Quadrature based three-step iterative method for non-linear equations

Nazir Ahmad Mir, Naila Rafiq, Nusrat Yasmin

Abstract

In this paper, we present three-step quadrature based iterative method for solving non-linear equations. The convergence analysis of the method is discussed. It is established that the new method has convergence order eight. Numerical tests show that the new method is comparable with the well known existing methods and in many cases gives better results. Our results can be considered as an improvement and refinement of the previously known results in the literature.

2010 Mathematics Subject Classification: 65H05, 34A34.
Key words and phrases: Iterative methods, three-step methods, Quadrature rule, Predictor-corrector methods, Nonlinear equations.

1 Introduction

Let us consider a single variable non-linear equation

\[ f(x) = 0. \]

Finding zeros of a single variable nonlinear equation (1) efficiently, is an interesting and very old problem in numerical analysis and has many applications in applied sciences.

1 Received 26 August, 2008
Accepted for publication (in revised form) 23 May, 2010
In recent years, researchers have developed many iterative methods for solving equation (1). These methods can be classified as one-step, two-step and three-step methods, see[1 – 14]. These methods have been proposed using Taylor series, decomposition techniques, error analysis and quadrature rules, etc. Abbasbandy[2], Chun[4] and Grau[8] have proposed many two-step and three-step methods.

In this paper, we present three-step quadrature based iterative method for solving non-linear equations. We prove that the new method has order of convergence eight. The method and its algorithm is described in section 2. The convergence analysis of the method is discussed in section 3. Finally, in section 4, the method is tested on numerical examples given in the literature. It was noted that the new method is comparable with the well known existing methods and in many cases gives better results. Our results can be considered as an improvement and refinement of the previously known results in the literature.

2 The Iterative Method

Weerakoon and Fernando [13], Gyurhan Nedzhibov [12] and M. Frontini and E. Sormani [6 – 7] have proposed various methods by the approximation of the indefinite integral

\[ f(x) = f(x_n) + \int_{x_n}^{x} f'(t) dt, \]

using Newton Cotes formulae of order zero and one. We approximate, here however the integral (2) by rectangular rule at a generic point \( x + z_n^2 \) with the end-points \( x \) and \( z_n \). We thus have:

\[ \int_{z_n}^{x} f'(t) dt = (x - z_n)f'\left(\frac{x + z_n}{2}\right), \]

this gives

\[ -f(z_n) = (x - z_n)f'\left(\frac{x + z_n}{2}\right). \]
From (3), we have:

\[ x - z_n = -\frac{f(z_n)}{f'(\frac{x^* + z_n}{2})} \]  

Therefore, we have:

\[ x_{n+1} = z_n - \frac{f(z_n)}{f'(\frac{x^* + z_n}{2})} \]  

For a generic point \( w_n = \frac{x^* + z_n}{2} \), consider the Ostrowski’s method and the Newton’s method:

\[ x^* = y_n - \frac{(x_n - y_n)f(y_n)}{f(x_n) - 2f(y_n)}, \]
\[ z_n = y_n - \frac{f(y_n)}{f'(y_n)}. \]

This formulation allows to suggest many one-step, two-step and three-step methods. We however define the following three-step iterative method:

**Algorithm 2.1** For a given initial guess \( x_0 \), find the approximate solution by the iterative scheme:

\[ y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \]
\[ w_n = y_n - \frac{1}{2} \left[ \frac{(x_n - y_n)f(y_n)}{f(x_n) - 2f(y_n)} + \frac{f(y_n)}{f'(y_n)} \right], \]
\[ x_{n+1} = z_n - \frac{f(z_n)}{f'(w_n)}. \]

where \( z_n \) is defined by (7).

Algorithm 2.1 can further be modified by using an approximation for \( f'(y_n) \) with the help of Taylor’s expansion.

Let \( y_n \) be defined by (8). If we use Taylor expansion of \( f'(y_n) \):

\[ f'(y_n) \simeq f'(x_n) + f''(x_n)(y_n - x_n), \]

(where the higher derivatives are neglected) in combination with Taylor approximation of \( f(y_n) \):

\[ f(y_n) \simeq f(x_n) + f'(x_n)(y_n - x_n) + \frac{1}{2}f''(x_n)(y_n - x_n)^2, \]
we can remove the second derivative and approximate $f'(y_n)$ as:

$$f'(y_n) \approx 2 \left[ \frac{f(y_n) - f(x_n)}{y_n - x_n} \right] - f'(x_n).$$

then Algorithm 2.1 can be written in the form of the following algorithm:

**Algorithm 2.2** For a given initial guess $x_0$, find the approximate solution by the iterative scheme:

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

(13) $w_n = y_n - \frac{1}{2} \left[ \frac{(x_n - y_n)f(y_n)}{f(x_n) - 2f(y_n)} + \frac{f(y_n)}{2 \left[ \frac{f(y_n) - f(x_n)}{y_n - x_n} \right] - f'(x_n)} \right],$

(14) $x_{n+1} = z_n - \frac{f(z_n)}{f'(w_n)},$

where $z_n$ is defined by (7).

We will compare this method with the Ostrowski’s method, Grau’s method and seventh order method defined in [1] by Jisheng Kou et al. The algorithms of these methods are given below:

**Algorithm 2.3** For a given initial guess $x_0$, find the approximate solution by the iterative scheme:

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

(16) $x_{n+1} = y_n - \frac{(x_n - y_n)f(y_n)}{f(x_n) - 2f(y_n)}.$

**Algorithm 2.4** For a given initial guess $x_0$, find the approximate solution by the iterative scheme:

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

(18) $\mu = \frac{x_n - y_n}{f(x_n) - 2f(y_n)},$

(19) $z_n = y_n - \mu f(y_n),$

(20) $x_{n+1} = z_n - \mu f(z_n).$
**Algorithm 2.5** For a given initial guess $x_0$, find the approximate solution by the iterative scheme:

\begin{align}
    y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\
    z_n &= y_n - \frac{(x_n - y_n)f(y_n)}{f(x_n) - 2f(y_n)}, \\
    x_{n+1} &= z_n - \left[\left(1 + \frac{f(y_n)}{f(x_n) - 2f(y_n)}\right)^2 + \frac{f(z_n)}{f(y_n)}\right] \frac{f(z_n)}{f'(x_n)}
\end{align}

3 Convergence Analysis

Let us now discuss the convergence analysis of the algorithm 2.2 discussed above.

**Theorem 1** Let $\alpha \in I$ be a simple zero of sufficiently differentiable function $f : I \subseteq \mathbb{R} \to \mathbb{R}$ for an open interval $I$. If $x_0$ is sufficiently close to $\alpha$, then the algorithm 2.2 has eighth order convergence.

**Proof.** Let $\alpha$ be a simple zero of $f$ and $x_n = \alpha + e_n$. By Taylor’s expansion, we have:

\begin{align}
    f(x_n) &= f'(\alpha)(e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 + c_6 e_n^6 + c_7 e_n^7 + c_8 e_n^8) + O(e_n^9), \\
    f'(x_n) &= f'(\alpha)(1 + 2c_2 e_n + 3c_3 e_n^2 + 4c_4 e_n^3 + 5c_5 e_n^4 + 6c_6 e_n^5 + 7c_7 e_n^6 + 8c_8 e_n^7) + O(e_n^9).
\end{align}

where

\begin{align}
    c_k &= \left(\frac{1}{k!}\right) \frac{f^{(k)}(\alpha)}{f'(\alpha)}, k = 2, 3, \ldots \text{and } e_n = x_n - \alpha.
\end{align}
Using (24) and (25), we have

\[
(27) \quad \frac{f(x_n)}{f'(x_n)} = e_n - c_2 e_n^2 + 2 \left( c_2^2 - c_3 \right) e_n^3 + \left( 7 c_2 c_3 - 3 c_4 - 4 c_5^2 \right) e_n^4 + \left( 6 c_3^2 - 4 c_5 \right) e_n^5 + 8 c_2 e_n^2 + 10 c_2 c_4 - 20 c_3 c_2^2 e_n^5 + \left( -5 c_6 + 13 c_2 c_5 - 33 c_2 c_5^2 - 16 c_5^2 \right) e_n^6 + 52 c_3 c_2^2 + 17 c_4 c_2 - 28 c_4 c_2 e_n^6 + \left( -32 c_5^2 + c_7 - 8 c_3 c_5 + 24 c_2 c_5 \right) e_n^7 - 8 c_2 c_6 - 56 c_3 c_2^2 - 90 c_3 c_3^2 + 52 c_2 c_4 c_3 - 4 c_7^2 + 9 c_3^2 + 112 c_2 c_3 e_n^7 + \left( 33 c_3^2 c_3 - 54 c_3^2 c_2 + 16 c_2^2 c_2 - 9 c_4 c_5 + 96 c_2^2 c_2 - 84 c_3 c_4 c_2 \right) + 32 c_3 c_2 c_5 - 2 c_7 c_2 - 32 c_3 c_2^2 - 8 c_5 c_3^2 - 32 c_3 c_5 + 16 c_4 c_2 + 4 c_6 c_2^2 e_n^8 + O(e_n^9).
\]

Using (27) in (12), we thus have:

\[
(28) \quad y_n = \alpha + c_2 e_n^2 + \left( -2 c_2^2 + 2 c_3 \right) e_n^3 - \left( 7 c_2 c_3 - 4 c_3^2 - 3 c_4 \right) e_n^4 + \left( 4 c_5 - 10 c_2 c_4 + 20 c_3 c_2^2 - 8 c_4^2 - 6 c_5^2 \right) e_n^5 + \left( 28 c_4 c_2^2 + 33 c_2 c_3^2 + 5 c_6 \right) - 52 c_3 c_2^2 - 17 c_4 c_3 - 13 c_2 c_5 + 16 c_5^2 e_n^6 + \left( -c_7 - 52 c_2 c_4 c_3 + 4 c_4^2 \right) - 9 c_3^2 + 56 c_3 c_2 + 8 c_2 c_6 - 24 c_2 c_3 + 90 c_2 c_3^2 + 32 c_6 + 8 c_3 c_5 - 112 c_2 c_3 e_n^7 + \left( 32 c_3 c_2^2 + 54 c_3^2 c_3 + 32 c_4 c_2 - 32 c_3 c_2 c_5 + 2 c_7 c_2 - 4 c_6 c_2^2 + 8 c_5 c_3^2 - 16 c_4 c_2^2 - 16 c_3^2 c_2 + 9 c_4 c_5 - 96 c_2 c_3^2 \right) e_n^8 + O(e_n^9).
\]

By Taylor’s series, we have:

\[
f(y_n) = (y_n - \alpha) f'(\alpha) + \frac{1}{2!} (y_n - \alpha)^2 f''(\alpha) + \ldots.
\]

Using (28) in the above relation and on simplifying, we have:

\[
(29) \quad f(y_n) = f'(\alpha)(c_2 e_n^2 + 2 \left( c_3 - c_2^2 \right) e_n^3 + \left( 7 c_2 c_3 + 3 c_4 + 5 c_5^2 \right) e_n^4 + \left( 24 c_3 c_2^2 \right) - 12 c_2 + 4 c_5 - 10 c_2 c_4 - 6 c_3^2 e_n^4 + \left( 37 c_2 c_3^2 - 73 c_2 c_5 + 28 c_5^2 + 34 c_4 c_2^2 \right) + 5 c_6 - 17 c_4 c_3 - 13 c_2 c_5 e_n^5 + \left( -40 c_2 c_4 c_3 + 56 c_5 c_3^2 - 34 c_2^2 c_3 + 24 c_3 c_4 \right) + 16 c_2 c_5 - c_7 + 4 c_4^2 - 9 c_3^2 + 8 c_2 c_6 + 8 c_3 c_5 e_n^6 + \left( -23 c_3 c_4 c_2 - 16 c_3 c_2 c_5 \right) - 32 c_3 c_4 + 42 c_3^2 c_2 - 7 c_2 c_2 + 9 c_4 c_5 + 78 c_3^2 c_2 + 2 c_7 c_2 - 216 c_3 c_5^2 - 34 c_2 c_2^2 + 9 c_3 c_6 + 105 c_4 c_2^2 + 6 c_4 c_2^2 + 80 c_2^2) e_n^7 + O(e_n^9).
\]
Using (24), (25), (28) and (29) in (11), we have:

\[
(30) \quad f'(y_n) = f'(\alpha)(1+(2c_2^2-c_3)e_n^2+(-4c_2^3-2c_4+6c_2c_3)e_n^4+(-3c_5
+16c_3c_2^2+4c_3^2+8c_2c_4+8c_4^2)e_n^6+(-22c_4c_2^2+10c_2c_5
+40c_3c_2^3-16c_5+10c_4c_3-18c_2c_3
-4c_6)e_n^8+(-5c_7+6c_4^2-48c_2c_4c_3+58c_2^3c_4-28c_2c_5^2
+62c_2^2c_3^2+32c_2^3+12c_2c_6-4c_3^2-96c_2^3c_3+12c_3c_5)e_n^9+(64c_2^5-6c_8
+108c_4c_2^4+32c_3c_2^2+14c_6c_2^2+14c_4c_5+244c_3c_2^3-46c_2c_3^2
+14c_3c_6-104c_3c_4c_2^2-256c_3c_2^5-14c_2^3c_4)c_n^3+(870c_2c_3^3-14c_6c_2^3
-528c_2c_3^2c_4+2c_8c_2+48c_2c_5c_4+8c_2^5
+768c_2^4c_4^2+16c_6c_4-1504c_2^2c_3^2-128c_2^4c_3-1050c_3c_2^3c_4-288c_4c_2^5
-126c_2^3c_4^2-50c_3^2+2c_8c_7+44c_3c_2^3+88c_6c_2c_3-348c_2^3c_5c_3
-2c_7c_2^2+16c_4c_3+86c_2c_3^2)c_n^5+O(c_n^9).
\]

Using (28), (29) and (30) in (7), we have:

\[
(31) \quad z_n = \alpha + (-c_2c_3 + c_3^2)e_n^4 + (-2c_3^2 + 8c_3c_2^2 - 2c_2c_4 - 4c_2^4)e_n^6 + (10c_2^5
+18c_2c_3^2 - 7c_4c_3 + 12c_2c_4^2 - 30c_3c_2^2 - 3c_2c_5)e_n^8 + (-4c_2c_6
+80c_2^4c_3 - 40c_2^3c_4 + 16c_2c_5c_3 + 52c_2c_4c_3 - 10c_3c_5 - 80c_2^2c_3
+12c_3^3 - 20c_6^2 - 6c_4^2)e_n^9 + (252c_2^3c_3^2 + 37c_2c_4^2 + 68c_3c_2c_5
+50c_2c_4 - 17c_4c_5 - 17c_3c_5^2 - 209c_3c_4c_2^2 + 101c_4c_2^3 - 51c_5c_2^3
+20c_6c_2^2 - 5c_7c_2 - 13c_3c_6 - 91c_3^2c_2 + 36c_2^4)c_n^8 + O(c_n^9).
\]

By Taylor's series, we have:

\[
f(z_n) = (z_n - \alpha)f'(\alpha) + \frac{1}{2!}(z_n - \alpha)^2f''(\alpha) + \ldots.
\]

Using (31) in the above relation and on simplifying, we have:

\[
(32) \quad f(z_n) = f'(\alpha)(c_2(-c_3+c_2^2)e_n^4+(8c_2c_3^2-2c_2c_4-4c_2^4-2c_2^2)e_n^6+(-30c_3c_2^3
+18c_2c_3^2+10c_2^5-3c_2c_5+12c_2c_4^2-7c_3c_4)e_n^8+(-4c_2c_6+80c_2c_3
-40c_3c_2^4+16c_2c_4^2+52c_2c_4c_3-10c_3c_5-80c_2^2c_3
-6c_4^2)e_n^9+(253c_2^3c_3^2+37c_2^2c_4+68c_3c_2c_5+50c_2c_4^2-17c_4c_5
-180c_3c_2^5-209c_3c_4c_2^2+101c_4c_2^3-51c_5c_2^3+20c_6c_2^2-5c_7c_2
-13c_3c_6-91c_3^2c_2+37c_2^4)c_n^8+O(c_n^9).
\]
Using (24), (27), (28) and (29) in (13), we have:

\[
(33) \quad w_n = \alpha + (-c_2c_3 + c_2^3) c_n^4 - 2c_2^3 + 8c_3c_2^2 - 2c_2c_4 - 4c_2^4) c_n^5 + (10c_2^5 + 18c_2c_3^2 \\
- 7c_4c_3 + 12c_4c_2^2 - 30c_3c_2^2 - 3c_2c_5) c_n^6 + (-4c_2c_6 + 80c_2c_3 - 40c_3c_4 \\
+ 16c_2^3c_5 + 52c_2c_4c_3 - 10c_3c_5 - 80c_2c_3^2 + 12c_3^3 - 20c_2^2 - 6c_3^2) c_n^7 \\
+ (4c_2^7 + 50c_2^3c_4 - 137c_3c_4c_2^2 + 44c_2^3c_2 - \frac{3}{2} c_7c_2 - 13c_3c_6 + 53c_3c_2c_5 \\
- \frac{155}{2} c_3c_2 - 21c_5c_2^3 + 37c_4c_2^2 - 58c_3c_2^4 + 8c_6c_2^2 + 29c_3c_2^2 - 17c_4c_5) c_n^8 + O(e_n^9).
\]

By Taylor’s series, we have:

\[
(34) \quad f'(w_n) = f'(\alpha)(1 + (-2c_3c_2^2 + 2c_2^4) c_n^4 + (-4c_2c_3^2 - 8c_2^3 + 16c_3c_2^3 - 4c_4c_2^2) c_n^5 \\
+ (-14c_2c_4c_3 + 24c_2^3c_4 + 20c_2^5 + 36c_2^5c_2^2 - 20c_2^2c_3 - 6c_2c_5) c_n^6 \\
+ (32c_3c_2^3 - 160c_2^2c_2^2 - 80c_4c_2^2 + 24c_2c_4 + 104c_3c_4c_2^2 - 40c_2^7 - 20c_3c_2c_5 - 8c_6c_2^2 - 12c_3c_2^2 + 160c_3c_2^2) c_n^7 + (282c_3c_2^4 - 34c_2c_5c_4 \\
+ 100c_2c_3c_5 - 3c_7c_2^2 + 74c_4c_2^2 - 152c_2c_3^2 + 104c_3c_4c_2^2 + 58c_2^2c_2^2 \\
- 113c_3c_3 - 274c_4c_3c_2^3 + 8c_2^8 + 16c_6c_2^2 - 42c_4c_2^2 - 26c_6c_2c_3) c_n^8 + O(e_n^9).
\]

Using (31), (32) and (34) in (14), we have:

\[
(35) \quad x_{n+1} = \alpha + (c_2^7 + c_3^2c_2^3 - 2c_3c_2^2) c_n^8 + O(e_n^9),
\]

or

\[
(36) \quad e_{n+1} = (c_2^7 + c_3^2c_2^3 - 2c_3c_2^2) e_n^8 + O(e_n^9).
\]

Thus, we observe that the algorithm 2.2 has eighth order convergence.

4 Numerical examples

We consider here some numerical examples to demonstrate the performance of the new developed three-step iterative method, namely algorithm 2.2. We compare the methods defined in J.Kou et al. (algorithm 2.3 \((G_4)\)), algorithm 2.4 \((G_6)\), and algorithm 2.5 \((G_7)\) and the new developed three-step method
algorithm 2.2 (MN) in this paper. All the computations are performed using Maple 10.0. We take $\varepsilon = 10^{-15}$ as tolerance.

The following criteria is used for estimating the zero:

(i) $\delta = |x_{n+1} - x_n| < \varepsilon$
(ii) $|f(x(n))| < \varepsilon$

The following examples of J.Kou et al. [1] are used for numerical testing:

\begin{tabular}{ll}
\textbf{Example} & \textbf{Exact Zero} \\
\hline
$f_1 = x^3 + 4x^2 - 15$, & $\alpha = 1.6319808055660636,$ \\
$f_2 = xe^{x^2} - \sin^2(x) + 3 \cos(x) + 5$, & $\alpha = -1.207647827130919,$ \\
$f_3 = \sin(x) - \frac{1}{2}x$, & $\alpha = 1.8954942670339809,$ \\
$f_4 = 10xe^{-x^2} - 1$, & $\alpha = 1.67963061042845,$ \\
$f_5 = \cos(x) - x$, & $\alpha = 0.73908513321516067,$ \\
$f_6 = \sin^2(x) - x^2 + 1$, & $\alpha = 1.4044916482153411,$ \\
$f_7 = e^{-x} + \cos(x)$, & $\alpha = 1.74613953040801241765.$ \\
\end{tabular}

For convergence criteria, it was required that $\delta$, the distance between two consecutive iterates was less than $10^{-15}$, $n$ represents the number of iterations and $f(x_n)$, the absolute value of the function. All the values are computed with 350 significant digits. The numerical comparison is given in Table 4.1.

\begin{tabular}{llll}
$n$ & $f(x_n)$ & \\
\hline
$G_4$ & 3 & 1.03e-228 \\
$G_6$ & 3 & 4.46e-179 \\
$G_7$ & 3 & 1.06e-274 \\
MN & 3 & 1.00e-348 \\
\hline
$G_4$ & 3 & 8.82e-223 \\
$G_6$ & 3 & 2.54e-155 \\
$G_7$ & 3 & 1.20e-264 \\
MN & 3 & 2.79e-259 \\
\hline
$G_4$ & 3 & 5.12e-313 \\
$G_6$ & 3 & 8.44e-252 \\
$G_7$ & 3 & 3.00e-320 \\
MN & 3 & 3.00e-350 \\
\end{tabular}
\[ f_4, x_0 = 1.8 \]
\[
\begin{array}{ccc}
| \text{G}_4 | & 3 & 1.16e-236 \\
| \text{G}_6 | & 3 & 9.37e-187 \\
| \text{G}_7 | & 3 & 1.34e-281 \\
| \text{MN} | & 3 & 0 \\
\end{array}
\]
\[ f_5, x_0 = 1 \]
\[
\begin{array}{ccc}
| \text{G}_4 | & 3 & 7.05e-296 \\
| \text{G}_6 | & 3 & 4.12e-237 \\
| \text{G}_7 | & 3 & 0 \\
| \text{MN} | & 3 & 0 \\
\end{array}
\]
\[ f_6, x_0 = 1.6 \]
\[
\begin{array}{ccc}
| \text{G}_4 | & 3 & 3.26e-226 \\
| \text{G}_6 | & 3 & 7.54e-178 \\
| \text{G}_7 | & 3 & 6.26e-271 \\
| \text{MN} | & 3 & 1.00e-349 \\
\end{array}
\]
\[ f_7, x_0 = 2 \]
\[
\begin{array}{ccc}
| \text{G}_4 | & 3 & 1.05e-279 \\
| \text{G}_6 | & 3 & 1.58e-223 \\
| \text{G}_7 | & 3 & 3.00e-320 \\
| \text{MN} | & 3 & 3.00e-350 \\
\end{array}
\]

Table 4.1.

5 CONCLUSION

From Table 4.1, we observe that our three-step iterative method is comparable with the methods defined in the paper of Jisheng Kou et al. [1] and in many cases gives better results in terms of the function evaluation \( f(x_n) \). Moreover, the computational efficiency of the algorithm \( 2.2 \) i.e. \( 8^{\frac{3}{11}} \simeq 1.515717 \) is better than the efficiency of most of the other methods defined in the literature.

References

Quadrature based three-step iterative...


**Nazir Ahmad Mir**  
Mathematics Department  
Preston University  
44000 Islamabad, Pakistan  
e-mail: nazirahmad.mir@gmail.com

**Naila Rafiq, Nusrat Yasmin**  
Centre for Advanced Studies in Pure & Applied Mathematics,  
Bahauddin Zakariya University, Multan, Pakistan  
e-mail: nrafiqnaila@yahoo.com, nusyasmin@yahoo.com