A THREE-STEP NINTH ORDER ITERATIVE METHOD FOR SOLVING NON-LINEAR EQUATIONS

MANI SANDEEP KUMAR MYLAPALLI 1, RAJESH KUMAR PALLI, AND RAMADEVI SRI

ABSTRACT. The scope of this paper is to establish a new ninth order iterative method to find the root of non-linear equations. In this paper we came up with a new modification of Newton’s method with higher-order convergence and here itself we proved that the order of convergence is ninth. Finally, we tested with several problems to show the efficiency of the method over the existing ones.

1. INTRODUCTION

In present days much development happening on solving non-linear scalar equations of the form

\[ g(t) = 0. \]

Newton’s method (NR) [2] is one of the optimal second order method to obtain the root of non-linear scalar equation and is given by

\[ t_{n+1} = t_n - \frac{g(t_n)}{g'(t_n)} \quad n = 0, 1, 2, \ldots \]

and the NR method converges quadratically and its efficiency index is \( \sqrt{2} = 1.414 \).

1 corresponding author

2010 Mathematics Subject Classification. 65F08.
Key words and phrases. Iterative method, non-linear scalar equation, functional evolutions, convergence analysis, efficiency index.
An iterative method with ninth order convergence (ZONG) for solving non-linear equations proposed by Zhongyong et.al, [7] is given by

\[ y_n = t_n - \frac{g(t_n)}{g'(t_n)} \]

\[ z_n = y_n - \left\{ 1 + \left( \frac{g(y_n)}{g(t_n)} \right)^2 \right\} \frac{g(y_n)}{g'(y_n)} \]

\[ t_{n+1} = z_n - \left\{ 1 + 2 \left( \frac{g(y_n)}{g(t_n)} \right)^2 + 2 \frac{g(z_n)}{g(y_n)} \right\} \frac{g(z_n)}{g'(y_n)} . \]

A quadrature based three-step ninth-order iterative method (SK) proposed by Khattri [5] is given by

\[ y_n = t_n - \frac{g(t_n)}{g'(t_n)} \]

\[ z_n = y_n - \left( t_n - y_n \right) \frac{g(y_n)}{g(t_n) - 2g(y_n)} \]

\[ x_{n+1} = z_n - \frac{g(z_n) g'(z_n)}{(g'(z_n))^2 - g(z_n) \left[ \frac{g(z_n) - g(t_n) - g'(t_n)(z_n - t_n)}{(z_n - t_n)^2} \right]} . \]

New ninth order J-Halley method for solving non-linear equations (FA) proposed by Farooq et.al, [1] is given by

\[ y_n = t_n - \frac{2h(t_n)}{3h'(t_n)} \]

\[ z_n = t_n - J_f \frac{h(t_n)}{h'(t_n)} \]

\[ x_{n+1} = z_n - \frac{2h(z_n) h'(z_n)}{2h'(z_n) - h(z_n) L} . \]

where \( J_f = \frac{3h'(y_n) + h'(t_n)}{6h'(y_n) - 2h'(t_n)} \) and \( L = \frac{h'(z_n) - h'(t_n)}{z_n - t_n} . \)

In section 2, we defined the new three-step iterative method and in section 3, we concluded our method is converging with order nine. Finally, in section 4, we compared our new scheme with other methods of the same order of convergence discussed in section 1.
2. Ninth order convergent (SM) method

Consider $t^*$ is an exact root of (1.1) where $g(t)$ is continuous and has well defined first derivatives. Let $t_n$ be the root of nth approximation of (1.1) and is

$$t^* = t_n + \varepsilon_n,$$

where $\varepsilon_n$ is the error. Thus, we get

$$g(t^*) = 0.\tag{2.2}$$

Writing $g(t^*)$ by Taylor’s series about $t_n$, we have

$$g(t^*) = g(t_n) + \varepsilon_n g'(t_n) + \frac{\varepsilon_n^2}{2!} g''(t_n) + \cdots \tag{2.3}$$

by neglecting higher power $\varepsilon_n$, i.e. neglect terms from $\varepsilon_n^3$ onwards. Using (2.2) and (2.3), we have

$$\varepsilon_n^2 g''(t_n) + 2\varepsilon_n g'(t_n) + 2g(t_n) = 0 \tag{2.4}$$

On Substituting $t^*$ by $t_{n+1}$ in (2.1) and from (2.4), we get

$$t_{n+1} = t_n - \frac{2g(t_n)}{g'(t_n)} \left( \frac{1}{1 + \sqrt{1 - 2\mu_n}} \right), \tag{2.5}$$

where, $\mu_n = \frac{g(t_n)g''(t_n)}{[g'(t_n)]^2}$ and

$$g''(t_n) = \frac{2}{t_{n-1} - t_n} \left[ \frac{3g(t_{n-1}) - g(t_n)}{t_{n-1} - t_n} - 2g'(t_n) - g'(t_{n-1}) \right].$$

Here we developed a new algorithm by taking the first two steps from [3] and (2.5) as the third step.

2.1. Algorithm. The iterative scheme is computed by $x_{n+1}$ as $z_n = t_n - \frac{g(t_n)}{g'(t_n)}$

$$y_n = z_n + \left( g'(z_n) - g'(t_n) \right) \frac{g(t_n)}{2(g'(t_n))^2}$$

$$x_{n+1} = y_n - \frac{2g(y_n)}{g'(y_n)} \left( \frac{1}{1 + \sqrt{1 - 2\mu_n}} \right).$$
where \( \mu_n = \frac{g'(y_n) g''(y_n)}{[g'(y_n)]^2} \) and

\[
g''(y_n) = \frac{2}{z_n - y_n} \left[ 3g(z_n) - g(y_n) \right] - 2g'(y_n) - g'(z_n) .
\]

The method (2.6) is known as the ninth order convergent method (SM), it requires two functional evaluations and three first-order derivatives.

### 3. Convergence Criteria

**Theorem 3.1.** Let \( t_0 \in I \) be a single zero of a sufficiently differentiable function \( g \) for an open interval \( I \). If \( t_0 \) is in the neighborhood of \( t^* \). Then the algorithm (2.6) has tenth order convergence.

**Proof.** Let the single zero of (1.1) be \( t^* \) and \( t_n = t_n + \varepsilon_n \). Thus, \( g(t^*) = 0 \). By Taylor’s series, writing \( g(t^*) \) about \( t_n \), we obtain:

\[
g(t_n) = g'(t^*) (\varepsilon_n + c_2 \varepsilon_n^2 + c_3 \varepsilon_n^3 + c_4 \varepsilon_n^4 + \cdots )
\]

Dividing (3.1) by (3.2), we get:

\[
\frac{g(t_n)}{g'(t^*)} = (\varepsilon_n - c_2 \varepsilon_n^2 - (2c_3 - 2c_2^2) \varepsilon_n^3 - (3c_4 - 7c_2 c_3 + 4c_2^2) \varepsilon_n^4 + \cdots ) .
\]

From \( z_n = t_n - \frac{g(t_n)}{g'(t^*)} \), we get \( z_n = t^* + \omega_n \), where \( \omega_n = c_2 \varepsilon_n^2 + (2c_3 - 2c_2^2) \varepsilon_n^3 + (3c_4 - 7c_2 c_3 + 4c_2^2) \varepsilon_n^4 + \cdots . \) Now

\[
g(z_n) = g'(t^*) (c_2 \varepsilon_n^2 + (2c_3 - 2c_2^2) \varepsilon_n^3 + (3c_4 - 7c_2 c_3 + 5c_2^3) \varepsilon_n^4 + \cdots )
\]

\[
g'(z_n) = g'(t^*) (1 + 2c_2^2 \varepsilon_n^2 + 2c_4 (2c_3 - 2c_2^2) \varepsilon_n^3 + (6c_2 c_4 - 11c_2 c_3 + 8c_2^2) \varepsilon_n^4 + \cdots )
\]

\[
\left(g'(z_n) - g'(t_n)\right) \frac{g(t_n)}{2(g'(t_n))^2} = \left(-c_2^2 \varepsilon_n^2 + \left(4c_2^2 - \frac{3}{2} c_3 \right) \varepsilon_n^3 + \left(-13c_2^3 + \frac{23}{2} c_2 c_3 - 2c_4 \right) \varepsilon_n^4 + \cdots \right).
\]

From the second step in the scheme (2.6), we get \( y_n = t^* + Y \), where

\[
Y = \left(2c_2^2 + \frac{1}{2} c_3 \right) \varepsilon_n^3 + \left(-9c_2^3 + \frac{9}{2} c_2 c_3 + c_4 \right) \varepsilon_n^4 + \cdots .
\]
(3.3) \[ g(y_n) = g'(t^*) \left( Y + c_2 Y^2 + c_3 Y^3 + c_4 Y^4 + \cdots \right) \]
(3.4) \[ g'(y_n) = g'(t^*) \left( 1 + 2c_2 Y + 3c_3 Y^2 + 4c_4 Y^3 + \cdots \right). \]

Now, we obtain

\[ g''(y_n) = g'(t^*) \left( 2c_2 + 3c_2 c_3 \left( 2c_2^2 + \frac{1}{2} c_3 \right) \varepsilon_n^3 + \cdots \right). \]

From \( \mu_n = \frac{g(y_n) g''(y_n)}{[g'(y_n)]^2} \), we get

(3.5) \[ \mu_n = P_1 \varepsilon_n^3 + P_2 \varepsilon_n^4 + \cdots, \]

where, \( P_1 = 2c_2 \left( 2c_2^2 + \frac{1}{2} c_3 \right) \), \( P_2 = 2c_2 \left( -9c_2^3 + \frac{9}{2} c_2 c_3 + c_4 \right) \cdots \).

Using (3.5), we get

(3.6) \[ \left( 1 + \sqrt{1 - 2\mu_n} \right)^{-1} = \frac{1}{2} \left( 1 + \frac{P_1}{2} \varepsilon_n^3 + \frac{P_2}{2} \varepsilon_n^4 + \cdots \right). \]

On dividing (3.3) and (3.4),

(3.7) \[ \frac{g(y_n)}{g'(y_n)} = (Y - c_2 Y^2 - (2c_3 - 2c_2^2) Y^3 - (3c_4 - 7c_2 c_3 + 4c_2^3) Y^4 + \cdots) \]

From (3.6) and (3.7), we get

\[ \frac{2g(y_n)}{g'(y_n)} \left( 1 + \frac{1}{\sqrt{1 - 2\mu_n}} \right) = y_n + \left( c_2^2 - 2c_3 \right) \left( 2c_2^2 + \frac{1}{2} c_3 \right) \varepsilon_n^9 + o(\varepsilon_n^{10}), \]

and from the third step of (2.6), i.e. \( x_{n+1} = y_n - \frac{2g(y_n)}{g'(y_n)} \left( \frac{1}{1 + \sqrt{1 - 4\mu_n}} \right) \), we get

\[ \varepsilon_{n+1} = \left( c_2^2 - 2c_3 \right) \left( 2c_2^2 + \frac{1}{2} c_3 \right) \varepsilon_n^9 + o(\varepsilon_n^{10}). \]

Thus, it’s proved that this new scheme is ninth order convergence and its efficiency index is \( \sqrt[9]{5} = 1.5518 \). \( \square \)
4. Numerical Examples

We consider the some examples considered by Vatti [6] and MMS [4] and compared our method with NR, SK, ZONG, FA methods. The computations are carried out by using mpmath-PYTHON and the number of iterations for these methods are obtained for comparisons such that \( |x_{n+1} - x_n| < 10^{-59} \) and \( |g(x_{n+1})| < 10^{-201} \). The test functions and simple zeros are given below

\[
egin{align*}
    h_1(x) &= \sin(2 \cos x) - 1 - x^2 + e^{\sin(x^3)}, \quad t^* = -0.7848959876612125 \\
    h_2(x) &= \sin x + \cos x + x, \quad t^* = -0.4566247045676308 \\
    h_3(x) &= (x + 2)e^x - 1, \quad t^* = -0.442854010023885 \\
    h_4(x) &= x^2 + \sin \left( \frac{x}{5} \right) - \frac{1}{4}, \quad t^* = -0.060960589605896 \\
    h_5(x) &= \cos x - x, \quad t^* = 0.7390851332151606 \\
    h_6(x) &= x^3 - 10, \quad t^* = 2.1544346900318837 \\
    h_7(x) &= e^{-x} + \cos x, \quad t^* = 1.7461395304080124 \\
    h_8(x) &= e^{\sin x} - x + 1, \quad t^* = 2.6306641479279036 \\
    h_9(x) &= \sin^2 x - x^2 + 1, \quad t^* = 1.404491648215341
\end{align*}
\]

Where \( P \) is the order of convergence, \( N \) is the number of functional values per iteration and \( EI \) is the Efficiency Index.

<table>
<thead>
<tr>
<th>Methods</th>
<th>( P )</th>
<th>( N )</th>
<th>( EI )</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR</td>
<td>2</td>
<td>2</td>
<td>1.414</td>
</tr>
<tr>
<td>FA</td>
<td>9</td>
<td>6</td>
<td>1.442</td>
</tr>
<tr>
<td>SK</td>
<td>9</td>
<td>5</td>
<td>1.551</td>
</tr>
<tr>
<td>ZONG</td>
<td>9</td>
<td>5</td>
<td>1.551</td>
</tr>
<tr>
<td>SM</td>
<td>9</td>
<td>5</td>
<td>1.551</td>
</tr>
</tbody>
</table>

Where \( x_0 \) is the initial approximation, \( n \) is the number of iterations, \( er \) is the error and \( fv \) is the functional value.
Here in this scheme, we introduced a new ninth order convergent iterative method with efficiency index 1.5518. It requires two functional evaluations and three first derivatives. Table 1 compares the efficiency of different methods and
the computational results in Table 2 show good results when compared with the other methods.

REFERENCES


DEPARTMENT OF MATHEMATICS GITAM (DEEMED TO BE UNIVERSITY) VISAKHAPATNAM, INDIA
E-mail address: mmylapal@gitam.edu

DEPARTMENT OF MATHEMATICS GITAM (DEEMED TO BE UNIVERSITY) VISAKHAPATNAM, INDIA
E-mail address: rajeshkumar.viit@gmail.com

DEPARTMENT OF MATHEMATICS DR. L. BULLAYYA COLLEGE VISAKHAPATNAM, INDIA
E-mail address: ramadevisri9090@gmail.com