An efficient family of weighted-Newton methods with optimal eighth order convergence

Janak Raj Sharma *, Himani Arora
Department of Mathematics, Sant Longowal Institute of Engineering and Technology, Longowal-148106, Punjab, India

**Abstract**

Based on Newton’s method, we present a family of three-point iterative methods for solving nonlinear equations. In terms of computational cost, the family requires four function evaluations and has convergence order eight. Therefore, it is optimal in the sense of Kung–Traub hypothesis and has the efficiency index 1.682 which is better than that of Newton’s and many other higher order methods. Some numerical examples are considered to check the performance and to verify the theoretical results. Computational results confirm the efficient and robust character of presented algorithms.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

In this study, we consider iterative methods for solving the nonlinear equation \( f(x) = 0 \), where \( f: D \subseteq \mathbb{R} \to \mathbb{R} \) is a scalar function on an open interval \( D \). Newton’s method is probably the most widely used algorithm for solving such equations, which starts with an initial approximation \( x_0 \) closer to a root (say, \( r \)) and generates a sequence of successive iterates \( \{x_i\}_{i=0}^{\infty} \) converging quadratically to the root \( r \). It is given by

\[
    x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)},
\]

where \( f'(x) \) is the first order derivative of the function \( f(x) \). In order to improve the local order of convergence of Newton’s method a number of modified methods have been developed in the literature, see for example [1–27] and references therein.

Ostrowski [1] proposed the concept of efficiency index as a measure for comparing the efficiency of different methods. This index is described by \( E = \frac{p}{m^{1/n}} \), wherein \( p \) is the order of convergence and \( m \) is the total number of function evaluations needed per iteration. Later on, Kung and Traub [28] conjectured that multipoint methods without memory [29] based on \( m \) function evaluations have the optimal order of convergence \( 2^{m-1} \). Multipoint methods with this property are usually called optimal methods. For example, with three function evaluations a two-point method of optimal fourth order convergence can be constructed (see [1–8]) and with four function evaluations a three-point method of optimal eighth order convergence can be developed (see [9–20]). A more extensive list of references as well as a survey on progress made on the class of multipoint methods may be found in the recent book by Petković et al. [30].

In this paper, our aim is to develop an iterative method that may satisfy the basic requirements of generating a quality numerical algorithm, that is, an algorithm which has (i) high convergence speed, (ii) minimum computational cost, and (iii) simple structure. Thus, we derive a new family of three-point methods with optimal eighth order of convergence. The
scheme is composed of three steps of which first two steps consist of any fourth order method with the base as well-known Newton’s iteration and the third step is weighted Newton iteration. Rest of the paper is organized as follows. In Section 2 the new family is developed and its convergence analysis is discussed. The theoretical results proved in Section 2 are verified in Section 3 through numerical experimentation along with a comparison of the new methods with the existing methods of same class.

2. The methods and analysis of convergence

Based on the above considerations of a quality numerical algorithm, we begin with the three-point iteration scheme

\[
\begin{align*}
    y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
    z_i &= M_4(x_i, y_i), \\
    x_{i+1} &= z_i - \frac{f'(x_i) - f[y_i, x_i] + f[z_i, y_i] f(z_i)}{2f[z_i, y_i] - f[z_i, x_i]} f'(x_i).
\end{align*}
\]

(2)

Here \(M_4(x_i, y_i)\) is any optimal fourth order scheme with the base as Newton’s iteration \(y_i\) and \(f[\cdot, \cdot]\) is Newton’s first order divided difference. Through the following theorem we prove that this scheme has eighth order of convergence.

**Theorem 1.** Let the function \(f(x)\) be sufficiently differentiable in a neighborhood of its zero \(r\) and \(M_4(x_i, y_i)\) is an optimal fourth order method which satisfies

\[
z_i - r = B_0 e_i^4 + B_1 e_i^5 + B_2 e_i^6 + B_3 e_i^7 + B_4 e_i^8 + O(e_i^9),
\]

(3)

where \(B_0 \neq 0\) and \(e_i = x_i - r\). If an initial approximation \(x_0\) is sufficiently close to \(r\), then the order of convergence of (2) is at least 8.

**Proof.** Let \(\tilde{e}_i = y_i - r\) and \(\tilde{e}_i = z_i - r\) be the errors in the \(i\)-th iteration. Using Taylor’s expansion of \(f(x_i)\) about \(r\) and taking into account that \(f(r) = 0\) and \(f'(r) \neq 0\), we have

\[
f(x_i) = f'(r)[e_i + A_2 e_i^2 + A_3 e_i^3 + \cdots + A_8 e_i^8 + O(e_i^9)].
\]

(4)

where \(A_k = (1/k!)f^{(k)}(r)/f'(r), k = 2, 3, 4, \ldots\).

Also,

\[
f'(x_i) = f'(r)[1 + 2A_2 e_i + 3A_3 e_i^2 + \cdots + 9A_8 e_i^8 + O(e_i^9)].
\]

(5)

Substitution of (4) and (5) in the first step of (2) gives

\[
\tilde{e}_i = A_2 \tilde{e}_i^2 + (-2A_2^2 + 2A_3) \tilde{e}_i^3 + (4A_2^3 - 7A_2 A_3 + 3A_4) \tilde{e}_i^4 - 2(4A_2^4 - 10A_2^2 A_3 + 3A_5 + 5A_2 A_4 - 2A_5) \tilde{e}_i^5
\]

\[
+ (16A_2^5 - 52A_2^3 A_3 + 33A_2 A_4^2 + 28A_2^2 A_4 - 17A_2 A_5 - 13A_3 A_4 + 5A_6) \tilde{e}_i^6
\]

\[
- 2(16A_2^6 - 64A_2^4 A_3 + 63A_2^3 A_4^2 - 9A_4^3 + 36A_2^2 A_5 + 64A_2 A_6 + 64A_4 - 18A_5 A_5 + 11A_3 A_5)
\]

\[
+ 8A_2 A_6 - 3A_7) \tilde{e}_i^7 + (64A_2 A_6 - 304A_2^2 A_4 + 408A_2 A_5 A_4 - 135A_2 A_6^2 + 176A_4 A_6 - 348A_2^2 A_4 A_4 + 75A_2 A_4^2 A_4 + 64A_2 A_5 A_4^2 - 92A_5 A_5 A_4 + 118A_2 A_5 A_6 - 31A_4 A_5 + 44A_5 A_6 - 27A_2 A_6 - 19A_2 A_7 + 7A_8) \tilde{e}_i^8 + O(\tilde{e}_i^9).
\]

(6)

Expanding \(f(y_i)\) about \(r\) we obtain

\[
f(y_i) = f'(r)[\tilde{e}_i + A_2 \tilde{e}_i^2 + A_3 \tilde{e}_i^3 + A_4 \tilde{e}_i^4 + O(\tilde{e}_i^5)].
\]

(7)

Also, expansion of \(f(z_i)\) about \(r\) yields

\[
f(z_i) = f'(r)[\tilde{e}_i + A_2 \tilde{e}_i^2 + O(\tilde{e}_i^3)].
\]

(8)

Substituting Eqs. (3)–(8) in the third step of (2) and simplifying, we obtain

\[
e_{i+1} = x_{i+1} - r = -B_0(A_2^3 + A_2(-A_4 + B_0)) \tilde{e}_i^8 + O(\tilde{e}_i^9).
\]

(9)

This completes the proof of Theorem 1. □

Thus, the scheme (2) defines a new family of three-point eighth order methods which utilizes four function evaluations, namely \(f(x_i), f(y_i), f(z_i)\) and \(f'(x_i)\). This family is, therefore, optimal in the sense of Kung–Traub conjecture. The efficiency index \(E\) for this family is, \(8^{1/4} \approx 1.682\) which is better than the efficiency of Newton’s method, fourth order methods [1–8], sixth order methods [21–24] and seventh order methods [25–27] whose \(E\)-values are: \(2^{1/2} \approx 1.414, 4^{1/3} \approx 1.587, 6^{1/4} \approx 1.565\) and \(7^{1/4} \approx 1.627\), respectively. However, this \(E\)-value is equal to the \(E\)-values of existing eighth order methods cited in the previous section. Some simple members of the family (2) are as follows:
This method has the following error equation
\[ e_{i+1} = -A_2(A_2^3 - A_3)(A_2^3 - A_3^2 + A_2^2 - A_2 A_4)e_i^6 + O(e_i^8). \]

In this method, the fourth order scheme \( M_4(x_i, y_i) \) is the well-known Ostrowski's iteration [1].

**Method 2 (NM2):**

\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= y_i - \left( \frac{2}{f[y_i, x_i]} - \frac{1}{f'(x_i)} \right) f(y_i), \\
x_{i+1} &= z_i - \frac{f'(x_i)}{2f[z_i, y_i] - f[z_i, x_i]} f(z_i),
\end{align*}
\]

This method has the following error equation
\[ e_{i+1} = -A_2(3A_2^3 - A_3)(3A_2^3 - A_3^2 + A_2^2 - A_2 A_4)e_i^6 + O(e_i^8). \]

Here, fourth order scheme is the Ostrowski-like iteration considered in [8].

**Method 3 (NM3):**

\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= y_i - \left( 3 - 2f[y_i, x_i] \right) f'(x_i), \\
x_{i+1} &= z_i - \frac{f'(x_i) - f[y_i, x_i] + f[z_i, y_i] f(z_i)}{2f[z_i, y_i] - f[z_i, x_i]} f'(x_i),
\end{align*}
\]

which follows the error equation
\[ e_{i+1} = -A_2(5A_2^3 - A_3)(5A_2^3 - A_3^2 + A_2^2 - A_2 A_4)e_i^6 + O(e_i^8). \]

In this case we have considered a new fourth order scheme.

### 3. Numerical results

In order to illustrate the convergence behavior of new methods and to check the validity of the theoretical results we employ NM1, NM2 and NM3 to solve some nonlinear equations. We also compare the present methods with some existing optimal eighth order methods. For example, we choose the methods proposed by Bi–Wu–Ren [17], Thukral–Petković [13], Liu–Wang [10], Cordero–Torrregrosa–Vassileva [19] and Khan–Fardi–Sayevand [12]. These methods are given as follows:

**Bi–Wu–Ren Method (BWRM):**

\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= y_i - \frac{2f'(x_i) - f(y_i)}{2f'(x_i)} f'(x_i), \\
x_{i+1} &= z_i - \frac{f'(x_i) + \gamma f(z_i)}{f[z_i, y_i] + f[z_i, x_i] f(z_i)} f'(x_i),
\end{align*}
\]

where \( f[z_i, x_i] = \frac{f(z_i) - f(x_i)}{z_i - x_i} \).

**Thukral–Petković Method (TPM):**

\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= y_i - \frac{f(x_i) + bf(y_i)}{f(x_i) + (b - 2f(y_i)) f'(x_i)}, \\
x_{i+1} &= z_i - \left[ \frac{\phi(t) + f(z_i)}{f(y_i) - af(z_i)} + \frac{4f(z_i)}{f(x_i)} \right] f'(x_i),
\end{align*}
\]

where \( \phi(t) = 1 + 2t + (5 - 2b)t^2 + (12 - 12b + 2b^2)t^3 \) and \( t = \frac{f(y_i)}{f(x_i)} \).
\[ f(x) = x^3 + x^4 + 4x^2 - 15; \text{ see [18]} \]
\[ f_2(x) = e^{x^2} - \tan^{-1}(x); \text{ [10]} \]
\[ f_3(x) = e^{-x} - \cos(x + 1) + x^3 + 1; \text{ [14]} \]
\[ f_4(x) = e^{x^2 + x} - \cos(x + 1) + x^3 + 1; \text{ [30]} \]
\[ f_5(x) = \log(x^2 + x + 2) - x + 1; \text{ [30]} \]
\[ f_6(x) = (x - 2)(x^{10} + x + 1)e^{-x^2}; \text{ [30]} \]

<table>
<thead>
<tr>
<th>( f(x) )</th>
<th>( x_0 )</th>
<th>( r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1(x) )</td>
<td>3</td>
<td>1.3474280989683050...</td>
</tr>
<tr>
<td>( f_2(x) )</td>
<td>3.25</td>
<td>3.0</td>
</tr>
<tr>
<td>( f_3(x) )</td>
<td>-1</td>
<td>0.0</td>
</tr>
<tr>
<td>( f_4(x) )</td>
<td>-0.7</td>
<td>-1</td>
</tr>
<tr>
<td>( f_5(x) )</td>
<td>3</td>
<td>4.1525907367571583...</td>
</tr>
<tr>
<td>( f_6(x) )</td>
<td>2.2</td>
<td>2.0</td>
</tr>
</tbody>
</table>

Liu–Wang Method (LWM):
\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= y_i - \frac{f(x_i) - f(y_i)}{f(x_i) - 2f(y_i)}, \\
x_{i+1} &= z_i - \left( \frac{f(x_i) - f(y_i)}{f(x_i) - 2f(y_i)} \right)^2 + \frac{4f(z_i)}{f(x_i) + 2f(z_i)} \frac{f'(z_i)}{f'(x_i)}.
\end{align*}
\]

Cordero–Torregrosa–Vassileva Method (CTVM):
\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= x_i - \frac{f(x_i) - f(y_i)}{f(x_i) - 2f(y_i)} f'(x_i), \\
x_{i+1} &= u_i - \frac{3(\beta_2 + \beta_3)(u_i - z_i) f(z_i)}{\beta_1(u_i - z_i) + \beta_2(y_i - x_i) + \beta_3(z_i - x_i) f'(x_i)},
\end{align*}
\]

where \( \beta_i \in \mathbb{R} \) (\( i = 1, 2, 3 \)), \( \beta_2 + \beta_3 \neq 0 \) and \( u_i = z_i - \frac{f(z_i)}{f'(x_i)} \left( \frac{f(x_i) - f(y_i)}{f(x_i) - 2f(y_i)} + \frac{1}{2} \frac{f(z_i)}{f'(y_i)} \right)^2. \]

Khan–Fardi–Sayevand Method (KFSM):
\[
\begin{align*}
y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\
z_i &= y_i - \frac{f^2(x_i) - 2f(x_i)f(y_i) + \alpha f^2(y_i)}{f(x_i) + \alpha f'(y_i)^2} f'(x_i), \\
x_{i+1} &= z_i - \frac{1}{1 + qv_i^2} \frac{K - C(y_i - z_i) - D(x_i - y_i)}{K - D(y_i - z_i)^2},
\end{align*}
\]

where \( q_i = \frac{f(z_i)}{f'(x_i)}, D = \frac{f(x_i) - f(y_i)}{(x_i - y_i)(x_i - z_i)}, C = \frac{H - K}{(x_i - z_i)^2}, H = \frac{f(x_i) - f(y_i)}{y_i - z_i}, K = \frac{f(y_i) - f(z_i)}{y_i - z_i}. \)

The considered test functions with corresponding root and its initial approximation are displayed in Table 1. The references from which the functions are selected are also shown in this table. All computations are performed in the programming package Mathematica [31] using multiple-precision arithmetic with 4096 significant digits. The computer specifications are: Intel (R) Core (TM) i5-2430M CPU @ 2.40 GHz (32-bit Machine) Microsoft Windows 7 Ultimate 2009. For the parameters used in BWRM, TPM, LWM, CTVM and KFSM we choose those values which the respective authors have chosen in their numerical work. To verify the theoretical order of convergence, we calculate the computational order of convergence \( (p_c) \) using the formula [32]
\[
p_c = \frac{\log |f(x_i)/f(x_{i-1})|}{\log |f(x_{i-1})/f(x_{i-2})|}
\]

Taking into consideration the last three approximations in the iterative process. In the comparison of performance of methods, we also include CPU time utilized in the execution of program which is computed by the Mathematica command “TimeUsed[ ”].

The errors \( |x_{i+1} - x_i| \) for \( i = 1, 2, 3 \) of approximations to the corresponding zeros of the functions \( f_1(x) - f_6(x) \), the computational order of convergence \( (p_c) \) and the mean CPU time (CPUtime) are displayed in Table 2, where \( a \times 10^{-b} \) denotes \( a \times 10^{-b} \). The mean CPU time is calculated by taking the mean of 100 performances of the program, where we use \( |(x_{i+1} - x_i) + |f(x_i)|| < 10^{-200} \) as the stopping criterion in single performance of the program.

It can be seen from the numerical results displayed in Table 2 that the proposed methods support the theoretical results proved in Section 2 and show consistent convergence behavior like existing methods. Moreover, the CPUtime displayed in the last column of Table 2 verifies the efficient character of the present techniques when compared with the CPUtime
of existing methods of same nature. Similar numerical experimentations, carried out for a number of different problems, confirmed the above conclusions to a great extent.

Acknowledgments

The authors wish to thank the anonymous referees for their valuable suggestions on the first version of this paper. The author, Himani Arora, would like to be thankful to the University Grants Commission, New Delhi for the financial support under the Grant F. 2-13/2009 (SA-I).
References