis

In this section, we discuss the convergence order of the main and general iteration scheme (12).

Theorem 3.1. *Suppose that α is a root of the equation $f(x) = 0$. If $f(x)$ is sufficiently smooth in the neighborhood of α , then the convergence order of the main and general iteration scheme, described in relation (12) is at least six.*

Proof. To analysis the convergence of the main and general iteration scheme, described in relation (12), suppose that α is a root of the equation $f(x) = 0$ and e_n be the error at nth iteration, then $e_n = x_n - \alpha$ and by using Taylor series expansion, we have

$$\begin{aligned} f(x) &= f'(\alpha)e_n + \frac{1}{2!}f''(\alpha)e_n^2 + \frac{1}{3!}f'''(\alpha)e_n^3 + \frac{1}{4!}f^{(iv)}(\alpha)e_n^4 + \frac{1}{5!}f^{(v)}(\alpha)e_n^5 \\ &+ \frac{1}{6!}f^{(vi)}(\alpha)e_n^6 + O(e_n^7) \\ f(x) &= f'(\alpha)[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + c_6e_n^6 + O(e_n^7)] \end{aligned} \quad (13)$$

$$\begin{aligned} f'(x_n) &= f'(\alpha)[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + 6c_6e_n^5 + 7c_7e_n^6 \\ &+ O(e_n^7)] \end{aligned} \quad (14)$$

$$\begin{aligned} f''(x_n) &= f'(\alpha)[2c_2 + 6c_3e_n + 12c_4e_n^2 + 20c_5e_n^3 + 30c_6e_n^4 \\ &+ 42c_7e_n^5 + 56c_8e_n^6 + O(e_n^7)] \end{aligned} \quad (15)$$

$$\begin{aligned} f'''(x_n) &= f'(\alpha)[6c_3 + 24c_4e_n + 60c_5e_n^2 + 120c_6e_n^3 + 210c_7e_n^4 \\ &+ 336c_8e_n^5 + 504c_9e_n^6 + O(e_n^7)]. \end{aligned} \quad (16)$$

Where

$$c_n = \frac{1}{n!} \frac{f^{(n)}(\alpha)}{f'(\alpha)}.$$

With the help of 13, 14 and 15, we get

$$\begin{aligned} y_n &= \alpha + (-2c_3 + 2c_2^2)e_n^3 + (-7c_4 + 17c_2c_3 - 9c_2^3)e_n^4 \\ &+ (-16c_5 + 44c_2c_4 + 24c_3^2 - 82c_3c_2^2 + 30c_2^4)e_n^5 \\ &+ (-30c_6 + 90c_2c_5 + 104c_3c_4 - 202c_2c_3^2 + 314c_3c_2^3 - 188c_4c_2^2 - 88c_2^5)e_n^6 \\ &+ O(e_n^7) \end{aligned} \quad (17)$$

$$\begin{aligned} f(y_n) &= f'(\alpha)[(-2c_3 + 2c_2^2)e_n^3 + (-7c_4 + 17c_2c_3 - 9c_2^3)e_n^4 \\ &+ (-16c_5 + 44c_2c_4 + 24c_3^2 - 82c_3c_2^2 + 30c_2^4)e_n^5 \\ &+ (-198c_2c_3^2 + 306c_3c_2^3 - 84c_2^5 - 30c_6 + 90c_2c_5 + 104c_3c_4 - 188c_4c_2^2)e_n^6 \end{aligned}$$

$$+ O(e_n^7)] \quad (18)$$

$$\begin{aligned} f'(y_n) &= f'(\alpha)[1 + (-4c_2c_3 + 4c_2^3)e_n^3 + (34c_3c_2^2 - 18c_2^4 - 14c_2c_4)e_n^4 \\ &+ (-32c_2c_5 + 88c_4c_2^2 + 48c_2c_3^2 - 164c_3c_2^3 + 60c_2^5)e_n^5 \\ &+ (-60c_2c_6 + 180c_5c_2^2 + 208c_4c_2c_3 - 428c_3^2c_2^2 + 640c_3c_2^4 - 376c_4c_2^3 \\ &- 176c_2^6 + 12c_3^3)e_n^6 + O(e_n^7)] \end{aligned} \quad (19)$$

$$\begin{aligned} g(x_n) &= g(\alpha) + g'(\alpha)e_n + \frac{g''(\alpha)}{2!}e_n^2 + \frac{g'''(\alpha)}{3!}e_n^3 + \frac{g^{(iv)}(\alpha)}{4!}e_n^4 + \frac{g^{(v)}(\alpha)}{5!}e_n^5 \\ &+ \frac{g^{(vi)}(\alpha)}{6!}e_n^6 + O(e_n^7) \end{aligned} \quad (20)$$

$$\begin{aligned} g'(x_n) &= g'(\alpha) + g''(\alpha)e_n + \frac{g'''(\alpha)}{2!}e_n^2 + \frac{g^{(iv)}(\alpha)}{3!}e_n^3 + \frac{g^{(v)}(\alpha)}{4!}e_n^4 + \frac{g^{(vi)}(\alpha)}{5!}e_n^5 \\ &+ \frac{g^{(vii)}(\alpha)}{6!}e_n^6 + O(e_n^7). \end{aligned} \quad (21)$$

Using equations (13 – 20) in general iteration scheme(12), we get:

$$x_{n+1} = \alpha + \frac{4(-c_3 + c_2^2)^2[g(\alpha)c_2 + g'(\alpha)]}{g(\alpha)}e_n^6 + O(e_n^7),$$

which implies that

$$e_{n+1} = \frac{4(-c_3 + c_2^2)^2[g(\alpha)c_2 + g'(\alpha)]}{g(\alpha)}e_n^6 + O(e_n^7).$$

The above relation shows that the main and general iteration scheme(12) is of sixth order of convergence and all iterative methods deduce from it have also convergence of order six.

□

4. Applications

In this section we included some nonlinear functions to illustrate the efficiency of our developed algorithms for $\beta = 1$. We compare our developed algorithms with Newton's method (NM)[12], Ostrowski's method (OM) [7], Traub's method (TM)[12], and modified Halley's method (MHM) [28]. We used $\varepsilon = 10^{-15}$. The following stopping criteria is used for computer programs:

- (1) $|x_{n+1} - x_n| < \varepsilon$.
- (2) $|f(x_{n+1})| < \varepsilon$.

Table 1: Comparison of various iterative methods				
$f_1 = x^3 + 4x^2 - 10, x_0 = -0.7.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	20	40	$1.056394e - 24$	
OM	51	153	$9.750058e - 26$	
TM	10	30	$1.056394e - 24$	
MHM	30	90	$2.819181e - 35$	1.365230013414096845760806828980
Algorithm 2.1	3	9	$1.037275e - 74$	
Algorithm 2.2	3	9	$7.962504e - 39$	
Algorithm 2.3	3	9	$2.748246e - 39$	

Table 2: Comparison of various iterative methods				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
$f_2 = \ln(x) + x, x_0 = 2.6.$				
NM	8	16	$6.089805e - 28$	
OM	4	12	$3.421972e - 53$	
TM	4	12	$6.089805e - 28$	
MHM	4	21	$4.247135e - 28$	0.567143290409783872999968662210
Algorithm 2.1	3	9	$1.034564e - 21$	
Algorithm 2.2	3	9	$1.994520e - 38$	
Algorithm 2.3	3	9	$7.258268e - 15$	

Table 3: Comparison of various iterative methods				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
$f_3 = \ln x + \cos(x), x_0 = 0.1.$				
NM	6	12	$2.313773e - 18$	
OM	3	9	$5.848674e - 26$	
TM	3	9	$2.313773e - 18$	
MHM	4	12	$1.281868e - 60$	0.397748475958746982312388340926
Algorithm 2.1	2	6	$2.915840e - 34$	
Algorithm 2.2	2	6	$3.380437e - 32$	
Algorithm 2.3	2	6	$5.839120e - 23$	

Table 4: Comparison of various iterative methods				
$f_4 = xe^x - 1, x_0 = 1.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	5	10	$8.478184e - 17$	
OM	3	9	$8.984315e - 40$	
TM	3	9	$2.130596e - 33$	
MHM	3	9	$1.116440e - 68$	0.567143290409783872999968662210
Algorithm 2.1	2	6	$5.078168e - 24$	
Algorithm 2.2	3	9	$4.315182e - 82$	
Algorithm 2.3	3	9	$1.027667e - 46$	

Table 5: Comparison of various iterative methods				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
$f_5 = x^3 - 1, x_0 = 2.3.$				
NM	7	14	$9.883568e - 25$	
OM	4	12	$2.619999e - 56$	
TM	4	12	$3.256164e - 49$	
MHM	3	9	$3.233029e - 46$	1.00000000000000000000000000000000
Algorithm 2.1	2	6	$1.472038e - 16$	
Algorithm 2.2	3	9	$1.847778e - 15$	
Algorithm 2.3	3	9	$1.620804e - 18$	

Table (1-5) Shows the numerical comparisons of Newton's method, Ostrowski's method, Traub's method, modified Halley's method and our developed methods. The columns represent the number of iterations N and the number of functions or derivatives evaluations N_f required to meet the stopping criteria, and the magnitude $|f(x)|$ of $f(x)$ at the final estimate x_n .

5. Polynomiography

Polynomials are one of the most significant objects in many fields of mathematics. Polynomial root-finding has played a key role in the history of mathematics. It is one of the oldest and most deeply studied mathematical problems. The last interesting contribution to the polynomials root finding history was made by Kalantari [29], who introduced the polynomiography. As a method which generates nice looking graphics, it was patented by Kalantari in USA in 2005 [30, 31]. Polynomiography is defined to be "the art and science of visualization in approximation of the zeros of complex polynomials, via fractal and non fractal images created using the mathematical convergence properties of iteration functions" [29]. An individual image is called a "polynomiograph". Polynomiography combines both art and science aspects.

Polynomiography gives a new way to solve the ancient problem by using new algorithms and computer technology. Polynomiography is based on the use of one or an infinite number of iteration methods formulated for the purpose of approximation of the root of polynomials e.g. Newton's method, Halley's method etc. The word "fractal", which partially appeared in the definition of polynomiography, was coined by the famous mathematician Benoit Mandelbrot [32]. Both fractal images and polynomiographs can be obtained via different iterative schemes. Fractals are self-similar has typical structure and independent of scale. On the other hand, polynomiographs are quite different. The "polynomiographer" can control the shape and designed in a more predictable way by using different iteration methods to the infinite variety of complex polynomials. Generally, fractals and polynomiographs belong to different classes of graphical objects. Polynomiography has diverse applications in math, science, education, art and design. According to Fundamental Theorem of Algebra, any complex

polynomial with complex coefficients $\{a_n, a_{n-1}, \dots, a_1, a_0\}$:

$$p(z) = a_n z^n + a_{n-1} z^{n-1} + \dots + a_1 z + a_0 \quad (22)$$

or by its zeros (roots) $\{r_1, r_2, \dots, r_{n-1}, r_n\}$:

$$p(z) = (z - r_1)(z - r_2)\dots(z - r_n) \quad (23)$$

of degree n has n roots (zeros) which may or may not be distinct. The degree of polynomial describes the number of basins of attraction and placing roots on the complex plane manually localization of basins can be controlled.

Usually, polynomiographs are colored based on the number of iterations needed to obtain the approximation of some polynomial root with a given accuracy and a chosen iteration method. The description of polynomiography, its theoretical background and artistic applications are described in [29, 30, 31].

5.1. Iteration. During the last century, the different numerical techniques for solving nonlinear equation $f(x) = 0$ have been successfully applied. Now we define our developed algorithms as:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}, n = 0, 1, 2, \dots, \\ x_{n+1} &= y_n - \frac{f(y_n)}{[f'(y_n) + \beta f(y_n)]}, \end{aligned}$$

which we call algorithm (2.1) for solving nonlinear equations.

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}, n = 0, 1, 2, \dots, \\ x_{n+1} &= y_n - \frac{f(y_n)}{[f'(y_n) + \beta f(y_n)f'(x_n)]} \end{aligned}$$

which we call algorithm (2.2) for solving nonlinear equations.

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} - \frac{f^3(x_n)f'''(x_n)}{6f'^4(x_n)}, n = 0, 1, 2, \dots, \\ x_{n+1} &= y_n - \frac{f^2(x_n)f(y_n)}{[f^2(x_n)f'(y_n) + \beta f'(x_n)f(y_n)]} \end{aligned}$$

which we call algorithm (2.3) for solving nonlinear equations.

Let $p(z)$ be the complex polynomial, then

$$\begin{aligned} y_n &= z_n - \frac{p(z_n)}{p'(z_n)} - \frac{p^2(z_n)p''(z_n)}{2p'^3(z_n)} - \frac{p^3(z_n)p'''(z_n)}{6p'^4(z_n)}, n = 0, 1, 2, \dots, \\ z_{n+1} &= y_n - \frac{p(y_n)}{[p'(y_n) + \beta p(y_n)]} \end{aligned}$$

which is algorithm (2.1) for solving nonlinear complex equations.

$$\begin{aligned} y_n &= z_n - \frac{p(z_n)}{p'(z_n)} - \frac{p^2(z_n)p''(z_n)}{2p'^3(z_n)} - \frac{p^3(z_n)p'''(z_n)}{6p'^4(z_n)}, n = 0, 1, 2, \dots, \\ z_{n+1} &= y_n - \frac{p(y_n)}{[p'(y_n) + \beta p(y_n)p'(z_n)]} \end{aligned}$$

which is algorithm (2.2) for solving nonlinear complex equations.

$$\begin{aligned} y_n &= z_n - \frac{p(z_n)}{p'(z_n)} - \frac{p^2(z_n)p''(z_n)}{2p'^3(z_n)} - \frac{p^3(z_n)p'''(z_n)}{6p'^4(z_n)}, n = 0, 1, 2, \dots, \\ z_{n+1} &= y_n - \frac{p^2(z_n)p(y_n)}{[p^2(z_n)p'(y_n) + \beta p'(z_n)f(y_n)]} \end{aligned}$$

Which is algorithm (2.3) for solving nonlinear complex equations.

Where $z_o \in \mathbb{C}$ is a starting point. The sequence $\{z_n\}_{n=0}^{\infty}$ is called the orbit of the point z_o converges to a root z^* of p then, we say that z_o is attracted to z^* . A set of all such starting points for which $\{z_n\}_{n=0}^{\infty}$ converges to root z^* is called the basin of attraction of z^* .

6. Convergence test

In the numerical algorithms that are based on iterative processes we need a stop criterion for the process, that is, a test that tells us that the process has converged or it is very near to the solution. This type of test is called a convergence test. Usually, in the iterative process that use a feedback, like the root finding methods, the standard convergence test has the following form:

$$|z_{n+1} - z_n| < \varepsilon, \quad (24)$$

where z_{n+1} and z_n are two successive points in the iteration process and $\varepsilon > 0$ is a given accuracy. In this paper we also use the stop criterion (24).

7. Applications

In this section we present some examples of polynomiographs for different complex polynomials equation $p(z) = 0$ and some special polynomials using our developed algorithms. The different colors of a images depend upon number of iterations to reach a root with given accuracy $\varepsilon = 0.001$. One can obtain infinitely many nice looking polynomiographs by changing parameter k , where k is the upper bound of the number of iterations.

7.0.1. Polynomiographs Of Different Complex Polynomial. In this section, we present polynomiographs of the following complex polynomials, using our developed methods for $\beta = 1$.

Example 7.1. Polynomiograph for $z^2 - 1 = 0$, via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

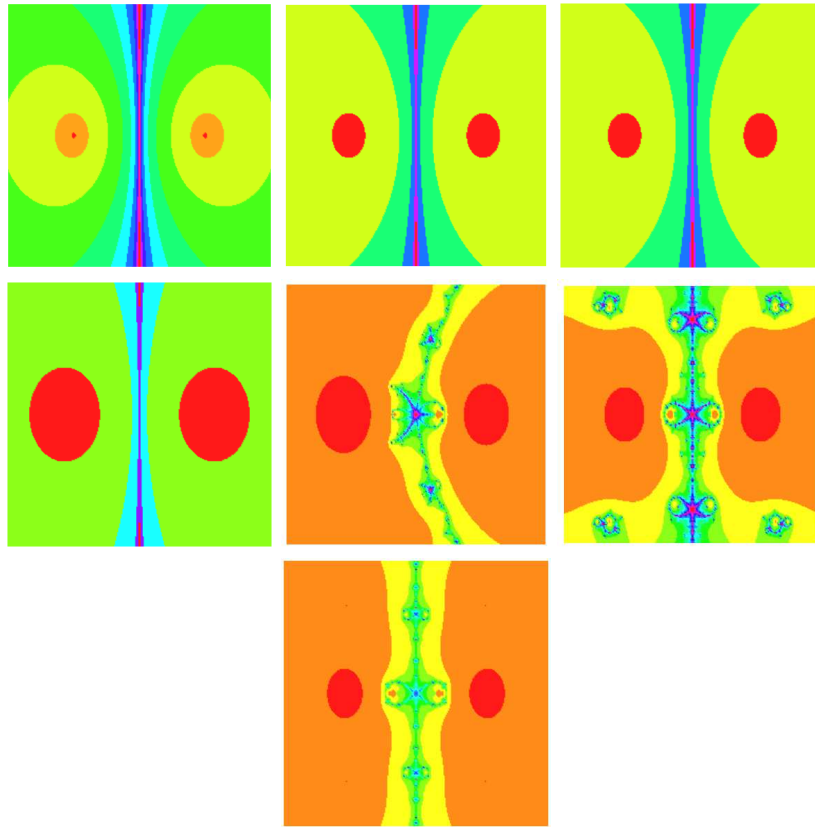


Figure 1. Polynomiographs of $z^2 - 1 = 0$.

Example 7.2. Polynomiograph for $z^3 - 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

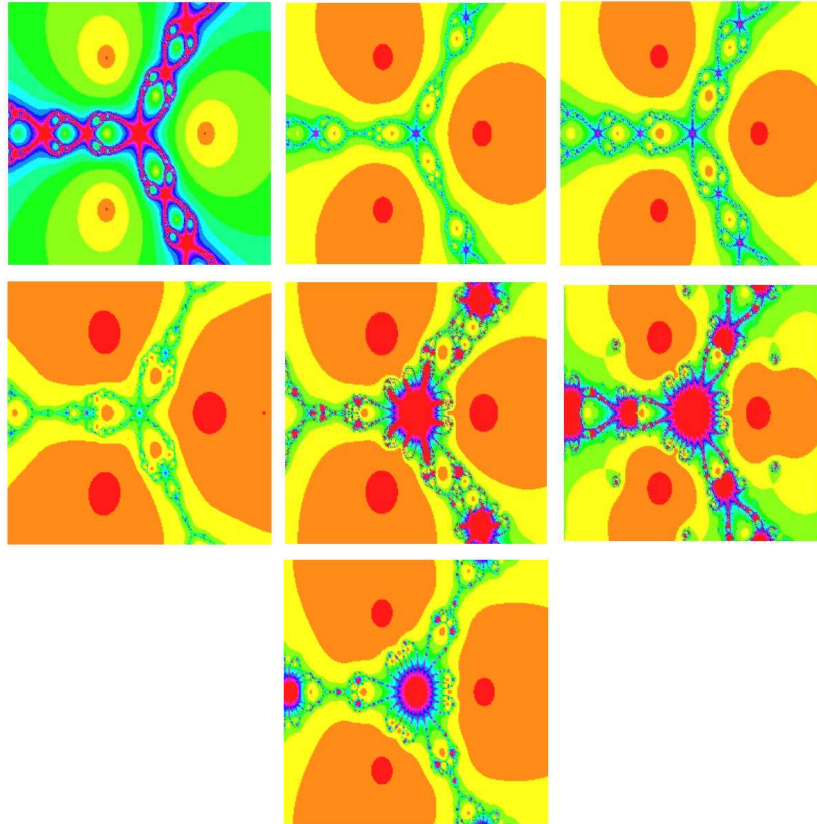


Figure 2. Polynomiographs of $z^3 - 1 = 0$

Example 7.3. Polynomiograph for $z^3 - z^2 + 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

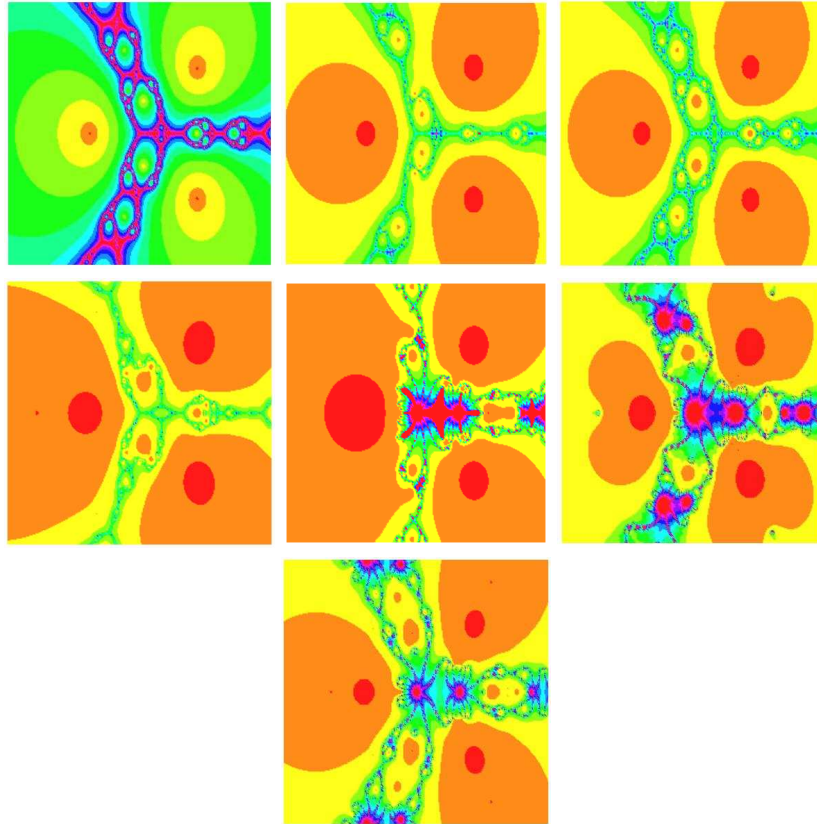


Figure 3. Polynomiographs of $z^3 - z^2 + 1 = 0$

Example 7.4. Polynomiograph for $z^4 - 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

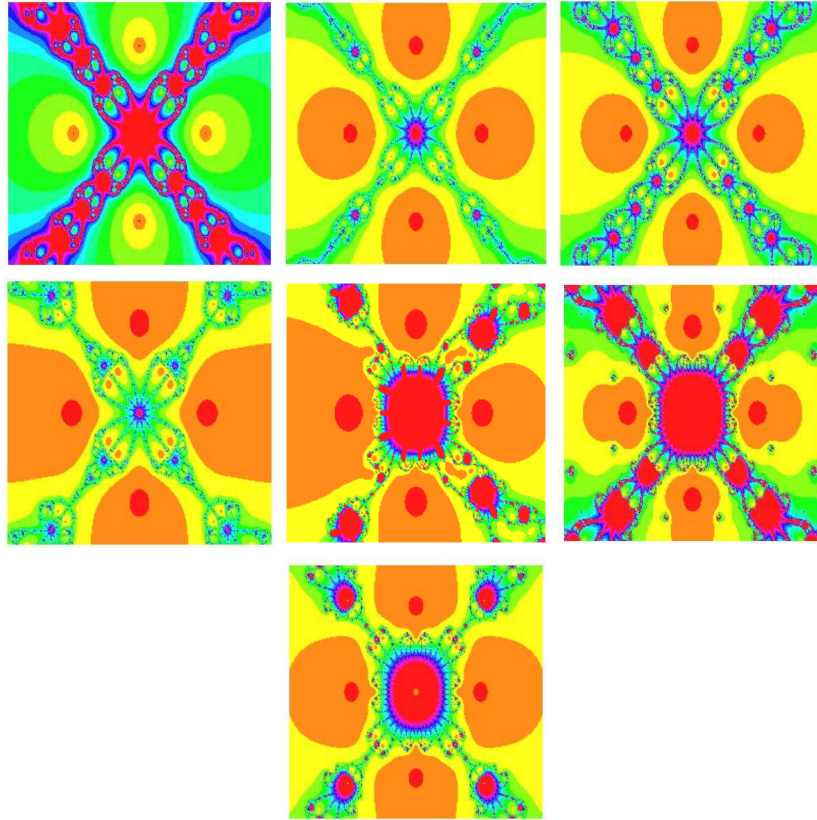


Figure 4. Polynomiographs of $z^4 - 1 = 0$

Example 7.5. Polynomiograph for $z^4 - z^2 - 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

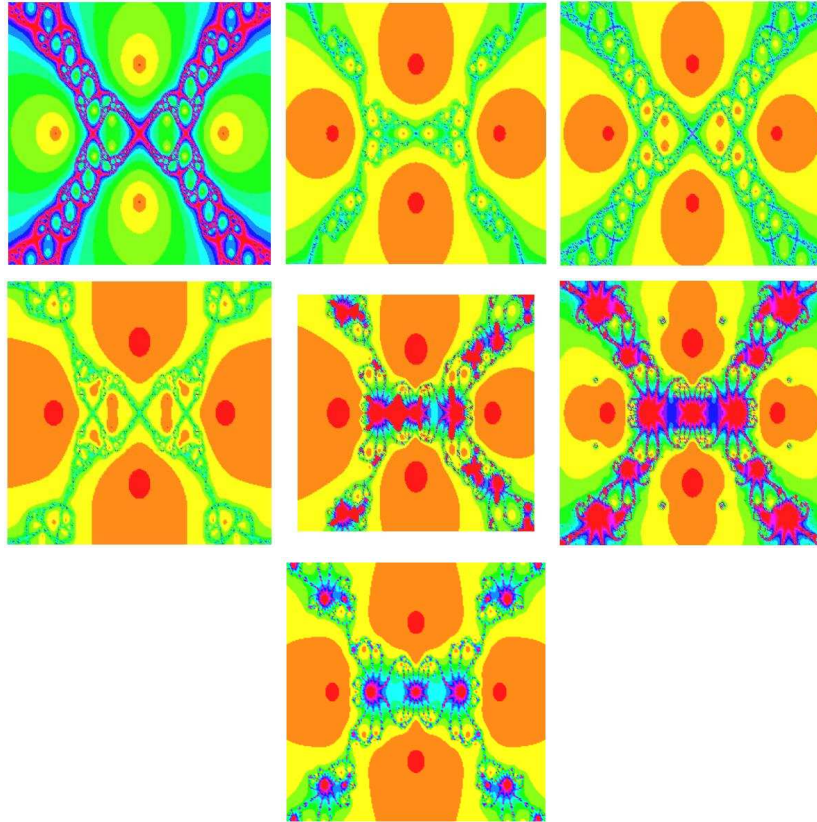


Figure 5. Polynomiographs of $z^4 - z^2 - 1 = 0$

Example 7.6. Polynomiograph for $z^5 - 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

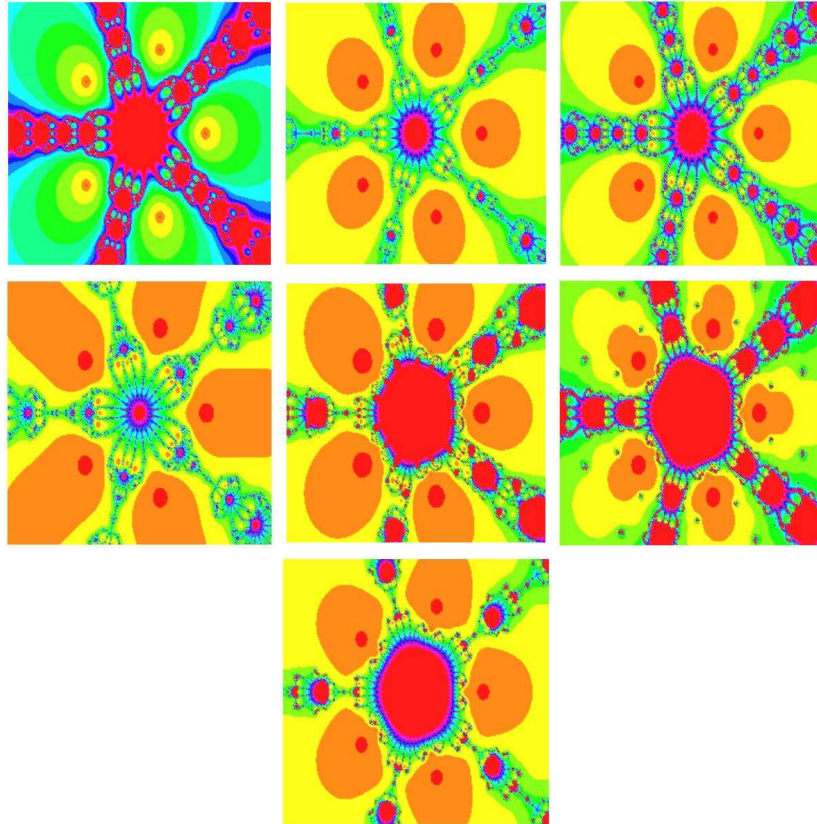


Figure 6. Polynomiographs of $z^5 - 1 = 0$

Example 7.7. Polynomiograph for $z^5 - z^3 + 2 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

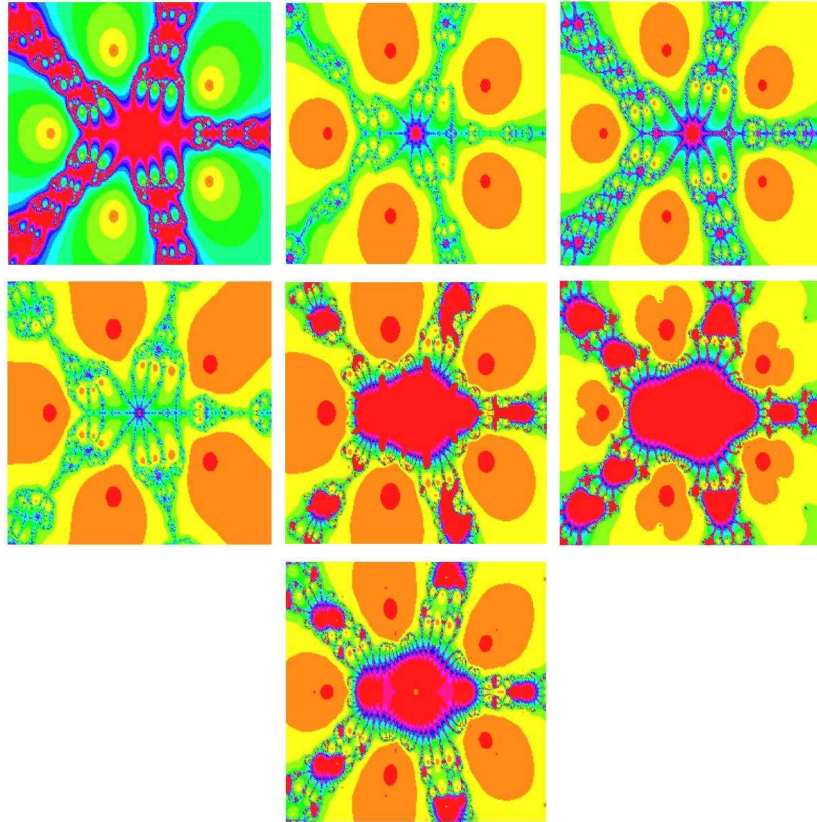


Figure 7. Polynomiographs of $z^5 - z^3 + 2 = 0$

Example 7.8. Polynomiograph for $z^6 - 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

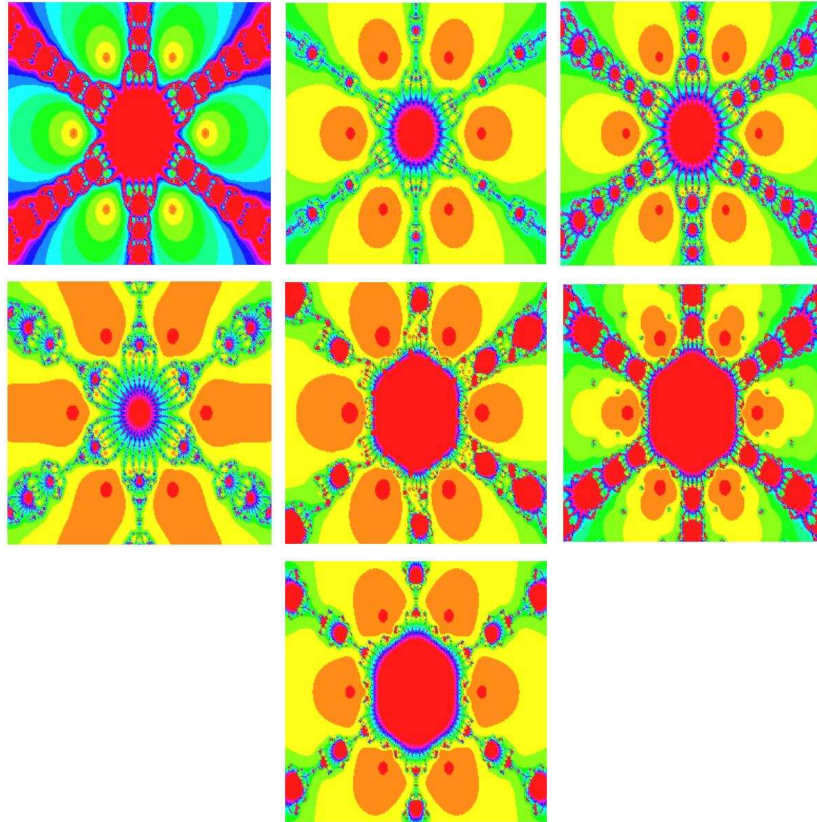


Figure 8. Polynomiographs of $z^6 - 1 = 0$

Example 7.9. Polynomiograph for $z^6 - z^4 + 4 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

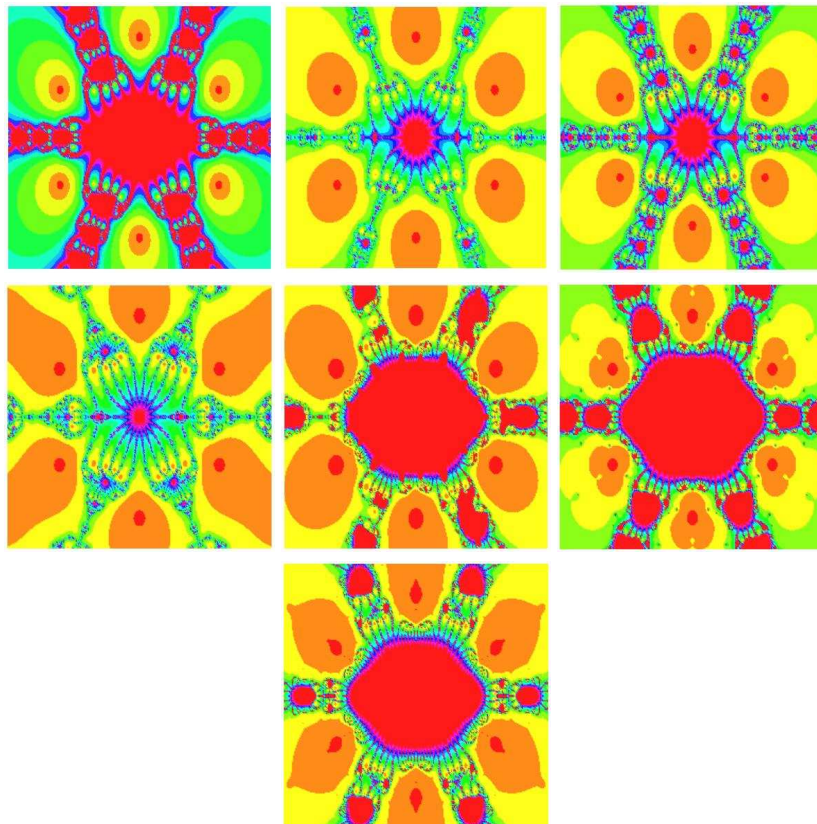


Figure 9. Polynomiographs of $z^6 - z^4 + 4 = 0$

Example 7.10. Polynomiograph for $z^7 - 1 = 0$ via Newton's method (row one left figure), Ostrowski's method (row one middle figure), Traub's method (row one right figure), modified Halley's method (row two left figure), Algorithm (2.1) (row two middle figure), Algorithm (2.2) (row two right figure) and Algorithm (2.3) (row three) are given below

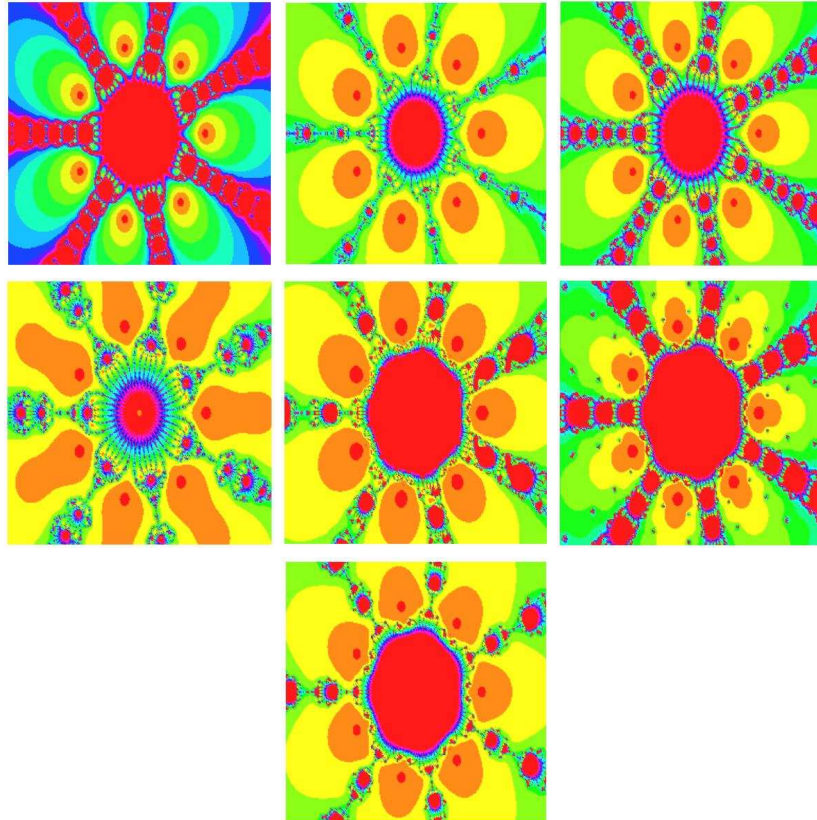


Figure 9. Polynomiographs of $z^7 - 1 = 0$

8. Conclusions

We have established three new sixth order iterative methods for solving non linear functions. We solved some test examples to check the efficiency of our developed methods. Table 1-5 shows that our methods perform better than Newton's method, Ostrowski's method, Traub's method and modified Halley's method. We also compare our methods with Newton's method, Ostrowski's method, Traub's method and modified Halley's method by presenting polynomiographs of different complex polynomials.

Competing Interests

The authors declare that they have no competing interests.

REFERENCES

1. Nazeer, W., Naseem, A., Kang, S. M., & Kwun, Y. C. (2016). Generalized Newton Raphson's method free from second derivative. *J. Nonlinear Sci. Appl.*, 9 (2016), 2823, 2831.

2. Nazeer, W., Tanveer, M., Kang, S. M., & Naseem, A. (2016). A new Householder's method free from second derivatives for solving nonlinear equations and polynomiography. *J. Non-linear Sci. Appl.*, 9, 998-1007.
3. Chun, C. (2006). Construction of Newton-like iteration methods for solving nonlinear equations. *Numerische Mathematik*, 104(3), 297-315.
4. Burden, R. L., Faires, J. D., & Reynolds, A. C. (2001). *Numerical analysis*.
5. Stoer, J., & Bulirsch, R. (2013). *Introduction to numerical analysis* (Vol. 12). Springer Science & Business Media.
6. Quarteroni, A., Sacco, R., & Saleri, F. (2010). *Numerical mathematics* (Vol. 37). Springer Science & Business Media.
7. Chen, D., Argyros, I. K., & Qian, Q. S. (1993). A note on the Halley method in Banach spaces. *Applied Mathematics and Computation*, 58(2-3), 215-224.
8. Gutierrez, J. M., & Hernandez, M. A. (2001). An acceleration of Newton's method: Super-Halley method. *Applied Mathematics and Computation*, 117(2-3), 223-239.
9. Gutiérrez, J. M., & Hernandez, M. A. (1997). A family of Chebyshev-Halley type methods in Banach spaces. *Bulletin of the Australian Mathematical Society*, 55(1), 113-130.
10. Householder, A. S. (1970). *The numerical treatment of a single nonlinear equation*. McGraw-Hill, New York.
11. Sebah, P., & Gourdon, X. (2001). Newtons method and high order iterations. *Numbers Comput.*, 1-10.
12. Traub, J. F. (1982). *Iterative methods for the solution of equations* (Vol. 312). American Mathematical Soc..
13. Inokuti, M., Sekine, H., & Mura, T. (1978). General use of the Lagrange multiplier in nonlinear mathematical physics. *Variational method in the mechanics of solids*, 33(5), 156-162.
14. He, J. H. (2007). Variational iteration methodsome recent results and new interpretations. *Journal of computational and applied mathematics*, 207(1), 3-17.
15. He, J. H. (1999). Variational iteration methoda kind of non-linear analytical technique: some examples. *International journal of non-linear mechanics*, 34(4), 699-708.
16. Noor, M. A. (2007). New classes of iterative methods for nonlinear equations. *Applied Mathematics and Computation*, 191(1), 128-131.
17. Noor, M. A., & Shah, F. A. (2009). Variational iteration technique for solving nonlinear equations. *Journal of Applied Mathematics and Computing*, 31(1-2), 247-254.
18. Kou, J. (2007). The improvements of modified Newtons method. *Applied Mathematics and Computation*, 189(1), 602-609.
19. Abbasbandy, S. (2003). Improving NewtonRaphson method for nonlinear equations by modified Adomian decomposition method. *Applied Mathematics and Computation*, 145(2-3), 887-893.
20. Naseem, A., Awan, M. W., & Nazeer, W. (2016). Dynamics of an iterative method for nonlinear equations. *Sci. Int.(Lahore)*, 28(2), 819-823.
21. Naseem, A., Nazeer, W., & Awan, M. W. (2016). Polynomiography via modified Abbasbandays method. *Sci. Int.(Lahore)*, 28(2), 761-766.
22. Naseem, A., Attari, M. Y., & Awan, M. W. (2016). Polynomiography via modified Golbabi and Javidi's method. *Science International*, 28(2).*Sci. Int.(Lahore)*, 28(2), 867-871.
23. Traub, J. F. (1982). *Iterative methods for the solution of equations* (Vol. 312). American Mathematical Soc..
24. Nawaz, M., Naseem, A., Nazeer, W. (2018). New iterative methods using variational iteration technique and their dynamical behavior. *Open. Math. Anal.*, 2(2), 01-09.
25. Noor, M. A., & Shah, F. A. (2009). Variational iteration technique for solving nonlinear equations. *Journal of Applied Mathematics and Computing*, 31(1-2), 247-254.

26. Noor, M. A., Shah, F. A., Noor, K. I., & Al-Said, E. (2011). Variational iteration technique for finding multiple roots of nonlinear equations. *Scientific Research and Essays*, 6(6), 1344-1350.
27. Noor, M. A., Khan, W. A., & Hussain, A. (2007). A new modified Halley method without second derivatives for nonlinear equation. *Applied Mathematics and Computation*, 189(2), 1268-1273.
28. Noor, K. I., & Noor, M. A. (2007). Predictorcorrector Halley method for nonlinear equations. *Applied Mathematics and Computation*, 188(2), 1587-1591.
29. Kalantari, B. (2005). U.S. Patent No. 6,894,705. Washington, DC: U.S. Patent and Trade-mark Office.
30. Kalantari, B. (2005). Polynomiography: from the fundamental theorem of Algebra to art. *Leonardo*, 38(3), 233-238.
31. Kotarski, W., Gdawiec, K., & Lisowska, A. (2012, July). Polynomiography via Ishikawa and Mann iterations. In *International Symposium on Visual Computing* (pp. 305-313). Springer, Berlin, Heidelberg.
32. Kalantari, B. (2009). Polynomial root-finding and polynomiography. World Scientific, Singapore.

Sabir Yasin

Department of Mathematics, University of Lahore, Pakpattan Campus, Lahore Pakistan.
e-mail: sabiryasin77@gmail.com

Amir Naseem

Department of Mathematics, University of Management and Technology, Lahore Pakistan.
e-mail: amir14514573@yahoo.com