New Modification of Chebyshev’s Method with Seventh-Order Convergence

M. N. Muhaijir, M. Soleh and E. Safitri

Department of Mathematics, Faculty of Science and Technology
UIN Sultan Syarif Kasim Riau, Pekanbaru 28293, Indonesia

Abstract

In this paper, based on Chebyshev’s method, a new family of seventh-order methods for solving nonlinear equations is derived. This method includes finite difference and Lagrange interpolation to eliminate derivative. The results of this research show that at each iteration of this method requires three evaluation of the function and one evaluation if its first derivative, which has the efficiency index 1.6266. Numerical simulations show that the effectiveness and performance of the new modification in solving nonlinear equations are encouraging.

Mathematics Subject Classification: 65H05, 65D99

Keywords: Chebyshev’s method, finite differences, Lagrange interpolation, order of convergence

1 Introduction

This paper is concerned with finding solutions to the scalar, nonlinear equation

\[ f(x) = 0, \]  

where the variable \( x \) runs in an interval \([a, b]\). The topic provides us with an opportunity to discuss various issues and concepts that arise in more general
circumstances. Many iteration methods can be used to solve equation (1). Famous iteration method to solve (1) is Newton’s method written as

\[ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, 2, \ldots \text{ and } f'(x) \neq 0, \quad (2) \]

where order convergence is quadratic for simple roots [2]. To increase the convergence of (2) many researcher have modified it was can see in [5–8, 12, 15, 17, 18]. Another iteration method can be used to solve equation (1) is Chebyshev’ method in the form of

\[ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left(1 + \frac{1}{2} \frac{f(x_n)f''(x_n)}{f'(x_n)^2}\right), \quad (3) \]

ded that the order of convergence of three [1, 4]. Kou, et al. [14] modificated the form (3) by approxmiate \( f''(x_n) \) using the finite difference thus obtained

\[ y_n = x_n + \gamma \frac{f(x_n)}{f'(x_n)}, \quad (4) \]

\[ x_{n+1} = x_n - \left(1 + \frac{1}{2\gamma} \frac{f'(y_n) - f'(x_n)}{f'(x_n)}\right) \frac{f(x_n)}{f'(x_n)}, \quad (5) \]

which also has a three-order convergence [14]. Another modification that resulted in four-order convergence of (3) can be seen at [3,10,11].

The processes of removing the derivatives usually increase the number of function evalution per iteration. In this paper, we used the tecnique of the combination of Newtons method and Chebyshevs method into a three-step iteration method. We also incorporate polinomial degre two to approximate the first derivative and finite difference to approximate the second derivative in second step and Lagrange interpolation to approximate the first derivative in the third step. In our methods not only increase the order of the method as hig as possible but also reduce the number of function evalutions and improve the efficiency index of the composed method. The discussion of the new method and their convergence and analysis are carried out in Section 2. Then, in Section 3 we perform numerical simulations using some test functions, and compare the new method with some other methods.

# 2 Proposed Methods

In this section, for construction of new iterative methods, we use iterative methods given by equations (2) and (3). We consider the following three-step
method:

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

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

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

We now approximate \( f'(y_n) \) in second step, to reduce the number of per iteration by combination of already data in the past steps. Toward this end, an estimation of the function \( p_2(t) \) is taken into consideration as follows

\[ p_2(t) = a + b(t - x_n) + c(t - x_n)^2, \]

\[ p_2(t) = b + 2c(t - x_n). \]

By substituting value \( t = y_n \) and \( t = x_n \) to equation (9) and (10), we get

\[ p_2(y_n) = f(y_n) = a + b(y_n - x_n) + c(y_n - x_n)^2, \]

\[ p_2'(y_n) = f'(y_n) = b + 2c(y_n - x_n), \]

\[ p_2(x_n) = f(x_n) = a, \]

\[ p_2'(x_n) = f'(x_n) = b. \]

Based on equation (11)–(14), we could easily obtain the unknown parameters. So that

\[ f'(y_n) = 2\left(\frac{f(y_n) - f(x_n)}{y_n - x_n}\right) - f'(x_n) =: N_1(x_n, y_n). \]

We replace \( f''(y_n) \) in equation (7) with a finite difference formula [16], that is

\[ f''(y_n) \approx \frac{f'(y_n) - f'(x_n)}{y_n - x_n}. \]

By Substituting equation (15) to (16), we have

\[ f''(y_n) = 2\left(\frac{(f(y_n) - f(x_n)) - (y_n - x_n)f'(x_n)}{(x_n - y_n)^2}\right) := N_2(x_n, y_n). \]

Furthermore, we approximate \( f'(z_n) \) by a derivative of Lagrange interpolation polynomial \( L_2(x) \) passing the points \( (x_n, f(x_n)), (y_n, f(y_n)), \) and \( (z_n, f(z_n)), \) yield

\[ L_2'(z_n) = \frac{2z_n - (y_n + z_n)}{(x_n - y_n)(x_n - z_n)} f(x_n) + \frac{2z_n - (x_n + z_n)}{(y_n - x_n)(y_n - z_n)} f(y_n) \]

\[ + \frac{2z_n - (x_n + y_n)}{(z_n - x_n)(z_n - y_n)} f(z_n). \]
Simplifying the equation (18) yields
\[ L'_2(z_n) = f[x_n, z_n] + f[y_n, z_n] - f[x_n, y_n]. \] (19)

where
\[ f[x_n, z_n] = \frac{f(z_n) - f(x_n)}{z_n - x_n}, \] (20)
\[ f[y_n, z_n] = \frac{f(z_n) - f(y_n)}{z_n - y_n}, \] (21)
\[ f[x_n, y_n] = \frac{f(y_n) - f(x_n)}{y_n - x_n}. \] (22)

Assume that \( f'(z_n) \approx L'(z_n) \), so by substituting the equation (15) and (17) to (7) and equation (19) to (8) be obtained
\[ y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \] (23)
\[ z_n = y_n - \frac{f(y_n)}{N_1(x_n, y_n)} \left( 1 + \frac{1}{2} \frac{f(y_n)}{N_2(x_n, y_n)^2} \right), \] (24)
\[ x_{n+1} = z_n - \frac{f(z_n)}{f[x_n, z_n] + f[y_n, z_n] - f[x_n, y_n]}. \] (25)

**Theorem 1** Assume that functions \( f \) is sufficiently differentiable and \( f \) has a simple \( \alpha \in I \). If the initial point \( x_0 \) is sufficiently to \( \alpha \), then the method of iteration in equation (23)–(25) have seventh-order convergence and satisfies the following error equation:
\[ e_{n+1} = (c_2^2 c_3^3) e_7^7 + O(e_n^8), \]
where \( e_n = x_n - \alpha \) and \( c_k = f^{(k)}(\alpha)/k! f'(\alpha) \).

**Proof.** Let \( \alpha \) be simple root of the equation \( f(x) = 0 \), then \( f'(\alpha) \neq 0 \). Furthermore, using Taylor expansion of the \( f(x) \) about \( x_n = \alpha \), we obtain
\[ f(x_n) = f(\alpha) + (x_n - \alpha)f'(\alpha) + \frac{(x_n - \alpha)^2}{2!} f''(\alpha) + \cdots + O(x_n - \alpha)^8. \] (26)

Because \( f(\alpha) = 0 \) so that the equation (26) can be rewritten in the form
\[ f(x_n) = f'(\alpha)(e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + \cdots + O(e_n^8)), \] (27)
where \( c_j = \frac{f^j(\alpha)}{j!f'(\alpha)} \), \( j = 2, 3, \ldots, 7 \).

Furthermore, in the same way do the Taylor expansion again \( f'(x_n) \) about \( x_n = \alpha \) so after a simplified, we obtain

\[
f'(x_n) = f'(\alpha)(1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + \cdots + \mathcal{O}(e_n^7)). \tag{28}
\]

Based on the equation (27) and (28), we get

\[
\frac{f(x_n)}{f'(x_n)} = \frac{e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + \cdots + \mathcal{O}(e_n^8)}{1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + \cdots + \mathcal{O}(e_n^7)}. \tag{29}
\]

By considering a geometri series and after simplified the equation (29), we have

\[
\frac{f(x_n)}{f'(x_n)} = e_n - c_2e_n^2 + 2(c_2 - c_3)e_n^3 + \cdots + \mathcal{O}(e_n^8) \tag{30}
\]

On substituting equation (30) to (23), we get

\[
y_n = \alpha + c_2e_n^2 + 2(c_3 - c_2^2)e_n^3 + \cdots + \mathcal{O}(e_n^8). \tag{31}
\]

Use the Taylor expansion to determine \( f(y_n) \) about \( y_n = \alpha \), we have

\[
f(y_n) = f'(\alpha)(c_2e_n^2 + 2(c_3 - c_2^2)e_n^3 + \cdots + \mathcal{O}(e_n^8)), \tag{32}
\]

Using equation(27), (28), (31), and (32), we get

\[
N_1(y_n) = 1 + (-c_3 + 2c_2^2)e_n^2 + (-2c_4 - 4c_2^2 + 6c_2c_3)e_n^3 + \cdots + \mathcal{O}(e_n^8) \tag{33}
\]

and

\[
N_2(y_n) = 2c_2 + 4c_3e_n + (2c_2c_3 + 6c_4)e_n^2 + (4c_3^2 - 4c_2c_3^2 + 4c_2c_4 + 8c_5)e_n^3 + \cdots + \mathcal{O}(e_n^8). \tag{34}
\]

Furthermore, the division equation (32) by (33), we have

\[
\frac{f(y_n)}{N_1(y_n)} = c_2e_n^2 + (2c_3 - 2c_2^2)e_n^3 + (3c_4 - 6c_2c_3 + 3c_2^2)e_n^4 + \cdots + \mathcal{O}(e_n^9). \tag{35}
\]

By Substituting equation (32), (33), (34) and (35) to (24), we obtain

\[
z_n = \alpha - c_2c_3e_n^4 + (2c_3c_2^2 - 2c_2c_4 - 2c_3^2)e_n^5 + \cdots + \mathcal{O}(e_n^8). \tag{36}
\]

Furthermore, applying Taylor expansion of \( f(z_n) \) about \( z_n = \alpha \), we obtain

\[
f(z_n) = f'(\alpha)\left(- c_2c_3e_n^4 + (2c_3c_2^2 - 2c_2c_4 - 2c_3^2)e_n^5 + \cdots + \mathcal{O}(e_n^8) \right). \tag{37}
\]
To obtain the $f[x_n, z_n]$ substituting equation (27), (36) and (37) to (20), we get
\[ f[x_n, z_n] = f'(\alpha) \left( 1 + c_2 e_n + c_3 e_n^2 + c_4 e_n^3 + (c_5 - c_3 c_2^2) e_n^4 + \cdots + O(e_n^8) \right). \] (38)

Using the same strategy, $f[y_n, z_n]$ can be obtained by substituting equation (31), (32), (36) and (37) to (21) we obtain
\[ f[y_n, z_n] = f'(\alpha) \left( 1 + c_2^2 e_n + (c_2^2 + c_3) e_n^2 + (c_4 - 2 c_2^3 + 3 c_2 c_3) e_n^3 + \cdots + O(e_n^8) \right). \] (39)

To obtain $f[x_n, y_n]$ substituting (27), (31) and (32) to (22), we obtain
\[ f[x_n, y_n] = f'(\alpha) \left( 1 + c_2 e_n + (c_2 + c_3) e_n^2 + (c_4 - 2 c_2^3 + 3 c_2 c_3) e_n^3 + \cdots + O(e_n^8) \right). \] (40)

Combining equation (38), (39), and (40), we get
\[ f[x_n, z_n] + f[y_n, z_n] - f[x_n, y_n] = f'(\alpha) \left( 1 - c_2 c_3 e_n^3 + \cdots + O(e_n^8) \right). \] (41)

Dividing equation (37) by (41), we have
\[ \frac{f(z_n)}{f[x_n, z_n] + f[y_n, z_n] - f[x_n, y_n]} = -c_2 c_3 e_n^4 + \cdots + O(e_n^8). \] (42)

Substituting equation (36) and (42) to (25), so that we obtain
\[ x_{n+1} = \alpha + (c_2^2 c_3^2) e_n^7 + O(e_n^8). \] (43)

Therefore $e_{n+1} = x_{n+1} - \alpha$, then from equation (43) we get
\[ e_{n+1} = (c_2^2 c_3^2) e_n^7 + O(e_n^8), \]
which completes the proof of the theorem. \(\square\)

### 3 Numerical Experiments

In this section will be recalculated order of convergence of the method of computing by using the following equation:
\[ COC = \frac{\ln |(x_{n+1} - \alpha)/(x_n - \alpha)|}{\ln |(x_n - \alpha)/(x_{n-1} - \alpha)|}. \] (44)

Calculations using software with 800 digits of accuracy and tolerance $\epsilon = 1.0 \times 10^{-100}$. The stopping criteria of the iteration are $|x_{n+1} - x_n| < \epsilon$ and $|f(x_{n+1})| < \epsilon$, $x_{n+1}$ is taken as the exact root $\alpha$ computed.
Numerical simulation was performed to compare modification Chebyshev method (MCM) with some other methods, such as Newton method (NM) (2), Chebyshev method (CM) (3), Fardi et al. Method (FM) [9], and Khattri-Argyros method (KAM) [13]. The function used is as follows:

\[
\begin{align*}
  f_1(x) &= \sin^2(x) - x^2 + 1, & \alpha \in (1.0, 1.5), \\
  f_2(x) &= x^3 + 4x - 10, & \alpha \in (1.5, 2.0), \\
  f_3(x) &= x^2 - e^x - 3x + 2, & \alpha \in (0.0, 0.5), \\
  f_4(x) &= (x - 1)^3 - 2, & \alpha \in (2.0, 2.5), \\
  f_5(x) &= xe^{x^2} - \sin^2(x) + 3\cos(x) + 5, & \alpha \in (-1.5, -1.0), \\
  f_6(x) &= (x + 2)e^x - 1, & \alpha \in (-1.0, -0.5).
\end{align*}
\]

In Table 1, we give initial value \((x_0)\), number of iteration \((N)\), and the computational order of convergence \((COC)\). An asterisk (*) on the number of iterations indicates that the method converges to different roots and the sign (−) in Table 1 indicates that the method can not find the root. Table 1 shows a comparison of the number of iterations and \(COC\) several methods to resolve the above functions including NM, CM, FM, KM, and MCM for some given initial values.

**Table 1: Comparison of Iteration and \(COC\)**

<table>
<thead>
<tr>
<th>(f(x))</th>
<th>(x_0)</th>
<th>(N)</th>
<th>(COC)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NM</td>
<td>CM</td>
</tr>
<tr>
<td>(f_1(x))</td>
<td>0.8</td>
<td>9</td>
<td>5*</td>
</tr>
<tr>
<td>&amp;</td>
<td>2.0</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>(f_2(x))</td>
<td>-1.0</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>&amp;</td>
<td>3.0</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>(f_3(x))</td>
<td>-6.0</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>&amp;</td>
<td>2.5</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>(f_4(x))</td>
<td>-4.0</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>&amp;</td>
<td>5.0</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>(f_5(x))</td>
<td>-1.2</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>&amp;</td>
<td>-0.6</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>(f_6(x))</td>
<td>-2.0</td>
<td>16</td>
<td>-</td>
</tr>
<tr>
<td>&amp;</td>
<td>2.0</td>
<td>11</td>
<td>7</td>
</tr>
</tbody>
</table>

The computational in Table 1 show that MCM requires less iteration than NM and CM, and MCM has comparable to FM and KM. Therefore, the proposed new method is of practical interest and compete with NM, CM, FM, and KM.
Table 2 shows a comparison of absolute value the functions $|f(x_{n+1})|$ of several methods to resolves the above function including NM, CM, FM, KM, and MCM for some given initial values.

<table>
<thead>
<tr>
<th></th>
<th>$f(x)$</th>
<th>$x_0$</th>
<th>NM</th>
<th>CM</th>
<th>FM</th>
<th>KAM</th>
<th>MCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1(x)$</td>
<td>0.8</td>
<td>5.20e-107</td>
<td>3.09e-128</td>
<td>2.32e-375</td>
<td>3.68e-641</td>
<td>2.68e-526</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>8.24e-129</td>
<td>2.52e-286</td>
<td>1.25e-196</td>
<td>4.01e-196</td>
<td>8.59e-174</td>
<td></td>
</tr>
<tr>
<td>$f_2(x)$</td>
<td>-1.0</td>
<td>1.79e-187</td>
<td>5.39e-209</td>
<td>2.67e-146</td>
<td>1.80e-646</td>
<td>2.26e-127</td>
<td></td>
</tr>
<tr>
<td>$f_3(x)$</td>
<td>-6.0</td>
<td>1.41e-139</td>
<td>6.66e-144</td>
<td>6.48e-117</td>
<td>2.44e-115</td>
<td>1.84e-557</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>1.71e-125</td>
<td>9.54e-218</td>
<td>1.28e-442</td>
<td>8.20e-449</td>
<td>1.22e-181</td>
<td></td>
</tr>
<tr>
<td>$f_4(x)$</td>
<td>-4.0</td>
<td>2.40e-187</td>
<td>1.94e-235</td>
<td>2.08e-484</td>
<td>1.21e-683</td>
<td>8.39e-374</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.0</td>
<td>1.12e-110</td>
<td>4.31e-175</td>
<td>4.82e-495</td>
<td>8.21e-490</td>
<td>1.86e-286</td>
<td></td>
</tr>
<tr>
<td>$f_5(x)$</td>
<td>-1.2</td>
<td>4.47e-106</td>
<td>5.23e-294</td>
<td>2.92e-186</td>
<td>2.91e-186</td>
<td>6.15e-212</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.6</td>
<td>1.91e-169</td>
<td>1.69e-147</td>
<td>9.30e-279</td>
<td>9.58e-279</td>
<td>5.21e-232</td>
<td></td>
</tr>
<tr>
<td>$f_6(x)$</td>
<td>-2.0</td>
<td>1.25e-151</td>
<td>-</td>
<td>2.17e-144</td>
<td>2.69e-584</td>
<td>2.93e-219</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>3.84e-149</td>
<td>7.79e-129</td>
<td>-</td>
<td>-</td>
<td>3.45e-378</td>
<td></td>
</tr>
</tbody>
</table>

The computational results presented in Table 2 shows that some of cases, the MCM has a smaller absolute value compared to the other methods.

## 4 Conclusion

In this paper, we present and analyze new modification of Chebyshev method for solving nonlinear equations. This method is free from second derivatives and new method requires three evaluations of the function and one its derivatives and therefor has the efficiency index equal to 1.6266. Several numerical test demonstrate that new method proposed in this paper is more efficient and perform better than Newton method and Chebyshev method.

## References

https://doi.org/10.1016/j.cam.2007.07.022

New modification of Chebyshev’s method with seventh-order convergence


Received: July 24, 2017; Published: September 14, 2017