



A class of computationally efficient numerical algorithms for locating multiple zeros

Janak Raj Sharma¹ · Sunil Kumar¹

Received: 29 September 2020 / Accepted: 21 December 2020 / Published online: 18 January 2021
© African Mathematical Union and Springer-Verlag GmbH Deutschland, ein Teil von Springer Nature 2021

Abstract

In recent times, many iterative methods for computing multiple zeros of nonlinear functions have been appeared in literature. Different from these existing methods, here we propose a new class of methods with eighth order convergence for multiple zeros. With four evaluations per iteration, the methods satisfy the criterion of attaining optimal convergence of eighth order. Applicability is demonstrated on different examples that illustrates the computational efficiency of novel methods. Comparison of numerical results shows that the proposed techniques possess good convergence compared to existing optimal order techniques. Besides, the accuracy of existing techniques is also challenged which is the main advantage.

Keywords Nonlinear equations · Newton-like methods · Fast algorithm · Multiple roots · Convergence

Mathematics Subject Classification 65H05 · 41A25 · 49M15

1 Introduction

Constructing higher order multi-point numerical methods for the multiple zeros of the univariate function $f(t)$, where $f : \mathbb{C} \rightarrow \mathbb{C}$ is analytic in neighborhood of the required zero, is one of the most important and difficult problems in numerical analysis. The advantages of multi-point iterative methods over the one-point iterative method can be found in the excellent book by Traub [19]. More recently, Argyros and Regmi have also highlighted the newly optimized advantages of multi-point iterative methods in their excellent book (see [1]). The most basic one-point method is the Newton's method [13]

$$t_{i+1} = t_i - \mu \frac{f(t_i)}{f'(t_i)}, \quad i = 0, 1, 2, \dots \quad (1)$$

✉ Janak Raj Sharma
jrshira@yahoo.co.in

Sunil Kumar
sfageria1988@gmail.com

¹ Department of Mathematics, Sant Longowal Institute of Engineering and Technology, Longowal, Sangrur 148106, India

Here μ is the multiplicity of a zero (say, α) of the function $f(t)$, that is, $f^{(j)}(\alpha) = 0$, $j = 0, 1, \dots, \mu - 1$ and $f^{(\mu)}(\alpha) \neq 0$.

Many higher order multi-point methods, either independent or dependent on the Newton's scheme (1), have been proposed in literature, see [2,5,7,9–11,14–18,22] and the references cited therein. In particular, Liu and Zhou [11] have recently proposed the following scheme of two-point Newton-like methods:

$$\begin{aligned} y_i &= t_i - \mu \frac{f(t_i)}{f'(t_i)}, \\ t_{i+1} &= y_i - \mu G(u_i) \frac{f(t_i)}{f'(t_i)}, \end{aligned} \quad (2)$$

where $u_i = \left(\frac{f'(y_i)}{f'(t_i)} \right)^{\frac{1}{\mu-1}}$ and $G : \mathbb{C} \rightarrow \mathbb{C}$ is a holomorphic function in the neighborhood of origin zero. They have shown that this iterative scheme attains fourth order convergence provided that the function $G(u)$ satisfies the conditions:

$$G(0) = 0, \quad G'(0) = 1, \quad G''(0) = \frac{4\mu}{\mu - 1}, \quad \mu \neq 1.$$

In this work our aim is to develop multi-point iterations with high computational efficient, i.e. the iterative methods of higher convergence order that may use the computations as small in number as we please. The definition of computational efficiency is closely related to the well-known conjecture by Kung and Traub [8]. According to this the multi-point methods without memory requiring n functional evaluations can attain the maximum convergence order $2^n - 1$. The methods qualifying the Kung–Traub criterion are usually known as optimal methods. It is clear that Liu–Zhou method is optimal with fourth order convergence. Due to the complexity in developing iterative procedures, the optimal methods for computing multiple roots are seldom obtained.

Keeping the above points in view, we propose a family of eighth order methods that requires four new pieces of information per iteration and hence possesses optimal convergence of eighth order in the sense of Kung–Traub hypothesis. Proposed iterative scheme is the composition of three steps that uses the Liu–Zhou iteration (2) in the first two steps and Newton-type iteration in the third step. Iterative scheme is unique in the sense that it requires two functions and two derivatives per each iteration. The efficiency index (see [12]) is 1.682 which is better than the efficiency index 1.587 of the basic fourth order Liu–Zhou method. Therefore, the new scheme can also be observed as the modification of Liu–Zhou scheme.

We summarize the contents of this article. In Sect. 2, the eighth order iterative technique is developed and its convergence is studied. Some numerical tests are performed in Sect. 3 to check the stability of the methods and to verify the theoretical results. A comparison with the existing methods is also shown in this section. In Sect. 4 concluding remarks are reported.

2 Development of scheme

Authors of this research area have used a variety of techniques to develop higher order iterative methods for solving nonlinear equations. Some of these are: interpolation approach, sampling approach, composition approach, geometrical approach, adomian approach and weight-function approach. Of these the weight-function approach has been most popular in recent times, see, for example [5,11,14,22] and references cited therein. We also use this technique in the present work of computing a multiple root with multiplicity $\mu > 1$. Thereby consider the following three-step iterative scheme:

$$\begin{aligned}
 y_i &= t_i - \mu \frac{f(t_i)}{f'(t_i)}, \\
 z_i &= y_i - \mu G(u_i) \frac{f(t_i)}{f'(t_i)}, \\
 t_{i+1} &= z_i - \mu v_i B(u_i) H(v_i) K(w_i) \frac{f(t_i)}{f'(t_i)},
 \end{aligned} \tag{3}$$

where $u_i = \left(\frac{f'(y_i)}{f'(t_i)}\right)^{\frac{1}{\mu-1}}$, $v_i = \left(\frac{f(z_i)}{f'(t_i)}\right)^{\frac{1}{\mu}}$, $w_i = \frac{v_i}{u_i}$, and the functions $G, B, H, K : \mathbb{C} \rightarrow \mathbb{C}$ are analytic in a neighborhood of 0. Note that second and third steps are weighted by the factors G, B, H and K , so these factors are called weight factors or weight functions. Note that u_i and v_i are one-to- $\mu - 1$ and one-to- μ multi-valued functions, respectively, so we consider their principal analytic branches. Hence, it is convenient to treat them as the principal root.

In the sequel we will explore certain conditions under which the scheme (3) achieves convergence of order as high as possible. The lengthy calculations are handled by using computer algebra system such as Mathematica software (see [21]). In order to study the convergence following theorem is stated and proved:

Theorem 1 Let $f : \mathbb{C} \rightarrow \mathbb{C}$ be an analytic function in a region enclosing a multiple zero (say, α) with multiplicity $\mu > 1$. Assume that initial guess t_0 is sufficiently close to α , then the local order of convergence of scheme (3) is at least 8, provided that

$$\begin{aligned}
 G(0) &= 0, \quad G'(0) = 1, \quad G''(0) = \frac{4\mu}{\mu - 1}, \\
 G'''(0) &= \frac{6(\mu(\mu^3 - 3\mu^2 + \mu + 1)B'''(0) + 2(6\mu^4 + \mu^3 - 5\mu^2 - 3\mu - 3)B''(0))}{(\mu - 1)^2(\mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0))}, \\
 B(0) &= \frac{(\mu - 1)(\mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0))}{4(9\mu^3 - 8\mu^2 - 5\mu + 6)}, \\
 B'(0) &= 2B(0), \quad H(0) = \frac{\mu^2 H'(0)}{2(\mu^2 - \mu - 1)}, \quad K(0) = \frac{1}{B(0)H(0)}, \quad K'(0) = \frac{\mu - 1}{\mu} K(0), \\
 H'(0) &\neq 0, \quad \mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0) \neq 0.
 \end{aligned}$$

Proof Let the error at i -th iteration be $\epsilon_i = t_i - \alpha$. Developing $f(t_i)$ and $f'(t_i)$ about α by Taylor’s series

$$f(t_i) = \frac{f^{(\mu)}(\alpha)}{\mu!} \epsilon_i^\mu \left(1 + C_1 \epsilon_i + C_2 \epsilon_i^2 + C_3 \epsilon_i^3 + C_4 \epsilon_i^4 + C_5 \epsilon_i^5 + C_6 \epsilon_i^6 + \dots \right) \tag{4}$$

and

$$\begin{aligned}
 f'(t_i) &= \frac{f^{(\mu)}(\alpha)}{\mu!} \epsilon_i^{\mu-1} \left(\mu + (\mu + 1)C_1 \epsilon_i + (\mu + 2)C_2 \epsilon_i^2 + (\mu + 3)C_3 \epsilon_i^3 \right. \\
 &\quad \left. + (\mu + 4)C_4 \epsilon_i^4 + \dots \right),
 \end{aligned} \tag{5}$$

where $C_n = \frac{\mu!}{(\mu+n)!} \frac{f^{(\mu+n)}(\alpha)}{f^{(\mu)}(\alpha)}$ for $n \in \mathbb{N}$.

Using (4) and (5) in first step of (3), it follows that

$$\epsilon_{y_i} = y_i - \alpha = \frac{C_1}{\mu} \epsilon_i^2 + \frac{2\mu C_2 - (\mu + 1)C_1^2}{\mu^2} \epsilon_i^3 + \sum_{n=1}^5 \phi_n \epsilon_i^{n+3} + O(\epsilon_i^9). \tag{6}$$

where $\phi_n = \phi_n(\mu, C_1, C_2, C_3, \dots, C_7)$, $n = 1, 2, 3, 5$. For brevity, expressions of ϕ_n being very lengthy are not expressed explicitly. Such lengthy expressions obtained in the next computations will also not be shown explicitly.

Expansion of $f'(y_i)$ about α leads us to the expression

$$f'(y_i) = \frac{f^{(\mu)}(\alpha)}{\mu!} e^{y_i \mu - 1} \left(\mu + (\mu + 1)C_1 \epsilon_{y_i} + (\mu + 2)C_2 \epsilon_{y_i}^2 + (\mu + 3)C_3 \epsilon_{y_i}^3 + (\mu + 4)C_4 \epsilon_{y_i}^4 + \dots \right). \tag{7}$$

Using (5) and (7) in $u_i = \left(\frac{f'(y_i)}{f'(t_i)} \right)^{\frac{1}{\mu-1}}$, we have that

$$u_i = \frac{C_1}{\mu} \epsilon_i + \frac{2(\mu - 1)C_2 - (\mu + 1)C_1^2}{\mu(\mu - 1)} \epsilon_i^2 + \sum_{n=1}^6 \eta_n \epsilon_i^{n+2} + O(\epsilon_i^9), \tag{8}$$

where $\eta_n = \eta_n(\mu, C_1, C_2, \dots, C_8)$.

Expansion of weight function $G(u_i)$ in the neighborhood of origin yields

$$G(u_i) \approx G(0) + u_i G'(0) + \frac{1}{2} u_i^2 G''(0) + \frac{1}{6} u_i^3 G'''(0) + O(u_i^4). \tag{9}$$

Then the second step of scheme (3), on using Eqs. (4)–(6), (8) and (9), produces

$$\epsilon_{z_i} = z_i - \alpha = -G(0)\epsilon_i + \frac{1 + G(0) - G'(0)}{\mu} C_1 \epsilon_i^2 + \sum_{n=1}^6 \gamma_n \epsilon_i^{n+2} + O(\epsilon_i^9), \tag{10}$$

where $\gamma_n = \gamma_n(\mu, G(0), G'(0), G''(0), G'''(0), C_1, C_2, \dots, C_7)$.

It follows that the fourth order convergence can be achieved if the coefficients of ϵ_i, ϵ_i^2 and ϵ_i^3 vanish. Then, resulting equations yield

$$G(0) = 0, \quad G'(0) = 1 \quad \text{and} \quad G''(0) = \frac{4\mu}{\mu - 1}. \tag{11}$$

By using (11) in (10), we obtain that

$$\begin{aligned} \epsilon_{z_i} = & \frac{C_1}{6\mu^3(\mu - 1)^2} \left((3(\mu^3 + 8\mu^2 + \mu + 2) - G'''(0)(\mu - 1)^2) C_1^2 - 6(\mu - 1)\mu^2 C_2 \right) \epsilon_i^4 \\ & + \sum_{n=1}^4 \varphi_n \epsilon_i^{n+4} + O(\epsilon_i^9), \end{aligned} \tag{12}$$

where $\varphi_n = \varphi_n(\mu, G'''(0), C_1, C_2, \dots, C_6)$.

Developing $f(z_i)$ about α , then

$$f(z_i) = \frac{f^{(\mu)}(\alpha)}{\mu!} \epsilon_{z_i}^\mu \left(1 + C_1 \epsilon_{z_i} + C_2 \epsilon_{z_i}^2 + C_3 \epsilon_{z_i}^3 + C_4 \epsilon_{z_i}^4 + C_5 \epsilon_{z_i}^5 + C_6 \epsilon_{z_i}^6 + \dots \right). \tag{13}$$

From (4) and (13), we get expression of $v_i = \left(\frac{f(z_i)}{f(t_i)} \right)^{\frac{1}{\mu}}$ as

$$\begin{aligned} v_i = & \frac{C_1}{6\mu^3(\mu - 1)^2} \left((3(\mu^3 + 8\mu^2 + \mu + 2) - G'''(0)(\mu - 1)^2) C_1^2 - 6(\mu - 1)\mu^2 C_2 \right) \epsilon_i^3 \\ & + \sum_{n=1}^5 \tau_n \epsilon_i^{n+3} + O(\epsilon_i^9), \end{aligned} \tag{14}$$

where $\tau_n = \tau_n(\mu, G'''(0), C_1, C_2, \dots, C_6)$.

The use of (8) and (14) in $w_i = \frac{v_i}{u_i}$ gives

$$w_i = \frac{1}{6\mu^2(\mu - 1)^2} \left((3(\mu^3 + 8\mu^2 + \mu + 2) - G'''(0)(\mu - 1)^2)C_1^2 - 6(\mu - 1)\mu^2C_2 \right) \epsilon_i^2 + \sum_{n=1}^6 \psi_n \epsilon_i^{n+2} + O(\epsilon_i^9), \tag{15}$$

where $\psi_n = \psi_n(\mu, G'''(0), C_1, C_2, \dots, C_7)$.

Next, we expand weight functions $B(u_i)$, $H(v_i)$ and $K(w_i)$ in the neighborhood of origin by Taylor series, then we have

$$B(u_i) \approx B(0) + u_i B'(0) + \frac{1}{2} u_i^2 B''(0) + \frac{1}{6} u_i^3 B'''(0), \tag{16}$$

$$H(v_i) \approx H(0) + v_i H'(0), \tag{17}$$

$$K(w_i) \approx K(0) + w_i K'(0). \tag{18}$$

Hence by substituting (4), (5), (8), (12), (14)–(18) into the last step of scheme (3), we obtain the error equation

$$\epsilon_{i+1} = \frac{(B(0)H(0)K(0) - 1)C_1}{6\mu^3(\mu - 1)^2} \left((G'''(0)(\mu - 1)^2 - 3(\mu^3 + 8\mu^2 + \mu + 2))C_1^2 + 6(\mu - 1)\mu^2C_2 \right) \epsilon_i^4 + \sum_{n=1}^4 \xi_n \epsilon_i^{n+4} + O(\epsilon_i^8), \tag{19}$$

where $\xi_n = \xi_n(\mu, G'''(0), B(0), B'(0), B''(0), B'''(0), H(0), H'(0), K(0), K'(0), C_1, C_2, \dots, C_6)$.

To obtain eighth order, it is sufficient to fix coefficients of $\epsilon_i^4, \epsilon_i^5, \epsilon_i^6$ and ϵ_i^7 simultaneously equal to zero. This process will give us

$$\begin{cases} G'''(0) = \frac{6(\mu(\mu^3 - 3\mu^2 + \mu + 1)B'''(0) + 2(6\mu^4 + \mu^3 - 5\mu^2 - 3\mu - 3)B''(0))}{(\mu - 1)^2(\mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0))}, \\ B(0) = \frac{(\mu - 1)(\mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0))}{4(9\mu^3 - 8\mu^2 - 5\mu + 6)}, \\ B'(0) = 2B(0), \quad H(0) = \frac{\mu^2 H'(0)}{2(\mu^2 - \mu - 1)}, \quad K(0) = \frac{1}{B(0)H(0)}, \quad K'(0) = \frac{\mu - 1}{\mu} K(0), \end{cases} \tag{20}$$

wherein $H'(0) \neq 0$ and $\mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0) \neq 0$.

Substituting the values expressed by (20) in the error equation (19), then some simple calculations yield

$$\begin{aligned} \epsilon_{i+1} = & \frac{1}{48(\mu - 1)^5 \mu^8 (\mu(1 - \mu)B'''(0) + 6(\mu^2 - \mu - 1)B''(0))^3} \\ & \times \left(C_1 \left[(-\mu(\mu^3 + 9\mu^2 - 13\mu + 3)B'''(0) + 2(3\mu^4 + 9\mu^3 \right. \right. \\ & - 26\mu^2 - 11\mu - 3)B''(0))C_1^2 + 2(\mu - 1)\mu(\mu - 1)B'''(0) \\ & + (6 + 6\mu - 6\mu^2)B''(0))C_2 \left] [(\mu^2(\mu - 1)^2 B'''(0))^2 \right. \\ & \times (14\mu^7 + 131\mu^6 + 172\mu^5 - 1058\mu^4 + 374\mu^3 + 87\mu^2 + 280\mu + 48) \\ & - 12\mu(14\mu^{10} + 67\mu^9 - 310\mu^8 - 424\mu^7 \\ & \left. \left. + 1557\mu^6 - 469\mu^5 - 268\mu^4 - 558\mu^3 + 63\mu^2 + 280\mu + 48)B''(0)B'''(0) \right] \right) \end{aligned}$$

$$\begin{aligned}
 &+ 12(42\mu^{11} + 93\mu^{10} - 984\mu^9 \\
 &- 315\mu^8 + 3878\mu^7 + 200\mu^6 - 2468\mu^5 - 3143\mu^4 - 372\mu^3 + 1941 \\
 &\times \mu^2 + 1128\mu + 144)B''(0)^2C_1^4 - 12\mu \\
 &\times (\mu - 1)(\mu^2(\mu - 1)^2(4\mu^5 + 21\mu^4 - 36\mu^3 - \mu^2 + 8)B'''(0)^2 - 4\mu(12\mu^8 + 21 \\
 &\times \mu^7 - 182\mu^6 + 185\mu^5 + 53\mu^4 \\
 &- 50\mu^3 - 63\mu^2 + 24)B''(0)B'''(0) + 12(12\mu^9 + 3\mu^8 - 142 \\
 &\times \mu^7 + 130\mu^6 + 152\mu^5 - 46\mu^4 - 142\mu^3 - 51\mu^2 \\
 &+ 48\mu + 24)B''(0)^2C_1^2C_2 + 12(\mu - 1)^2\mu^4(2\mu - 3) \\
 &\times (\mu(\mu - 1)B'''(0) + (6 + 6\mu - 6\mu^2)B''(0))^2C_2^2 \\
 &+ 24(\mu - 1)^3\mu^3(1 + \mu)(\mu(\mu - 1)B'''(0) + (6 + 6\mu - 6 \\
 &\times \mu^2)B''(0))^2C_1C_3 \Big] \epsilon_i^8 + O(\epsilon_i^9). \tag{21}
 \end{aligned}$$

Hence, the eighth order convergence is established. This completes the proof of theorem. \square

2.1 Some concrete methods

Many special cases of the scheme (3) can be generated satisfying the corresponding conditions on the functions G, B, H and K shown above in Theorem 1 . Moreover, we will restrict choices to consider the forms of low order polynomials. These choices should be such that the resulting methods may converge to the root with order eight for $\mu > 1$. Accordingly, the following simple forms are chosen:

$$G(u_i) = u_i + \frac{2\mu}{\mu - 1}u_i^2 + \frac{(23\mu^4 + 7\mu^3 - 21\mu^2 - 13\mu - 12)}{(\mu - 1)^2(13\mu^2 - 13\mu - 12)}u_i^3. \tag{22}$$

$$B(u_i) = -\frac{13\mu^3 - 26\mu^2 + \mu + 12}{4(9\mu^3 - 8\mu^2 - 5\mu + 6)}\left(\frac{1}{2} + u_i\right) - \frac{u_i^2}{2}\left(1 - \frac{1}{6}u_i\right). \tag{23}$$

$$\begin{aligned}
 B(u_i) = &\frac{1}{8(9\mu^3 - 8\mu^2 - 5\mu + 6)(6 + u_i)}\left(-72 - 6\mu + 156\mu^2 - 78\mu^3 \right. \\
 &\left. - (169\mu^3 - 338\mu^2 + 13\mu + 156)u_i - (242\mu^3 - 244\mu^2 - 118\mu + 168)u_i^2\right). \tag{24}
 \end{aligned}$$

$$H(v_i) = \frac{\mu^2}{2(\mu^2 - \mu - 1)} + v_i. \tag{25}$$

$$H(v_i) = \frac{\mu^3v_i + \mu^2(1 + 2v_i) - 2\mu v_i - 2v_i}{2(\mu^2 - \mu - 1)(1 + \mu v_i)}. \tag{26}$$

$$H(v_i) = \frac{\mu^3(2\mu^2v_i - 2\mu v_i - 2v_i + \mu)}{2(\mu^2 - \mu - 1)(2v_i + 2\mu^3v_i + \mu^2(v_i^2 - 4v_i + 1))}. \tag{27}$$

$$K(w_i) = -\frac{16(\mu^2 - \mu - 1)(9\mu^3 - 8\mu^2 - 5\mu + 6)}{\mu^2(13\mu^2 - 13\mu - 12)}\left(\frac{1}{\mu - 1} + \frac{1}{\mu}w_i\right). \tag{28}$$

$$K(w_i) = -\frac{16(\mu^2 - \mu - 1)(9\mu^3 - 8\mu^2 - 5\mu + 6)(\mu - w_i + \mu w_i + \mu^2w_i)}{(\mu - 1)\mu^3(13\mu^2 - 13\mu - 12)(1 + \mu w_i)}. \tag{29}$$

Let us select the combination of (22), (23), (25), (28) in the scheme (3) and denote the resulting method by M-1; the combination of (22), (23), (26), (28) and denote the corresponding method by M-2; the combination of (22), (24), (25), (28) and denote the method by M-3; and the combination of (22), (23), (27), (28) and denote the method by M-4.

Remark 1 The computational efficiency (E) is defined as $E = p^{1/\theta}$, where p is the order of convergence of the considered method and θ is the number of function evaluations required per iteration (see [12]). With the conditions (11) and (20) the proposed scheme (3) reaches at eighth order convergence by using only four functional evaluations (viz. $f(t_i)$, $f(z_i)$, $f'(t_i)$ and $f'(y_i)$) per iteration. Thus, the E -value of the new scheme is $8^{1/4} \approx 1.682$, which is much better than the E -values of Newton’s method ($E = 2^{1/2} \approx 1.414$) and fourth order Liu–Zhou method ($E = 4^{1/3} \approx 1.587$).

Remark 2 The proposed algorithms require the knowledge of multiplicity μ of a root. To estimate μ , we can employ the formula

$$\mu \approx \frac{t_{i+1} - t_i}{F(t_{i+1}) - F(t_i)},$$

wherein $F(t_i) = \frac{f(t_i)}{f'(t_i)}$, which is approximately the reciprocal of divided difference of F for successive iterates t_i and t_{i+1} (see [6]).

3 Numerical examples

Convergence behavior and computational efficiency of the new methods M-1, M-2, M-3 and M-4 are hereby demonstrated by applying on some numerical problems. Performance is compared with some existing well-known methods. For example, we choose the optimal fourth order methods by Kansal et al. [5], Li et al. [9,10], Liu and Zhou [11], Sharma and Sharma [17], Soleymani et al. [18] and Zhou et al. [22]. These methods are expressed as follows:

Li–Liao–Cheng method (LLC):

$$y_i = t_i - \frac{2\mu}{\mu + 2} \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = t_i - \frac{\mu(\mu - 2) \left(\frac{\mu}{\mu+2}\right)^{-\mu} f'(y_i) - \mu^2 f'(t_i)}{f'(t_i) - \left(\frac{\mu}{\mu+2}\right)^{-\mu} f'(y_i)} \frac{f(t_i)}{2f'(t_i)}.$$

Li–Cheng–Neta method (LCN):

$$y_i = t_i - \frac{2\mu}{\mu + 2} \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = t_i - a_1 \frac{f(t_i)}{f'(y_i)} - \frac{f(t_i)}{a_2 f'(t_i) + a_3 f'(y_i)},$$

where

$$a_1 = -\frac{1}{2} \frac{\left(\frac{\mu}{\mu+2}\right)^\mu \mu(\mu^4 + 4\mu^3 - 16\mu - 16)}{\mu^3 - 4\mu + 8},$$

$$a_2 = -\frac{(\mu^3 - 4\mu + 8)^2}{\mu(\mu^4 + 4\mu^3 - 4\mu^2 - 16\mu + 16)(\mu^2 + 2\mu - 4)},$$

$$a_3 = \frac{\mu^2(\mu^3 - 4\mu + 8)}{\left(\frac{\mu}{\mu+2}\right)^\mu (\mu^4 + 4\mu^3 - 4\mu^2 - 16\mu + 16)(\mu^2 + 2\mu - 4)}.$$

Sharma–Sharma method (SS):

$$y_i = t_i - \frac{2\mu}{\mu + 2} \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = t_i - \frac{\mu}{8} \left[(\mu^3 - 4\mu + 8) - (\mu + 2)^2 \left(\frac{\mu}{\mu + 2}\right)^\mu \frac{f'(t_i)}{f'(y_i)} \right. \\ \left. \times \left(2(\mu - 1) - (\mu + 2) \left(\frac{\mu}{\mu + 2}\right)^\mu \frac{f'(t_i)}{f'(y_i)} \right) \right] \frac{f(t_i)}{f'(t_i)}.$$

Zhou–Chen–Song method (ZCS):

$$y_i = t_i - \frac{2\mu}{\mu + 2} \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = t_i - \frac{\mu}{8} \left[\mu^3 \left(\frac{\mu + 2}{\mu}\right)^{2\mu} \left(\frac{f'(y_i)}{f'(t_i)}\right)^2 - 2\mu^2(\mu + 3) \left(\frac{\mu + 2}{\mu}\right)^\mu \frac{f'(y_i)}{f'(t_i)} \right. \\ \left. + (\mu^3 + 6\mu^2 + 8\mu + 8) \right] \frac{f(t_i)}{f'(t_i)}.$$

Soleymani–Babajee–Lotfi method (SBL):

$$y_i = t_i - \frac{2\mu}{\mu + 2} \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = t_i - \frac{f'(y_i) f(t_i)}{p_1 (f'(y_i))^2 + p_2 f'(y_i) f'(t_i) + p_3 (f'(t_i))^2},$$

where

$$p_1 = \frac{1}{16} \mu^{3-\mu} (2 + \mu)^\mu,$$

$$p_2 = \frac{8 - \mu(2 + \mu)(\mu^2 - 2)}{8\mu},$$

$$p_3 = \frac{1}{16} (\mu - 2) \mu^{\mu-1} (2 + \mu)^{3-\mu}.$$

Kansal–Kanwar–Bhatia method (KKB):

$$y_i = t_i - \frac{2\mu}{\mu + 2} \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = t_i - \frac{\mu}{4} f(t_i) \left(1 + \frac{\mu^4 q^{-2\mu} \left(q^{\mu-1} - \frac{f'(y_i)}{f'(t_i)} \right)^2 (q^\mu - 1)}{8(2q^\mu + m(q^\mu - 1))} \right) \\ \times \left(\frac{4 - 2\mu + \mu^2 (q^{-\mu} - 1)}{f'(t_i)} - \frac{q^{-\mu} (2q^\mu + \mu(q^\mu - 1))^2}{f'(t_i) - f'(y_i)} \right),$$

Table 1 Test functions

Functions	Root (α)	μ	Initial guess
$f_1(t) = t^4 + 11.50t^3 + 47.49t^2 + 83.06325t + 51.23266875$ (see [3])	-2.85	2	-2.5
$f_2(t) = (t^3 - 5.22t^2 + 9.0825t - 5.2675)^2$	1.75	4	3.7
$f_3(t) = [\tan^{-1}(\frac{\sqrt{5}}{2}) - \tan^{-1}(\sqrt{t^2 - 1}) + \sqrt{6}(\tan^{-1}(\sqrt{\frac{t^2 - 1}{6}}) - \tan^{-1}(\frac{1}{2}\sqrt{\frac{5}}{6})) - \frac{11}{63}]^5$ (see [4])	1.841...	5	1.5
$f_4(t) = (t - 2)((t - 1)^3 - 1)^9 \sin(t - 2)(e^{t-2} - 1)$	2	12	3

where $q = \frac{\mu}{\mu+2}$.

Liu-Zhou method (LZ):

$$y_i = t_i - \mu \frac{f(t_i)}{f'(t_i)},$$

$$t_{i+1} = y_i - \mu \left(u_i + \frac{2\mu}{\mu - 1} u_i^2 \right) \frac{f(t_i)}{f'(t_i)},$$

where $u_i = \left(\frac{f'(y_i)}{f'(t_i)} \right)^{\frac{1}{\mu-1}}$.

The methods are tested on the various problems as shown in Table 1. Numerical calculations are performed in the programming package of Mathematica software using multiple-precision arithmetic. Computed results exhibited in Table 2 contain the following numerical values:

- The number of iterations (i) required to obtain the desired solution.
- The estimated errors $|t_{i+1} - t_i|$ for the first three iterations.
- The computational order of convergence (COC).
- The total number of function evaluations (TNFE).

Necessary iterations (i) are calculated by using the condition $|t_{i+1} - t_i| + |f(t_i)| < 10^{-350}$ as the stopping criterion. Computational order of convergence is calculated by the formula

$$COC = \frac{\ln |(t_{i+2} - \alpha)/(t_{i+1} - \alpha)|}{\ln |(t_{i+1} - \alpha)/(t_i - \alpha)|},$$

which is used to validate the theoretical order of convergence (see [20]).

From the numerical results displayed in Table 2 we observe that the errors generated by the proposed methods M-1, M-2, M-3 and M-4 show the greater accuracy in the successive approximations. This justifies the good convergence of the methods. The value 0 for $|t_{i+1} - t_i|$ indicates that the required accuracy has been achieved. Computational order of convergence (COC) shown in the penultimate column of the table overwhelmingly supports the theoretical convergence order. This feature points to the uniformity in the convergence speed of the iterations which is contrary to the belief that higher order iterations do not always preserve the order of convergence. Computational efficiency of the methods can be viewed by observing the entries of TNFE. Indeed, the new methods are efficient in general, since TNFE is less than that of the existing ones in all the cases. Similar numerical testing, carried out for many other problems, have confirmed the above conclusions to a large extent.

Table 2 Comparison of numerical results

Methods	i	$ t_2 - t_1 $	$ t_3 - t_2 $	$ t_4 - t_3 $	COC	TNFE
$f_1(t)$						
LLC	5	2.17×10^{-3}	2.43×10^{-13}	3.96×10^{-53}	4	15
LCN	5	2.17×10^{-3}	2.43×10^{-13}	3.96×10^{-53}	4	15
SS	5	2.20×10^{-3}	2.56×10^{-13}	4.91×10^{-53}	4	15
ZCS	5	2.30×10^{-3}	3.04×10^{-13}	9.72×10^{-53}	4	15
SBL	5	2.17×10^{-3}	2.43×10^{-13}	3.96×10^{-53}	4	15
KKB	5	2.05×10^{-3}	1.94×10^{-13}	1.62×10^{-53}	4	15
LZ	5	5.40×10^{-3}	8.75×10^{-12}	6.82×10^{-47}	4	15
M-1	3	2.34×10^{-6}	2.94×10^{-49}	0	8	12
M-2	3	2.35×10^{-6}	2.91×10^{-49}	0	8	12
M-3	3	2.36×10^{-6}	3.08×10^{-49}	0	8	12
M-4	3	2.34×10^{-6}	2.90×10^{-49}	0	8	12
$f_2(t)$						
LLC	8	2.56×10^{-1}	3.61×10^{-2}	1.75×10^{-3}	4	24
LCN	8	2.56×10^{-1}	3.63×10^{-2}	1.78×10^{-3}	4	24
SS	8	2.57×10^{-1}	3.67×10^{-2}	1.84×10^{-3}	4	24
ZCS	8	2.59×10^{-1}	3.74×10^{-2}	1.95×10^{-3}	4	24
SBL	8	2.61×10^{-1}	3.84×10^{-2}	2.11×10^{-3}	4	24
KKB	7	2.52×10^{-1}	3.48×10^{-2}	1.55×10^{-3}	4	21
LZ	8	2.90×10^{-1}	5.00×10^{-2}	4.73×10^{-3}	4	24
M-1	5	1.30×10^{-1}	4.03×10^{-3}	4.94×10^{-10}	8	20
M-2	5	1.34×10^{-1}	4.42×10^{-3}	1.02×10^{-9}	8	20
M-3	5	1.30×10^{-1}	4.03×10^{-3}	4.91×10^{-10}	8	20
M-4	5	1.34×10^{-1}	4.51×10^{-3}	1.19×10^{-9}	8	20
$f_3(t)$						
LLC	5	1.28×10^{-3}	2.92×10^{-14}	8.09×10^{-57}	4	15
LCN	5	1.28×10^{-3}	2.91×10^{-14}	7.99×10^{-57}	4	15
SS	5	1.28×10^{-3}	2.89×10^{-14}	7.74×10^{-57}	4	15
ZCS	5	1.28×10^{-3}	2.87×10^{-14}	7.41×10^{-57}	4	15
SBL	5	1.28×10^{-3}	2.78×10^{-14}	6.38×10^{-57}	4	15
KKB	5	1.28×10^{-3}	3.01×10^{-14}	9.40×10^{-57}	4	15
LZ	5	3.33×10^{-5}	2.43×10^{-20}	6.86×10^{-81}	4	15
M-1	3	9.07×10^{-9}	3.59×10^{-69}	0	8	12
M-2	3	9.07×10^{-9}	3.60×10^{-69}	0	8	12
M-3	3	9.07×10^{-9}	3.58×10^{-69}	0	8	12
M-4	3	9.07×10^{-9}	3.60×10^{-69}	0	8	12
$f_4(t)$						
LLC	6	1.10×10^{-1}	1.46×10^{-4}	6.53×10^{-16}	4	18
LCN	6	1.10×10^{-1}	1.46×10^{-4}	6.65×10^{-16}	4	18

Table 2 continued

Methods	i	$ t_2 - t_1 $	$ t_3 - t_2 $	$ t_4 - t_3 $	COC	TNFE
SS	6	1.11×10^{-1}	1.50×10^{-4}	7.40×10^{-16}	4	18
ZCS	6	1.11×10^{-1}	1.51×10^{-4}	7.72×10^{-16}	4	18
SBL	6	1.41×10^{-1}	4.59×10^{-4}	7.98×10^{-14}	4	18
KKB	6	1.06×10^{-1}	1.19×10^{-4}	2.80×10^{-16}	4	18
LZ	6	1.39×10^{-1}	5.81×10^{-4}	3.19×10^{-13}	4	18
M-1	4	1.97×10^{-2}	4.03×10^{-14}	1.01×10^{-107}	8	12
M-2	4	2.49×10^{-2}	2.78×10^{-13}	5.11×10^{-101}	8	12
M-3	4	1.97×10^{-2}	3.91×10^{-14}	7.66×10^{-108}	8	12
M-4	4	2.65×10^{-2}	4.78×10^{-13}	3.93×10^{-99}	8	12

4 Conclusions

A convergent three-step optimal eighth order scheme has been derived for locating multiple zeros of nonlinear functions. The methodology is based on Liu–Zhou optimal fourth order and Newton-like iterations. Analysis of the convergence has proved the order eight under standard assumptions. Performance has been checked by numerical testing. The theoretical eighth order convergence is verified by calculating computational order of convergence (COC). The comparison of performance of the methods with existing efficient methods has also been shown. Comparison of computational efficiency, measured in terms of total number of function evaluations (TNFE) required to achieve the desired solution with the specified accuracy, has also confirmed the efficient and robust character of the new methods. Finally, it is hoped that this study makes a significant contribution to solving nonlinear equations.

References

1. Argyros, I.K., Regmi, S.: Undergraduate Research at Cameron University on Iterative Procedures in Banach and Other Spaces. Nova Science Publisher, New York (2019)
2. Dong, C.: A family of multi-point iterative functions for finding multiple roots of equations. *Int. J. Comput. Math.* **21**, 363–367 (1987)
3. Douglas, J.M.: *Process Dynamics and Control*. Prentice Hall, Englewood Cliffs (1972)
4. Hoffman, J.D.: *Numerical Methods for Engineers and Scientists*. McGraw-Hill Book Company, New York (1992)
5. Kansal, M., Kanwar, V., Bhatia, S.: On some optimal multiple root-finding methods and their dynamics. *Appl. Appl. Math.* **10**, 349–367 (2015)
6. King, R.F.: A secant method for multiple roots. *BIT* **17**, 321–328 (1977)
7. Kumar, S., Kumar, D., Sharma, J.R., Cesarano, C., Agarwal, P., Chu, Y.M.: An optimal fourth order derivative-free numerical algorithm for multiple roots. *Symmetry* **12**, 1038 (2020). <https://doi.org/10.3390/sym12061038>
8. Kung, H.T., Traub, J.F.: Optimal order of one-point and multipoint iteration. *J. Assoc. Comput. Mach.* **21**, 643–651 (1974)
9. Li, S.G., Cheng, L.Z., Neta, B.: Some fourth-order nonlinear solvers with closed formulae for multiple roots. *Comput. Math. Appl.* **59**, 126–135 (2010)
10. Li, S., Liao, X., Cheng, L.: A new fourth-order iterative method for finding multiple roots of nonlinear equations. *Appl. Math. Comput.* **215**, 1288–1292 (2009)
11. Liu, B., Zhou, X.: A new family of fourth-order methods for multiple roots of nonlinear equations. *Non. Anal. Model. Cont.* **18**, 143–152 (2013)

12. Ostrowski, A.M.: Solutions of Equations and System of Equations. Academic Press, New York (1960)
13. Schröder, E.: Über unendlich viele Algorithmen zur Auflösung der Gleichungen. *Math. Ann.* **2**, 317–365 (1870)
14. Sharifi, M., Babajee, D.K.R., Soleymani, F.: Finding the solution of nonlinear equations by a class of optimal methods. *Comput. Math. Appl.* **63**, 764–774 (2012)
15. Sharma, J.R., Kumar, D., Cattani, C.: An efficient class of weighted-Newton multiple root solvers with seventh order convergence. *Symmetry* **11**, 1054 (2019). <https://doi.org/10.3390/sym11081054>
16. Sharma, J.R., Kumar, S., Jäntschi, L.: On derivative free multiple-root finders with optimal fourth order convergence. *Mathematics* **8**, 1091 (2020). <https://doi.org/10.3390/math8071091>
17. Sharma, J.R., Sharma, R.: Modified Jarratt method for computing multiple roots. *Appl. Math. Comput.* **217**, 878–881 (2010)
18. Soleymani, F., Babajee, D.K.R., Lotfi, T.: On a numerical technique for finding multiple zeros and its dynamics. *J. Egypt. Math. Soc.* **21**, 346–353 (2013)
19. Traub, J.F.: *Iterative Methods for the Solution of Equations*. Chelsea Publishing Company, New York (1982)
20. Weerakoon, S., Fernando, T.G.I.: A variant of Newton's method with accelerated third-order convergence. *Appl. Math. Lett.* **13**, 87–93 (2000)
21. Wolfram, S.: *Wolfram Mathematica*, 12th edn. Wolfram Research, Champaign (2020)
22. Zhou, X., Chen, X., Song, Y.: Constructing higher-order methods for obtaining the multiple roots of nonlinear equations. *J. Comput. Math. Appl.* **235**, 4199–4206 (2011)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.