ON THE EFFICIENCY OF A FAMILY OF QUADRATURE-BASED METHODS FOR SOLVING NONLINEAR EQUATIONS

OGHOVESE OGBEREYIVWE, KINGSLEY OBIAJULU MUKA

1Department of Mathematics and Statistics, Delta State Polytechnic, Ozoro, Nigeria.
2Department of Mathematics, University of Benin, Nigeria.
Corresponding author: vesenelly@yahoo.com

Abstract
This paper modifies the generalized family of two-step quadrature based iterative methods for approximating the solution of nonlinear equations developed in Noor et al., [9]. The modification improved the order of convergence and the efficiency of the iterative methods at the cost of no additional functional evaluation per iteration. The convergence of the proposed method was established. Some numerical examples are given to illustrate the efficiency and performance of the new family of methods proposed herein.

Keywords: Nonlinear equations, quadrature based methods, convergence order, efficiency index.

1. Introduction
Over the years, plethora of iterative methods for approximating the solution of nonlinear equation have been developed in literature. Some of these methods are quadrature-based, see [1-14] and some reference therein. Unfortunately, some of these quadrature-based methods are expensive when it comes to implementation because they involve the evaluation of many number of functions and functions derives at every iteration point. This disadvantage result to limitation in the applicability of some of these quadrature-based methods for solving nonlinear equation.
One of the most important goals in developing new methods that are iterative based, is to develop methods that can attain optimal convergence requiring few number of functions and derivatives evaluations in an iteration cycle. Consequently, in this paper, a technique utilized in accelerating the convergence and improving the efficiency of a family of quadrature based iterative methods for solving the zero of nonlinear equation, developed in Noor et al. [9] is proposed. The technique involves the introduction of weights to the second step of the family of methods developed in Noor et al. [9], thereby increasing its convergence order from three to four and efficiency at no cost of additional function evaluation.

2. The Iterative Methods

Consider the generic family of quadrature-based methods for solving the zero \( \alpha \) of the nonlinear equations \( f(x) = 0 \) developed as Algorithm 2.2 in Noor et al., [9],

\[
y_k = x_k - \frac{f(x_k)}{f'(x_k)}
\]

\[
Z_k = y_k - \frac{f(y_k)}{\sum_{i=1}^{m} \psi_i f'(x_k + \delta_i(y_k - x_k))}
\]  

(2.1)

where

\[
\sum_{i=1}^{m} \psi_i f'(x_k + \delta_i(y_k - x_k))
\]

is a generic quadrature function satisfying the following constituency conditions

\[
\sum_{i=1}^{m} \psi_i = 1
\]

(2.2)

and

\[
\sum_{i=1}^{m} \psi_i \delta_i = \frac{1}{2}
\]

(2.3)

The convergence order of the methods in equation (2.1) is proven to be \( \rho = 3 \) in Noor et al. [9]. For \( \psi_i \) satisfying equations (2.2) and (2.3) and \( \delta_i \neq 0 \) for some \( i = 1,2,3,... \), the highest efficiency index iterative method that the family of the iterative methods in equation (2.1) can produce is 1.348. However, with slight modification to its second step, the order of convergence can be increased and consequently leads to an increase in the computational efficiency index of all the iterative methods in (2.1). To achieve this, consider the following formulation:

\[
y_k = x_k - \frac{f(x_k)}{f'(x_k)},
\]

\[
Z_k = y_k - \frac{f(y_k)}{\beta \sum_{i=1}^{q} \psi_i f'(x_k + \delta_i(y_k - x_k)) + \Omega f'(x_k)}, k = 1,2,...
\]

(2.4)
The convergence of sequence of approximations \( \{x_k\}_{k \geq 0}, (x_k \in \mathbb{D}) \) obtained by using the method in equation (2.4) is established in Theorem 2.1. We state the following definitions that will be useful in the proof of the theorem on the convergence of the family of iterative method in equation (2.4).

**Definition 2.1.** [10] Let \( e_k = |x_k - \alpha| \) be the error in the \( k \)th iteration of an iterative method, then the equation

\[
e_{k+1} = \eta e_k^p + o(e_k^{p+1}) (2.5)
\]

is called the error equation and \( \rho \) is the order of convergence of the method.

**Definition 2.2** Let \( T \) be total number of new functional evaluation required by an iterative method. The efficiency of an iterative method is measured by efficiency index [11] and is defined by

\[
e = \rho^T, \tag{2.6}
\]

where \( \rho \) is the order of the method.

**Lemma 2.1** [6] Let \( f: x \in \mathbb{D} \subseteq \mathbb{R} \to \mathbb{R} \) be \( r \)-time differentiable in the domain \( \mathbb{D} \subseteq \mathbb{R} \), then for any \( x \in \mathbb{R} \)

\[
f(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2!} f''(x_k)(x - x_k)^2 + \frac{1}{3!} f'''(x_k)(x - x_k)^3 + \ldots + \frac{1}{(r-1)!} f^{(r-1)}(x_k)(x - x_k)^{r-1} + \int_0^1 (1-t)^{r-1} f^{(r)}(x_k + t(x-x_k))(x-x_k)^{r-1} dt \tag{2.7}
\]

Equation (2.7) is the Taylor’s series expansion of \( f(x) \).

**Theorem 2.1.** Let the function \( f: \mathbb{D} \subseteq \mathbb{R} \to \mathbb{R} \) be differentiable and \( f'(x) \neq 0 \) in some neighborhood \( \mathbb{D} \subseteq \mathbb{R} \) containing its zero \( \alpha \). If \( x_0 \) is an initial guess in the neighborhood of \( \alpha \), then the sequence of approximations \( \{x_k\}_{k \geq 0}, (x_k \in \mathbb{D}) \) generated by all methods that are developed from equation (2.4) converges to \( \alpha \) with order \( \rho = 4 \).

**Proof.** Using the Taylor’s series in equation (2.7) to expanding \( f(X) \) and \( f'(X) \) about \( \alpha \), the following equations are obtained respectively.

\[
f(x) = f(\alpha) + f'(\alpha)(x - \alpha) + \frac{1}{2!} f''(\alpha)(x - \alpha)^2 + \frac{1}{3!} f'''(\alpha)(x - \alpha)^3 + \ldots \tag{2.8}
\]

\[
f'(x) = f'(\alpha) + f''(\alpha)(x - \alpha) + \frac{1}{2!} f'''(\alpha)(x - \alpha)^2 + \frac{1}{3!} f^{(iv)}(\alpha)(x - \alpha)^3 + \ldots \tag{2.9}
\]

Set \( x = x_k \), yield

\[
f(x_k) = f'(\alpha) \left[ e_k + \sum_{n=2}^{5} c_n e_k^n + O(e_k^6) \right] \tag{2.10}
\]

and
\[
\frac{1}{f'(x_k)} = f'(x_k)^{-1} [1 - 2c_2e_k + (4c_2^2 - 3c_3^2)e_k^2 - 4(2c_2^3 - 3c_2c_3 + c_4^3) e_k^3 - 16c_2^4 - 36c_2^2c_3 + 9c_3^2 + 16c_2c_4 - 5c_5) e_k^4 + O(e_k^6)]
\] (2.11)

using equations (2.10) and (2.11),

\[
\frac{f(x_k)}{f'(x_k)} = e_k - c_2e_k^2 + 2(c_2^2 - c_3)e_k^3 + (-4c_2^3 + 7c_2c_3 - 3c_4)e_k^4 + O(e_k^5)
\] (2.12)

Inserting equation (2.12) in the first step of equation (2.4) yield

\[
y_k = \alpha + c_2e_k^2 + 2(-c_2^2 + c_3)e_k^3 + \delta(4c_2^3 - 7c_2c_3 + 3c_4)e_k^4 + O(e_k^5)
\] (2.13)

Setting \( x_k = y_k \) and \( x_k = x_k + \delta(y_k - x_k) \) in equations (2.8) and (2.9) respectively yields,

\[
f(y_k) = c_2e_k^2 + (-2c_2^2 + 2c_3)e_k^3 + (4c_2^3 - 7c_2c_3 + 3c_4) e_k^4 + O(e_k^4)
\] (2.14)

and

\[
f'(x_k + \delta(y_k - x_k)) = f'(x_k)[1 + 2c_2(1 - \delta)e_k + (3c_2^2(1 - \delta)^2 + 2c_2^3\delta)e_k^2
\]
+(4c_2^4(1 - \delta)^3 + 2c_2(-2c_2^2 + 2c_3)\delta + 6c_2c_3^2(1 - \delta)^2\delta^2 + O(e_k^3)]
\] (2.15)

From (2.9) and (2.15)

\[
\frac{1}{\beta \sum_{k=1}^q \psi_i f'(x_k + \delta_i(y_k - x_k))} + \Omega f'(x_k)
= f'(x_k)^{-1} \left[ \frac{1}{\beta + \Omega} - \frac{2c_2}{(\beta + \Omega)^2} \left( \Omega + \sum_{i=1}^q \psi_i (\beta - \beta \delta_i) \right) e_k
\]
+
\[
\frac{4c_2^2}{(\beta + \Omega)^3} \left( \Omega + \sum_{i=1}^q \psi_i (\beta - \beta \delta_i)^2 - (\Omega
\]
+
\[
+ \beta \left( 3\Omega c_3 + \beta \sum_{i=1}^q \psi_i (3c_3(-1 + \delta_i)^2 + 2c_2^2\delta_i) \right) \right) e_k^2 + O(e_k^3) \right]
\] (2.16)
Substituting equations (2.14) and (2.16) into the second step of equation (2.4), yields

\[
x_{k+1} = y_k - \frac{f(y_k)}{\beta \sum_{k=1}^q \psi_l f'(x_k + \delta_l (y_k - x_k)) + \Omega f'(x_k)}
\]

\[
= \alpha - \left( 1 - \frac{1}{(\beta + \Omega)} \right) c_2 e_k^2
\]
For equation (2.4) to converge to $\alpha$ with at least convergence order $\rho = 4$, the coefficients of $e_k^4$ and $e_k^3$ in equation (2.17) must vanish. Set $\beta + \Omega = 1$ in equation (2.17), leads to the vanishing of the coefficient of $e_k^2$, while the coefficient of $e_k^3$ expressed in terms of $\beta$ becomes:

$$e_k^3 = \left( -2c_2^2 + c_3 - \frac{2}{(\beta + \Omega)^2} \left( \Omega(-2c_2^2 + c_3) + \beta \sum_{i=1}^{q} \psi_i(c_3 + c_2^2(\delta_i - 2)) \right) \right) e_k^3$$

$$+ \left( 4c_2^3 - 7c_2c_3 + 3c_4 + \frac{1}{\beta + \Omega}(-5c_2^3 + 7c_2c_3 - 3c_4) \right) e_k^3$$

$$- \frac{1}{(\beta + \Omega)^2} \left( 4c_2(c_2^2 - c_3) \sum_{i=1}^{q} \psi_i(\Omega - \Omega \delta_i) - \frac{1}{(\beta + \Omega)^3} \left( c_2 \left( 4c_2^2 \left( \Omega + \sum_{i=1}^{q} \psi_i(\Omega - \delta_i) \right) \right)^2 \right. \right) e_k^4 + O(e_k^5) \tag{2.17}$$

For equation (2.4) to converge to $\alpha$ with at least convergence order $\rho = 4$, the coefficients of $e_k^4$ and $e_k^3$ in equation (2.17) must vanish. Setting $\Omega = 0$ and applying the consistency condition in equation (2.2) and (2.3), leads to obtaining $\beta = 2$. This implies that $\Omega = -1$. Hence, the general error equation of the family of the iterative methods in equation (2.4) becomes

$$e_{k+1} = \alpha + \left( c_2^3 + 4c_2(c_2^2 - c_3) \left( 1 + 2 \sum_{i=1}^{q} \psi_i(-1 + \delta_i) \right) + c_2 \left( 4c_2^2 \left( 2 \sum_{i=1}^{q} \psi_i(-1 + \delta_i) \right) \right)^2 \right)$$

$$\times \left( -3c_3 + 2 \sum_{i=1}^{q} \psi_i(3c_3(-1 + \delta_i)^2 + 2c_2^2\delta_i) \right) e_k^4 + O(e_k^5) \tag{2.18}$$

This completes the proof of Theorem 2.1.

Substituting the obtained values of the parameters $\beta$ and $\Omega$ into equation (2.4), yield the following generalized Two-step quadrature based family of convergence order $\rho = 4$ iterative method for approximating the solution of $\alpha$ of $f(x) = 0$ given as:

$$y_k = x_k - \frac{f(x_k)}{f'(x_k)}$$

$$Z_k = y_k - \frac{f(y_k)}{2 \sum_{k=1}^{q} \psi_i f'(x_k + \delta_i(y_k - x_k)) - f'(x_k)}, k = 1,2,3, \ldots \tag{2.19}$$

for $\delta_i \in (0,1)$, $\psi_i \in \mathbb{R}\{0\}$ and satisfying equations (2.2) and (2.3).
2.1 Some concrete form of the proposed method

In this section, some concrete forms of the iterative method developed from equation (2.19) are proposed by assigning arbitrary values to the parameters \( q, \delta_i \) and \( \psi_1 \) satisfying equations (2.2) and (2.3).

For \( q = 1, \delta_1 = \frac{1}{2}, \psi_1 = 1 \), equation (2.19) reduces to the following iterative method.

**Method 1:** Given an initial guess \( x_0 \), approximate the solution \( \alpha \) of \( f(x) = 0 \) by the iterative scheme:

\[
\begin{align*}
    y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\
    x_{k+1} &= y_k - \frac{f(y_k)}{2f'(\frac{x_k + y_k}{2})} - f'(x_k)
\end{align*}
\]

with error equation satisfying

\[
e_{k+1} = \alpha + \left(c_2^3 - \frac{3}{2}c_2c_3\right)e_k^4 + O(e_k^5)
\]

and efficiency index \( EFF = 1.4142 \). Method 1 is a modified form of Algorithm 2.6 in [9] that has \( EFF = 1.3161 \).

Upon setting \( q = 2, \delta_1 = 0, \delta_2 = \frac{2}{3}, \psi_1 = \frac{1}{4}, \psi_2 = \frac{3}{4} \), in equation (2.19), the following iterative method is proposed.

**Method 2:** Given an initial guess \( x_0 \), approximate the solution \( \alpha \) of \( f(x) = 0 \) by the iterative scheme:

\[
\begin{align*}
    y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\
    x_{k+1} &= y_k - \frac{3}{2f'(\frac{x_k + 2y_k}{3})} - \frac{1}{2}f'(x_k), k = 1,2,3,...
\end{align*}
\]

with error equation obtained as

\[
e_{k+1} = \alpha + (c_2^3 - c_2c_3)e_k^4 + O(e_k^5)
\]

and efficiency index is \( EFF = 1.4142 \). Method 2 is a modified form of Algorithm 2.7 in [9] that has \( EFF = 1.3161 \).
For $q = 3$, $\delta_1 = 0$, $\delta_2 = \frac{1}{2}$, $\delta_2 = 1$, $\psi_1 = \frac{1}{6}$, $\psi_2 = \frac{2}{3}$, and $\psi_3 = \frac{1}{6}$, in equation (2.18), the following iterative method is proposed.

**Method 3:** Given an initial guess $x_0$, approximate the solution $\alpha$ of $f(x) = 0$ by the iterative scheme:

$$
\begin{align*}
    y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\
    Z_k &= y_k - \frac{4}{3}f'\left(\frac{x_k + y_k}{2}\right) + \frac{1}{3}f'(y_k) - \frac{2}{3}f'(x_k)
\end{align*}
$$

with error equation satisfying

$$
e_{k+1} = \alpha + (c_2^3 - c_2c_3)e_k^4 + O(e_k^5)
$$

and efficiency index is $EFF = 1.3195$. Method 3 is a modified form of Algorithm 2.8 in [9] that has $EFF = 1.2457$.

### 3. Numerical Results

In this section, the proposed iterative methods (Method 1 (M1), Method 2 (M2) and Method 3 (M3)), are implemented on two problems (Problem 1-2) in literature in order to illustrate their efficiency. We compare the performance of the proposed methods with the methods developed in [9] (Algorithm 2.6 (Alg. 2.6), Algorithm 2.7 (Alg. 2.7) and Algorithm 2.8 (Alg. 2.8)). Note that M1 is a modification of Alg. 2.6, M2 is a modification of Alg. 2.7 and M3 is a modification of Alg. 2.8. Numerical computations are performed by PYTHON 2.7.12 program with 25 digits floating arithmetic. Intel Celeron(R) CPU 1.6 GHz with 2 GB of RAM processor was used to execute all programs. The stopping criterion used for all programs is $|f(x_i)| < 10^{-15}$. For comparison, the following measurements were used:

i. number of iteration (IT)
ii. total number of functional evaluations (NFE),
iii. function of the last iteration ($|f(x_{k+1})|$),
iv. Computer execution time in seconds (CPU).

The following problems used for implementation of the methods are taken from [2].

**Problem 1**

$$f_1(x) = x^3 - 10, \quad x_0 = -0.5$$

**Problem 2**

$$f_2(x) = e^{x^2 + 7x - 30} - 1, \quad x_0 = 3.5$$
PYTHON program for Method 2 on Problem 1
from mpmath import*
mp.dps=25
import numpy.linalg
import time
start_time=time.clock()
x=-0.5
def f(x):
    return mpf(x**3-10)
def f_(x):
    return mpf(3*x**2)
while norm(f(x))!=0:
    y=mpf(x)-fdiv(f(x),f_(x))
x=mpf(y)-fdiv(f(y),fdiv(3,2)*f_((x+2*y)/3)-fdiv(1,2)*f_(x))
    if norm(f(x))<=10**-15:
        break
print((x))
print time.clock()-start_time, "seconds"

The program above can be altered to suit other methods in solving
Problem 1 and 2. The computational results obtained by using each
method are presented in Tables 3.1 and 3.2.

| Method | IT | NFE | $|f(x_{k+1})|$ | CPU-time |
|--------|----|-----|----------------|----------|
| Alg. 2.6 | 13 | 52  | 2.9347e-07     | 0.045    |
| M1     | 3  | 12  | 1.2518e-05     | 0.012    |
| Alg. 2.7 |    |     | Fail to converge |          |
| M2     | 4  | 16  | 1.1988e-08     | 0.016    |
| Alg. 2.8 |    |     | Fail to converge |          |
| M3     | 4  | 20  | 1.1988e-08     | 0.017    |

Table 3.2 Computational results for Problem 2
| Method | IT | NFE | $|f(x_{k+1})|$ | CPU-time |
|--------|----|-----|----------------|----------|
| Alg. 2.6 | 7  | 28  | 7.3853e-14     | 0.028    |
| M1     | 4  | 16  | 2.3416e-09     | 0.018    |
| Alg. 2.7 | 7  | 28  | 8.2688e-13     | 0.030    |
| M2     | 5  | 20  | 6.2125e-15     | 0.023    |
| Alg. 2.8 | 7  | 35  | 6.7389e-13     | 0.032    |
| M3     | 5  | 25  | 1.0348e-15     | 0.025    |
From Tables 3.1-3.2, the computational results indicates that the proposed iterative methods are more efficient than the methods compared. This is further supported by the fact that the CPU- times for the proposed methods are less compared with their corresponding methods in [9]. Furthermore, in Table 4.1, Alg. 2.7 and Alg. 2.8 fail to converge to the solution of Problem 1 at the given initial approximation $x_0 = -0.5$, while their corresponding proposed modified methods converged in few iterations. Thus, the proposed methods can be considered as alternative methods where the compared methods fail to converge to solution.

4. Conclusion
In this paper, a modified Algorithm 2.2 developed in Noor et al. [9] was proposed. The modification improves the efficiencies of all methods developed from Algorithm 2.2 in Noor et al. [9]. Numerical results of the proposed methods shows that they are efficient and can be used as alternative to existing methods.

Competing Interests
The authors do not have any competing interest in the manuscripts.

REFERENCES


