Some fourth-order nonlinear solvers with closed formulae for multiple roots

S.G. Li, L.Z. Cheng, B. Neta *
School of Science, National University of Defense Technology, Changsha, 410073, China

A R T I C L E   I N F O
Article history:
Received 2 July 2009
Accepted 25 August 2009

Keywords:
Newton’s method
Multiple roots
Nonlinear equations
Iterative methods
Root finding

A B S T R A C T
In this paper, we present six new fourth-order methods with closed formulae for finding multiple roots of nonlinear equations. The first four of them require one-function and three-derivative evaluation per iteration. The last two require one-function and two-derivative evaluation per iteration. Several numerical examples are given to show the performance of the presented methods compared with some known methods.

1. Introduction
Finding the roots of nonlinear equations is very important in numerical analysis and has many applications in engineering and other applied sciences. In this paper, we consider iterative methods to find a multiple root \( \alpha \) of multiplicity \( m \), i.e., \( f^{(j)}(\alpha) = 0, j = 0, 1, \ldots, m - 1 \) and \( f^{(m)}(\alpha) \neq 0 \), of a nonlinear equation \( f(x) = 0 \).

The modified Newton’s method for multiple roots is written as [1]

\[
x_{n+1} = x_n - \frac{m f(x_n)}{f'(x_n)},
\]

which is quadratically convergent.

In recent years, some modifications of Newton’s method for multiple roots have been proposed and analyzed, most of which are of third-order convergence. For example, see Traub [2], Hansen and Patrick [3], Victory and Neta [4], Dong [5,6], Osada [7], Neta [8], Chun and Neta [9], Chun, Bae and Neta [10], etc. All of these methods require the knowledge of the multiplicity \( m \).

The third-order Chebyshev’s method for finding multiple roots [2,8] is given by

\[
x_{n+1} = x_n - \frac{m(3 - m)}{2} u_n - \frac{m^2 f(x_n)}{2} f''(x_n)
\]

(2)

where

\[
u_n = f(x_n) f'(x_n)^{-1}.
\]

Published by Elsevier Ltd
0898-1221/$ – see front matter. Published by Elsevier Ltd
doi:10.1016/j.camwa.2009.08.066
The cubically convergent Halley’s method which is a special case of the Hansen and Patrick’s method [3], is written as

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + \frac{f''(x_n)}{2f'(x_n)}.$$  \hspace{1cm} (4)

The third-order Osada method [7] is written as

$$x_{n+1} = x_n - \frac{1}{2} m(m+1)u_n + \frac{1}{2} (m-1)^2 f(x_n)/f'(x_n).$$  \hspace{1cm} (5)

There are, however, not yet so many fourth- or higher-order methods known that can handle the case of multiple roots. In [11], Neta and Johnson have proposed a fourth-order method requiring one-function and three-derivative evaluation per iteration. This method is based on the Jarratt method [12] given by the iteration function

$$x_{n+1} = x_n - \frac{f(x_n)}{a f'(x_n) + a_2 f'(y_n) + a_3 f'(\eta_n)},$$  \hspace{1cm} (6)

where

$$\begin{cases}
y_n = x_n - au_n, \\
u_n = f(x_n), \\
\eta_n = x_n - bu_n - cv_n.
\end{cases}$$  \hspace{1cm} (7)

Neta and Johnson [11] give a table of values for the parameters \(a, b, c, a_1, a_2, a_3\) for several values of \(m\).

Neta [13] has developed another fourth-order method requiring one-function and three-derivative evaluation per iteration. This method is based on Murakami’s method [14] given by

$$x_{n+1} = x_n - a_1 u_n - a_2 v_n - a_3 w_3(x_n) - \psi(x_n),$$  \hspace{1cm} (8)

where \(u_n\) is defined by (3), \(v_n, y_n\) and \(\eta_n\) are given by (7) and

$$\begin{align*}
w_3(x_n) &= \frac{f(x_n)}{f'(\eta_n)}, \\
\psi(x_n) &= \frac{b_1 f'(x_n) + b_2 f'(y_n)}{b_1 f'(x_n) + b_2 f'(y_n) + b_3 f'(\eta_n)}.
\end{align*}$$  \hspace{1cm} (9)

A table of values for the parameters \(a, b, c, a_1, a_2, a_3, b_1, b_2\) for several values of \(m\) is also given by Neta [13].

In [15], a fourth-order method is proposed,

$$\begin{cases}
y_n = x_n - \frac{2m}{m+2} u_n, \\
x_{n+1} = \frac{1}{2} m(m-2) \left( \frac{m-2}{m+2} \right) \frac{f'(y_n) - \frac{m^2}{m+2} f'(x_n)}{m f'(y_n)} u_n.
\end{cases}$$  \hspace{1cm} (10)

This method requires one-function and two-derivative evaluation per iteration.

The methods proposed in [11,13] do not have closed formulae there. In this paper, by further investigating these methods in [11,13], we present six fourth-order methods with closed formulae for multiple roots of nonlinear equations. The first four of these new methods require one-function and three-derivative evaluation per iteration. The last two require one-function and two-derivative evaluation per iteration. These last ones are more efficient since they require less functional evaluations. Finally, we use some numerical examples to compare the new fourth-order methods with some known third-order methods. From the results, we can see that the fourth-order methods can be competitive to these third-order methods and usually require less functional evaluations.

2. The fourth-order methods

For simplicity, we define

$$A_j = \frac{f^{(m+j)}(\alpha)}{f^{(m)}(\alpha)}, \quad j = 1, 2, \ldots,$$  \hspace{1cm} (11)

$$\mu = \frac{m-a}{m}.$$  \hspace{1cm} (12)

First, we consider the method (6) proposed in [11]. Let \(\alpha \in \mathbb{R}\) be a multiple root of multiplicity \(m\) of a sufficiently smooth function \(f(x)\). To maximize the order of convergence to the root \(\alpha\), we need to find six parameters \(a, b, c, a_1, a_2, a_3\).
Let $e_n, \hat{e}_n, \tilde{e}_n$ be the errors at the $n$th step, i.e.
\begin{align*}
  e_n &= x_n - \alpha, \\
  \hat{e}_n &= y_n - \alpha, \\
  \tilde{e}_n &= \eta_n - \alpha,
\end{align*}
where $y_n$ and $\eta_n$ are defined in (7).

Using the Taylor expansion of $f(x_n)$, $f(y_n)$ and $f(\eta_n)$ about $\alpha$, we have
\begin{align*}
  f(x_n) &= \frac{f^{(m)}(\alpha)}{m!} e_n^m [1 + C_1 e_n + C_2 e_n^2 + C_3 e_n^3 + C_4 e_n^4 + O(e_n^5)], \\
  f'(x_n) &= \frac{f^{(m)}(\alpha)}{(m-1)!} e_n^{m-1} [1 + D_1 e_n + D_2 e_n^2 + D_3 e_n^3 + D_4 e_n^4 + O(e_n^5)], \\
  f'(y_n) &= \frac{f^{(m)}(\alpha)}{(m-1)!} e_n^{m-1} [1 + D_1 \hat{e}_n + D_2 \hat{e}_n^2 + D_3 \hat{e}_n^3 + D_4 \hat{e}_n^4 + O(e_n^5)], \\
  f'(\eta_n) &= \frac{f^{(m)}(\alpha)}{(m-1)!} e_n^{m-1} [1 + D_1 \tilde{e}_n + D_2 \tilde{e}_n^2 + D_3 \tilde{e}_n^3 + D_4 \tilde{e}_n^4 + O(e_n^5)], \\
  u_n &= \frac{\eta_n}{m} [1 + (C_1 - D_1) e_n + (C_2 - D_2 + D_1^2 - C_1 D_1) e_n^2 + (C_3 - D_3 + (D_1 - C_1) D_2 + (D_2 - C_2 + C_1 D_1 - D_1^2) D_1) e_n^3 + O(e_n^4)],
\end{align*}
where $C_j = \frac{m!}{(m-j)!} A_j, D_j = \frac{(m-1)!}{(m-j-1)!} A_j$ and $u_n$ is defined by (3).

From (14), (15) and (18), we can get
\begin{align*}
  \tilde{e}_n &= e_n - a u_n = d_0 e_n + d_1 e_n^2 + d_2 e_n^3 + d_3 e_n^4 + d_4 e_n^5 + O(e_n^6),
\end{align*}
where
\begin{align*}
  d_0 &= \mu = \frac{m - a}{m}, \\
  d_1 &= -\frac{a(C_1 - D_1)}{m}, \\
  d_2 &= -\frac{a(C_2 - D_2 + D_1^2 - C_1 D_1)}{m}, \\
  d_3 &= -\frac{a(C_3 - D_3 + (D_1 - C_1) D_2 + (D_2 - C_2 + C_1 D_1 - D_1^2) D_1)}{m}, \\
  d_4 &= -\frac{a(C_4 - D_4 + (D_1 - C_1) D_3 + (D_2 - C_2 + C_1 D_1 - D_1^2) D_2 + (D_3 - C_3 + D_2 C_1 - 2D_2 D_1 + D_1 C_2 - D_1^2 C_1 + D_1^2) D_1)}{m}.
\end{align*}

Substituting into (16) we have
\begin{align*}
  f'(y_n) &= \frac{f^{(m)}(\alpha)}{(m-1)!} e_n^{m-1} [1 + D_1 \hat{e}_n + D_2 \hat{e}_n^2 + D_3 \hat{e}_n^3 + D_4 \hat{e}_n^4 + O(\hat{e}_n^5)],
\end{align*}
where
\begin{align*}
  \Delta &= (d_0 + d_1 \hat{e}_n + d_2 \hat{e}_n^2 + d_3 \hat{e}_n^3 + d_4 \hat{e}_n^4 + O(\hat{e}_n^5))^{m-1} \\
  &= d_0^{m-1} + (m - 1) d_0^{m-2} d_1 \hat{e}_n + (C_2^2 - 4C_2 D_1 d_0^{m-3} + (m - 1) d_2 d_0^{m-2} + C_3^2 - 4C_3 D_1 d_0^{m-3} + C_4^2 - 4C_4 D_1 d_0^{m-3} + 3C_5^2 - 4C_5 D_1 d_0^{m-3}) \hat{e}_n^3 \\
  &+ \left( (m - 1) d_4 d_0^{m-2} + 2C_2^2 - 4C_2 D_1 d_2 d_0^{m-3} + C_3^2 - 4C_3 D_1 d_2 d_0^{m-3} + C_4^2 - 4C_4 D_1 d_2 d_0^{m-3} + 3C_5^2 - 4C_5 D_1 d_2 d_0^{m-3} \right) d_0^{m-4} e_n^4 + O(e_n^5).
\end{align*}

In what follows, for simplicity, we let
\begin{align*}
  a &= \frac{2m}{m + 2},
\end{align*}
then $\mu = \frac{m}{m + 2}$. The error in $\eta_n$ is given by
\begin{align*}
  \tilde{e}_n &= e_n - bu_n - cv_n = d_0 e_n + d_1 e_n^2 + d_2 e_n^3 + d_3 e_n^4 + d_4 e_n^5 + O(e_n^6),
\end{align*}
where
\begin{align*}
  d_0 &= \frac{m - b}{m} - \frac{c}{m + 2} \mu^{-m}, \\
  d_1 &= \left[ \frac{(m^2 + 2m - 4)c}{m^2(m + 2)(m + 1)} \mu^{-m} + \frac{b}{m^2(m + 1)} \right] A_1.
\end{align*}
\[
\tilde{d}_2 = \left[ \frac{(m^4 + 5m^3 + 4m^2 - 8m - 16)c}{m^1(m + 1)^2(m + 2)^3} \mu^{-m} - \frac{b}{m^1(m + 1)} \right] A_1^2 + \left[ \frac{2(m^2 + 2m - 4)c}{m^1(m + 1)^2(m + 2)^3} \mu^{-m} + \frac{2b}{m^1(m + 1)(m + 2)} \right] A_2,
\]
\[
\tilde{d}_3 = \left[ \frac{3m^7 + 24m^6 + 67m^5 + 66m^4 - 64m^3 - 184m^2 - 144m - 32)c}{3m^3(m + 1)^3(m + 2)^4} \mu^{-m} + \frac{b}{m^4(m + 1)} \right] A_1^3 + \left[ \frac{3m^4 + 16m^3 + 20m^2 - 16m - 64)c}{m^3(m + 1)^2(m + 2)^4} \mu^{-m} + \frac{3m^4 + 4b}{m^3(m + 1)^2(m + 2)^4} \right] A_1 A_2,
\]
\[
\tilde{d}_4 = \left[ \frac{(3m^4 + 16m^3 + 20m^2 - 16m - 64)c}{m^3(m + 1)^2(m + 2)^4} \mu^{-m} + \frac{3m^4 + 4b}{m^3(m + 1)^2(m + 2)^4} \right] A_1 A_2 + \left[ \frac{2(2m^3 + 17m^2 + 57m + 86m^2 - 136m^2 - 160m - 32)c}{m^1(m + 1)^2(m + 2)^5} \mu^{-m} + \frac{2(2m^3 + 3b)}{m^1(m + 1)^2(m + 2)^5} \right] A_3 A_1 + \left[ \frac{2(2m^3 + 16m^2 + 72m^2 + 116m^2 - 240m - 192)c}{m^3(m + 1)^2(m + 2)^5} \mu^{-m} + \frac{2b}{m^3(m + 1)^2(m + 2)^5} \right] A_3^2 + \left[ \frac{4(2m^3 + 8m^2 + 24m^2 + 12m^2 - 32m - 32)c}{m^3(m + 1)^2(m + 2)^5(m + 4)} \mu^{-m} + \frac{4b}{m^3(m + 1)^2(m + 2)^5(m + 4)} \right] A_4.
\]

We now expand \( f'(\eta_n) \),
\[
f'(\eta_n) = \frac{f^{(m)}(\alpha)}{(m - 1)!} \tilde{\eta}^{m - 1} \Delta[1 + D_1 \tilde{\eta} + D_2 \tilde{\eta}^2 + D_3 \tilde{\eta}^3 + D_4 \tilde{\eta}^4 + O(\tilde{\eta}^5)],
\]
(24)

where
\[
\tilde{\eta} = (\tilde{d}_0 + \tilde{d}_1 \tilde{e}_n + \tilde{d}_2 \tilde{e}_n^2 + \tilde{d}_3 \tilde{e}_n^3 + \tilde{d}_4 \tilde{e}_n^4 + O(\tilde{e}_n^5))^{m - 1}
\]
\[
= \tilde{d}_0^{m - 1} + (m - 1)\tilde{d}_0^{m - 2} \tilde{e}_n + (C_{m - 1}^{3} \tilde{d}_0^{m - 3} + (m - 1)\tilde{d}_2 \tilde{d}_0^{m - 2}) \tilde{e}_n^2 + (2C_{m - 1}^{3} \tilde{d}_2 \tilde{d}_0^{m - 2} + (C_{m - 1}^{3} \tilde{d}_2 \tilde{d}_0^{m - 3} + (m - 1)\tilde{d}_3 \tilde{d}_2 \tilde{d}_0^{m - 2}) \tilde{e}_n^3 + (m - 1)\tilde{d}_4 \tilde{d}_2 \tilde{d}_0^{m - 2} + (C_{m - 1}^{3} \tilde{d}_4 \tilde{d}_2 \tilde{d}_0^{m - 2} + 3C_{m - 1}^{3} \tilde{d}_4 \tilde{d}_2 \tilde{d}_0^{m - 3} + 3C_{m - 1}^{3} \tilde{d}_4 \tilde{d}_2 \tilde{d}_0^{m - 4}) \tilde{e}_n^4 + O(\tilde{e}_n^5).
\]
(25)

Now substituting (14), (15), (20) and (24) into (6) and expand the quotient \( \frac{f(x)}{a f'(x_n) + a f'(\eta_n) + a f'(\eta_0)} \) in Taylor series, we get
\[
e_{n+1} = e_n - \frac{f(x_n)}{a f'(x_n) + a f'(\eta_n) + a f'(\eta_0)} = K_1 e_n + K_2 e_n^2 + K_3 e_n^3 + K_4 e_n^4 + O(e_n^5).
\]
(26)

where the coefficient \( K_1 \) depends on the parameters \( b, c, a_1, a_2, a_3 \) and the function \( f(x) \). If we choose
\[
b = a = \frac{2m}{m + 2}.
\]
(27)

we have
\[
K_1 = 1 + \frac{c - mx}{a_2 m(m + 2)x^2 + a_1 m(m + 2)x y - a_2 c(m + 2)x + a_1 m(m - c)^2},
\]
\[
K_2 = \frac{a_2 m^2 (2m^2 + 2m - 4)x^3 - (2m a_2 (m^2 + 2m - 4)c - m^2 a_3 (m^2 + 2m - 4)y - a_1 m^4)x^2}{m^2 (m + 1)(a_2 m(m + 2)x^2 + a_1 m(m + 2)x y - a_2 c(m + 2)x + a_1 m(m - c)^2)} + \frac{(a_2 m^2 (2m^2 + 2m - 4)c^2 + a_2 m^2 (2m^2 + 2m - 4)y c - 2a_1 m^4)c x + (a_3 m^2 (m + 1)y + a_1 m^2)c^2}{m^2 (m + 1)(a_2 m(m + 2)x^2 + a_1 m(m + 2)x y - a_2 c(m + 2)x + a_1 m(m - c)^2)},
\]
where
\[ x = \left( \frac{m}{m+2} \right)^m \]
\[ y = \left( \frac{m - x^{-1}c}{m+2} \right)^m. \]  

(28)

Before we list \( K_3 \), we choose \( a_1 \) and \( a_3 \) to annihilate the coefficients \( K_1 \) and \( K_2 \),
\[ a_1 = -\frac{2a_2(m^2 - 4) - m^2 - 2m + 4}{-4mx^2 + (2m^2 - 4m + 4)cx + m(m+1)c^2}x^2 \]
\[ -\frac{ma_2(m+1)(m+2)c^2x - (m^2 - 6m + 4)cx - m(m+1)c^2}{m(-4mx^2 + (2m^2 - 4m + 4)cx + m(m+1)c^2)}, \]  
\[ a_3 = \frac{4a_2m^2x^3 - m(m^2 + 8a_2c)x^2 + (2m^2 + 4a_2c)cx - mc^2}{my(-4mx^2 + (2m^2 - 4m + 4)cx + m(m+1)c^2)}, \]  

(29)

(30)

where \( x, y \) are given by (28).

In this case, we have
\[ K_3 = \psi(a_2, c)A_1 + \psi(a_2, c)A_2, \]  

(31)

where
\[ \psi(a_2, c) = \frac{(( -8m^5 - 40m^4 - 80m^3 + 64m)a_2 + 8m^4 + 32m^2)cx^3}{m^2(-4mx^2 + 2(m^2 - 2m + 2)cx + m(m+1)c^2)(c - mx)(m+1)^2(m+2)} \]
\[ -\frac{-4m^3 + 24m^3 - 4m^2 - 64 + 160m)a_2 + (-48m^2 + 2m^6 + 6m^5 + 28m^4 + 64mc^2)x^2}{(8m^4 + 32m^3 - 64m^2 - 32 - 8m^2)a_2 + (m^6 - m^5 - 4m^4 - 4m^3 - 40m^2 + 48m - 32)c^2x} \]
\[ +\frac{m^2(-4mx^2 + 2(m^2 - 2m + 2)cx + m(m+1)c^2)(c - mx)(m+1)^2(m+2)}{2m^2(m^3 + 3m^2 - 2m - 4)c^3}x \]
\[ \psi(a_2, c) = -\frac{-8ma_2x^3 + (2m^2 - 4(m - 2)a_2c)x^2 + (m^2 - 2m + 4a_2c)cx - mc^2}{m(-4mx^2 + (2m^2 - 4m + 4)cx^2 + m(m+1)c^2x)(m+1)(m+2)}c, \]  

(32)

(33)

and \( A_1 \) and \( A_2 \) are defined in (11).

Let \( \psi(a_2, c) = 0 \) and \( \psi(a_2, c) = 0 \), we can get
\[ a_2 = \frac{1}{8} \left( \frac{m}{m+2} \right)^m \left( m(m+8) \right), \]  

(34)

\[ c = -2x = -2 \left( \frac{m}{m+2} \right)^m. \]  

(35)

Substituting (34) and (35) into \( a_1 \) and \( a_3 \), we can get
\[ a_1 = -\frac{1}{16} \frac{3m^4 + 16m^3 + 40m^2 - 176}{m(m+8)}, \]  

(36)

\[ a_3 = \frac{1}{16} \frac{m^4 + 6m^3 + 8m^2 - 16m^2 - 48m - 32}{m^2(m+8)}. \]  

(37)

Substituting (34)–(37), (22) and (27) into (6), we will get a fourth-order method for finding multiple roots of nonlinear equations (denoted by M1).
\[ \begin{align*}
 x_{n+1} &= x_n - \frac{f(x_n)}{2m} \\
y_n &= x_n - \frac{af'(x_n) + af'(y_n) + af'(\eta_n)}{m+2} u_n \\
\eta_n &= x_n - \frac{2m m}{m+2} u_n + 2 \left( \frac{m}{m+2} \right)^m v_n,
\end{align*} \]  

(38)
where $a_1$ is given by (36), $a_2$ is given by (34) and $a_3$ is given by (37). The error equation for $M_1$ is
\[ e_{n+1} = K_4 e_n^4 + O(e_n^5), \]  
where
\[ K_4 = \frac{1}{6} \left( 2m^8 + 25m^7 + 103m^6 + 234m^5 + 320m^4 + 328m^3 - 624m^2 + 64m - 128 \right) A_1 \]
\[ + \frac{m^4 + 7m^2 - 2m^2 + 4m + 8}{m^4(m + 2)(m + 1)^2(m + 8)} A_1 A_2 + \frac{m}{(m + 1)(m + 2)^3(m + 3)} A_3. \]

If we choose $b = 0$
\[ K_1 = 1 + \frac{(m + 2)x - c}{(m - m(m + 2)x)A_1 - A_2 (m^2 + 4m + 4)x^2 + (a_2 c - a_3 y)(m + 2)x}, \]
\[ K_2 = - \frac{a_2 (m^2 + 2m - 4)(m + 2)x^2 + (a_2 c - a_3 y)(m + 2)x^2 - 2a_2 (m + 2)(m^2 + 2m - 4)c x^2}{m^2(m + 1)(a_1 m c - m + 2)((m + 2)a_2 c x^2 + m(a_1 + a_3 y) - a_2 c)x^2 - (m^2 + 2m - 4)a_2 c x^2 - m a_3 (3m^2 + 10m - 4)c x - 2m^2 a_1 (m + 2) c x + a_3 m c (m + 1) y c^2 + a_1 m c^2}, \]
where
\[ x = \left( \frac{m}{m + 2} \right)^m, \quad y = \left( \frac{m + 2 - x^{-1} c}{m + 2} \right)^m. \]

Let $K_1 = 0$ and $K_2 = 0$, then we can get
\[ a_1 = \frac{-1}{4} \left( 4m a_2 y + m^2 + 2m - 4(m + 2)x^2 + 2(m^2 a_3 (m + 4)y + m^2 + 2m - 4)(m + 2)c x \right) \]
\[ + \frac{1}{4} \frac{a_2 m^2 (m + 1)(m + 2) y + m^2 + 2m - 4}{m (m + 2)c x}, \]
\[ a_2 = \frac{m}{4} \left( (m + 2)^2 x^2 - 2(m^2 + 4m - 2)a_3 y + m + 2) c x + (1 + m a_3 (m + 1) y) c^2 \right), \]
where $x, y$ are defined in (42).

Substituting $a_1$ and $a_2$ into $K_3$, we get
\[ K_3 = \psi(a_3, c) A_1^2 + \psi(a_3, c) A_2, \]
where
\[ \psi(a_3, c) = \frac{-2(m - 2)(m + 2)^4 c x^3 + 2(a_3 (m^2 + 7m^4 + 12m^3 - 12m^2 - 48m + 16)y + 3(m - 2)(m + 2)^2)(m + 2)c x}{(c - mx - 2x) (c + mx - 2x)(m + 1) (m + 2)^2} \]
\[ - \frac{m a_3 (m^6 + 13m^4 + 42m^2 - 120m - 32)y + 6(m - 2)(m + 1)^2}{(c - mx - 2x) (m + 1)^2 (m + 2)} \]
\[ + \frac{2a_2 m (m + 1)(m + 3m^2 - 4m - 4)c^3 y + (2m^2 - 8)c^3}{(c - mx - 2x) (m + 1)^2 (m + 2)^2}. \]
\[ \psi(a_3, c) = \frac{(c - 2x) y c}{x(c - mx - 2x)(m + 1)(m + 2)}. \]
and $A_1$ and $A_2$ are defined in (11) and $x, y$ are defined in (42).

Let $\psi(a_3, c) = 0$ and $\psi(a_3, c) = 0$, we can get
\[ a_3 = \frac{1}{16} \left( \frac{m}{m + 2} \right)^m \frac{m^3 (m^2 - 4)}{(m^3 + 2m^2 - 8m + 4)}, \]
\[ c = 2 \left( \frac{m}{m + 2} \right)^m. \]
Substituting (48) and (49) into (43) and (44), we can get
\[
a_1 = \frac{1}{8} \frac{m^6 - m^5 - 14m^4 + 12m^3 + 48m^2 - 80m + 32}{m(m + 2)^2 2m^2 - 8m + 4},
\]
(50)
\[
a_2 = -\frac{m}{16} \frac{3m^4 - 6m^3 - 20m^2 + 40m - 16}{(m + 2)^2 (m^3 + 2m^2 - 8m + 4)}.
\]
(51)
Substituting (48)–(51), (22) and (41) into (6), we will get another fourth-order method (denoted by M2),
\[
\begin{align*}
x_{n+1} &= x_n - a_1 f'(x_n) + a_2 f''(y_n) + a_3 f''(\eta_n), \\
y_n &= x_n - \frac{m + 2}{m} u_n, \\
\eta_n &= x_n - 2 \left(\frac{m}{m + 2}\right)^m v_n,
\end{align*}
\]
(52)
where \(a_1, a_2\) and \(a_3\) are given by (50), (51) and (48) respectively. The error equation for M2 is
\[
e_{n+1} = K_4 e_n^4 + O(e_n^5),
\]
(53)
where
\[
K_4 = \frac{1}{6} \frac{2m^9 + 9m^8 + 9m^7 - 28m^6 - 124m^5 + 328m^4 + 144m^3 - 1248m^2 + 1280m - 384}{m(m + 1)(m + 2)(m^3 + 2m^2 - 8m + 4)} A_1^2 \\
- \frac{m^3 + m^2 - 6m^4}{m(m + 2)(m + 1)^2(m^3 + 2m^2 - 8m + 4)} A_1 A_2 + \frac{m}{(m + 1)(m + 2)^3(m + 3)} A_3.
\]
(54)
Now, we consider the method (8) proposed in [13]. There are eight parameters in (8). It is difficult to discuss this method directly with eight parameters. By considering the following method (dropping \(b_1\) and \(b_2\)) which is simpler, we get two fourth-order methods which require one-function and three-derivative evaluation per iteration, i.e.
\[
\begin{align*}
x_{n+1} &= x_n - a_1 u_n - a_2 v_n - a_3 \frac{f(x_n)}{f'\eta_n),} \\
y_n &= x_n - au_n, \\
\eta_n &= x_n - bu_n - cv_n.
\end{align*}
\]
(55)
There are six parameters \(a, b, c, a_1, a_2\) and \(a_3\) to be determined. The computing process is the same as that of getting the fourth-order methods (38) and (52) before. Substituting (14), (15), (20) and (24) into the error equation
\[
e_{n+1} = e_n - a_1 u_n - a_2 v_n - a_3 \frac{f(x_n)}{f'\eta_n),}.
\]
(56)
and expand the quotients in Taylor series, we get
\[
e_{n+1} = K_1 e_n + K_2 A_1 e_n^2 + (K_3 A_1)^2 A_1 e_n^3 + (K_4 A_1 + K_2 A_1 + K_3 A_1 + K_2 A_1 + K_4 A_1) e_n^4 + O(e_n^5),
\]
(57)
where the coefficient \(K_i\) depends on the parameters \(a, b, c, a_1, a_2\) and \(a_3\). Let the coefficients of \(e_n, e_n^2, e_n^3\) be zero, i.e. \(K_1 = 0, K_2 = 0, K_3 = 0\) and \(K_4 = 0\). Then, we can determine the parameters and get a fourth-order method.
If \(a = b\) given by (27), we can get a fourth-order method based on (58) (denoted by M3),
\[
\begin{align*}
x_{n+1} &= x_n - a_1 u_n - a_2 v_n - a_3 \frac{f(x_n)}{f'\eta_n),} \\
y_n &= x_n - \frac{2m}{m + 2} u_n, \\
\eta_n &= x_n - \frac{2m}{m + 2} u_n + 2 \left(\frac{m}{m + 2}\right)^m v_n,
\end{align*}
\]
(58)
where
\[
a_1 = \frac{m}{8} \frac{m^4 + 4m^3 - 8m + 48}{m^2 + 2m + 6},
\]
(59)
\[
a_2 = \frac{1}{4} \frac{\left(\frac{m}{m + 2}\right)^m m(m^3 + 12m^2 + 36m + 32)}{m^2 + 2m + 6},
\]
(60)
\[
a_3 = -\frac{1}{8} \frac{m^2 (m^3 + 6m^2 + 12m + 8)}{m^2 + 2m + 6},
\]
(61)
and which satisfies the error equation
\[ e_{n+1} = K_4 e_n^4 + O(e_n^5), \tag{62} \]
where
\[ K_4 = \frac{1}{3} \left( m^8 + 10m^7 + 51m^6 + 175m^5 + 412m^4 + 612m^3 + 488m^2 + 112m - 160 \right) A_1^3 \]
\[ - \frac{m^3 + 2m^2 + 8m + 4}{m^3(m + 1)^3(m + 2)^3(m^2 + 2m + 6)} A_1 A_2 + \frac{m}{(m + 1)(m + 2)^3(m + 3)} A_3. \]

If \( a \) is given by (22) and \( b \) is given by (41), we can get another fourth-order method based on (55) (denoted by M4),
\[
\begin{aligned}
  x_{n+1} &= x_n - a_1 u_n - a_2 v_n - a_3 f(x_n), \\
y_n &= x_n - \frac{2m}{m + 2} u_n, \\
\eta_n &= x_n - 2 \left( \frac{m}{m + 2} \right)^m v_n,
\end{aligned}
\]
where
\[
\begin{aligned}
a_1 &= -\frac{1}{4} \frac{m(2m^4 - m^3 - 12m^2 + 20m - 8)}{m^2 - 4m + 2}, \\
a_2 &= \frac{1}{8} \left( \frac{m}{m + 2} \right)^m m(5m^4 + 10m^3 - 16m^2 - 24m + 16), \\
a_3 &= -\frac{1}{8} \frac{m^3(m + 2)^2}{m^2 - 4m + 2} \left( \frac{m}{m + 2} \right)^m.
\end{aligned}
\]

and which satisfies the error equation
\[ e_{n+1} = K_4 e_n^4 + O(e_n^5), \tag{67} \]
where
\[ K_4 = \frac{1}{3} \left( m^8 + 2m^7 - 5m^6 - 15m^5 + 14m^4 - 100m^3 + 88m^2 + 32m - 32 \right) A_1^3 \]
\[ - \frac{m^3 - 2m^2 - 4m + 4}{m(m + 1)^3(m + 2)^2(m^2 - 4m + 2)} A_1 A_2 + \frac{m}{(m + 1)(m + 2)^3(m + 3)} A_3. \]

Furthermore, by considering the following iteration function having \( c = a_1 = a_2 = 0 \) in (8), we get two fourth-order methods which require one-function and two-derivative evaluation per iteration, i.e.
\[
\begin{aligned}
x_{n+1} &= x_n - a_3 f(x_n) \frac{f(x_n)}{b_1 f'(x_n) + b_2 f''(y_n)}, \\
y_n &= x_n - a_3 u_n, \\
\eta_n &= x_n - b_1 u_n.
\end{aligned}
\tag{68} \]

There are five parameters \( a, b, a_3, b_1, b_2 \). For simplicity, we let either \( b = a \) or \( b = 0 \). Then, \( f'(\eta_n) \) reduces to \( f'(y_n) \) and \( f'(x_n) \), respectively.

When \( b = a \), we can get a fourth-order method based on (69) (denoted by M5),
\[
\begin{aligned}
x_{n+1} &= x_n - a_3 f(x_n) \frac{f(x_n)}{b_1 f'(x_n) + b_2 f''(y_n)}, \\
y_n &= x_n - \frac{2m}{m + 2} u_n,
\end{aligned}
\tag{69} \]
where
\[
\begin{aligned}
a_3 &= -\frac{1}{2} \left( \frac{m}{m + 2} \right)^m \frac{m(4m^3 + 16m^2 - 16m + 16)}{m^3 - 4m + 8}, \\
b_1 &= -\frac{(m^3 - 4m + 8)^2}{m(m^4 + 4m^3 - 4m^2 - 16m + 16)(m^2 + 2m - 4)}, \\
b_2 &= \left( \frac{m}{m + 2} \right)^m \frac{m^2(m^3 - 4m + 8)}{(m^4 + 4m^3 - 4m^2 - 16m + 16)(m^2 + 2m - 4)}.
\end{aligned}
\]
and \( a = b \) given by (27). This method satisfies the error equation

\[
e_{n+1} = K_4 e_n^4 + O(e_n^5), \tag{73}
\]

where

\[
K_4 = \frac{1}{3} \frac{m^6 + 6m^5 + 10m^4 - 2m^3 - 24m^2 + 8m - 32}{m^4(m + 1)(m + 2)(m + 2m - 4)} A_1^3 - \frac{A_1 A_2}{m(m + 2)(m + 1)^2} + \frac{m}{(m + 1)(m + 2)^3(m + 3)} A_3. \tag{74}
\]

When \( b = 0 \), we can get another fourth-order method based on (75) (denoted by M6),

\[
\begin{aligned}
x_{n+1} &= x_n - a_3 f(x_n) - \frac{f(x_n)}{b f'(x_n) + b_2 f''(y_n)}, \\
y_n &= x_n - \frac{2m}{m + 2} u_n,
\end{aligned} \tag{75}
\]

where

\[
a = \frac{2m}{m + 2}, \quad a_3 = -\frac{1}{2} m^2 + m, \quad b_1 = -\frac{1}{m}, \quad b_2 = \frac{1}{m(m + 2)^2}. \tag{76}
\]

This method satisfies the error equation

\[
e_{n+1} = K_4 e_n^4 + O(e_n^5), \tag{77}
\]

where

\[
K_4 = \frac{1}{3} \frac{m^6 + 2m^5 + 2m^2 - 2m - 2}{m^4(m + 1)^3} A_1^3 - \frac{A_1 A_2}{m(m + 2)(m + 1)^2} + \frac{m}{(m + 1)(m + 2)^3(m + 3)} A_3. \tag{75}
\]

After some computations, it is easy to see that (75) can reduce to the method (10) proposed in [15]. So, (M6) and (10) are equivalent.

In this section, we have proposed six fourth-order methods with closed formulae. The first four methods denoted (M1)–(M4) require one-function and three-derivative evaluation per iteration. The last two methods, denoted (M5) and (M6), require one-function and two-derivative evaluation per iteration. (M6) is the equivalent to the method (10). We have tried to get a fifth-order method based on (8). But, the equations we get are very complex and we could not yet find such methods.

3. Numerical example

In this section, we employ the presented six methods (38), (52), (58), (63), (69) and (75) to solve some nonlinear equations and compare them with the modified Newton’s method (1) (NM), Chebyshev’s method (2) (CM), Halley’s method (4) (HM) and Osada’s method (5) (OM). All computations were done using the MAPLE using 1024 digit floating point arithmetics (Digits := 1024). We use the following functions, which have also been considered in [9,10], respectively.

\[
\begin{aligned}
f(x) &= x^3 + 4x^2 - 10, \\
f_1(x) &= (x^3 + 4x^2 - 10)^3, \\
f_2(x) &= (\sin x - x^2 + 1)^2, \\
f_3(x) &= (x^3 - e^x - 3x + 2)^5, \\
f_4(x) &= (\cos x - x)^3, \\
f_5(x) &= (e^x - 3x + 2)^5, \\
f_6(x) &= (e^x + x - 30 - 1)^4, \\
f_7(x) &= (e^x + x - 20)^2.
\end{aligned}
\]

Table 1 shows the number of iterations required such that \(|f(x)| < 10^{-60}\) and the number of function evaluations (NFEs) after required iterations in parentheses. From the results displayed in Table 1, we can see that, for the functions we tested, the fourth-order methods proposed in this paper can be competitive to the known third-order methods and Newton’s method and usually converge faster and require less function evaluations. Methods (69) and (75) require three function evaluations per iteration, which is the same as the third-order methods compared, while (69) and (75) require less iterations to converge to the root \( \alpha \).

Acknowledgment

The second author’s work was partially supported by the National Natural Science Foundation of China (60573027 and 60621003).
Table 1
The number of iterations and NFEs.

<table>
<thead>
<tr>
<th>f(x)</th>
<th>x0</th>
<th>NM</th>
<th>CM</th>
<th>HM</th>
<th>OM</th>
<th>M1</th>
<th>M2</th>
<th>M3</th>
<th>M4</th>
<th>M5</th>
<th>M6</th>
</tr>
</thead>
<tbody>
<tr>
<td>f₁</td>
<td>−0.4</td>
<td>167(334)</td>
<td>7(21)</td>
<td>92(276)</td>
<td>12(36)</td>
<td>5(20)</td>
<td>4(16)</td>
<td>3(12)</td>
<td>9(36)</td>
<td>9(27)</td>
<td>7(21)</td>
</tr>
<tr>
<td>f₂</td>
<td>3.0</td>
<td>7(14)</td>
<td>5(15)</td>
<td>4(12)</td>
<td>5(15)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>3(12)</td>
<td>4(12)</td>
<td>4(12)</td>
</tr>
<tr>
<td>f₃</td>
<td>3.5</td>
<td>7(14)</td>
<td>5(15)</td>
<td>5(15)</td>
<td>5(15)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>4(12)</td>
<td>4(12)</td>
</tr>
<tr>
<td>f₄</td>
<td>1.8</td>
<td>5(10)</td>
<td>3(9)</td>
<td>4(12)</td>
<td>3(9)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(9)</td>
<td>3(9)</td>
</tr>
<tr>
<td>f₆</td>
<td>1.7</td>
<td>5(10)</td>
<td>4(12)</td>
<td>4(12)</td>
<td>4(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(9)</td>
<td>3(9)</td>
</tr>
<tr>
<td>f₇</td>
<td>2.2</td>
<td>5(10)</td>
<td>4(12)</td>
<td>4(12)</td>
<td>4(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(9)</td>
<td>3(9)</td>
</tr>
<tr>
<td>f₈</td>
<td>−3.5</td>
<td>17(34)</td>
<td>11(33)</td>
<td>9(27)</td>
<td>12(36)</td>
<td>10(40)</td>
<td>8(32)</td>
<td>9(36)</td>
<td>12(48)</td>
<td>9(27)</td>
<td>9(27)</td>
</tr>
<tr>
<td>f₉</td>
<td>−2.5</td>
<td>11(22)</td>
<td>7(21)</td>
<td>6(18)</td>
<td>8(24)</td>
<td>6(24)</td>
<td>5(20)</td>
<td>6(24)</td>
<td>7(28)</td>
<td>6(18)</td>
<td>6(18)</td>
</tr>
<tr>
<td>f₁₀</td>
<td>3.2</td>
<td>8(16)</td>
<td>5(15)</td>
<td>4(12)</td>
<td>5(15)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>5(20)</td>
<td>4(12)</td>
<td>4(12)</td>
</tr>
<tr>
<td>f₁₁</td>
<td>4.2</td>
<td>22(44)</td>
<td>15(45)</td>
<td>12(36)</td>
<td>16(48)</td>
<td>13(52)</td>
<td>11(44)</td>
<td>12(48)</td>
<td>16(64)</td>
<td>12(36)</td>
<td>11(33)</td>
</tr>
<tr>
<td>f₁₂</td>
<td>3.0</td>
<td>5(10)</td>
<td>4(12)</td>
<td>3(9)</td>
<td>4(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(12)</td>
<td>3(9)</td>
<td>3(9)</td>
<td>3(9)</td>
</tr>
<tr>
<td>f₁₃</td>
<td>5.5</td>
<td>9(18)</td>
<td>6(18)</td>
<td>5(15)</td>
<td>6(18)</td>
<td>4(16)</td>
<td>4(16)</td>
<td>5(20)</td>
<td>4(16)</td>
<td>4(12)</td>
<td>4(12)</td>
</tr>
</tbody>
</table>

References