Further development on Traub's method for solving system of nonlinear equations and ODE's

Kalyanasundaram Madhu (✉ kalyan742@gmail.com)

Khalifa University of Science Technology and Research: Khalifa University of Science and Technology

Mo'tassem Al-arydah

Khalifa University of Science Technology and Research: Khalifa University of Science and Technology

Research article

Keywords: Kung-Traub conjecture, Multi-point iterations, Non-linear equation, Optimal order, Basins of attraction

Posted Date: September 9th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-70859/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Further development on Traub's method for solving system of nonlinear equations and ODE's

Kalyanasundaram Madhu ¹ Mo'tassem Al-arydah ^{*}, ¹

¹ *Department of Mathematics, Khalifa University, P.O.Box: 127788, Abu Dhabi, UAE*
email: kalyanasundaram.mathu@ku.ac.ae, motassem.alarydah@ku.ac.ae

^{*} Corresponding author.

Abstract

The foremost objective of this work is to propose a eighth and sixteenth order scheme for handling a nonlinear equation. The eighth order method uses three evaluations of the function and one assessment of the first derivative and sixteenth order method uses four evaluations of the function and one appraisal of the first derivative. Kung-Traub conjecture is satisfied, theoretical analysis of the methods are presented and numerical examples are added to confirm the order of convergence. The performance and efficiency of our iteration methods are compared with the equivalent existing methods on some standard academic problems. We tested projectile motion problem, Planck's radiation law problem as an application. The basins of attraction are also given to demonstrate their dynamical behavior in the complex plane.

Further, we attempt to proposed a sixteenth order iterative method for solving system of nonlinear equation with four functional evaluation, namely two F and two F' and only one inverse of Jacobian. The theoretical proof of the method is given and numerical examples are included to confirm the convergence order of the presented methods. We apply the new scheme to find solution on 1-D bratu problem. The performance and efficiency of our iteration methods are compared.

AMS Subject Classification: 65H05 · 65D05 · 41A25

Key Words: Kung-Traub conjecture, Multi-point iterations, Non-linear equation, Optimal order, Basins of attraction

1 Introduction

Tackling nonlinear equations could be a common and critical issue in science and engineering [9]. The boundary value problems in dynamic hypothesis of gasses, flexibility and other connected regions are generally diminished to solving single variable nonlinear equation. Thus, the issue of approximating a arrangement of the nonlinear equation is vital. Iterative strategies are one among the numerical strategies for finding the roots of such equations. Analytic strategies for fathoming such conditions are nearly nonexistent and consequently to get approximate solution by numerical strategies is based on iterative methods. With the progression of computers, the issue of solving nonlinear condition by numerical strategies has picked up more significance than some time recently. Famous Mathematicians who have contributed for the solution of equations are Cauchy, Chebyshev, Euler, Fourier, Gauss, Lagrange, Laguerre and Newton [43]. Here, we consider the problem of locating simple zeros and its denoted by x^* of a equation $f(x) = 0$, where $f(x)$ is a sufficiently continuously differentiable function. The Newton-Raphson method (NM) is the most widely used algorithm for locating or finding simple zeros, which works with an initial points at x_0 near to the approximate root and obtaining a sequence of successive iterates $\{x_n\}_0^\infty$ converging quadratically to simple roots. It is is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, n = 0, 1, 2, 3... \quad (1)$$

In the recent years many researchers are working with this problems in order to improve the convergence order and efficiency of NM method, in terms of additional functional evaluations, derivatives, and addition

step [38]. Traub [42] proposed a two step variant of Newton's method (*TM*) having convergence order three by evaluating two functions, one derivative is given by

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \quad x_{n+1} = y_n - \frac{f(y_n)}{f'(x_n)}. \quad (2)$$

The huge survey of the literature dealing with these methods of improved order and efficiency are in [37] and references therein. Ezquerro et al. [21], Halley [23], Ostrowski's square root method [37] are well known cubic order iterative methods which needs the three functional evaluation of f , f' and f'' per iteration. These methods are not an optimal method in the sense of Kung and Traub [30]. The conjectured says that the order of convergence of any multi-point without memory method with d function evaluations cannot exceed the bound 2^{d-1} , the optimal order. Thus the optimal order for three evaluations per iteration would be four, four evaluations per iteration would be eight, and so on.

To obtain optimality case, researchers are developed and analysed further, some examples of optimal fourth order multi-point methods without memory which requires three evaluations per iteration are Ostrowski's method [37], King's family of methods [28], which contains Ostrowski's method as a special case. Methods of Chun et al. [11, 12], Cordero et al. [16], Kou et al. [29], Jarratt [25, 26] are seen to be efficient when compared to classical *NM* method. Some examples of optimal eighth order method without memory which requires four function evaluations per iteration are Liu et al [31] called as *LWM*, Sharma et al [40] called as *SAM*, Cordero et al [13] called as *CFGT*, Cordero et al [19] called as *CTV*, and Neta et al [35] called as *NCS* respectively given below

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, w_n = y_n - \frac{f(x_n)}{f(x_n)-2f(y_n)} \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \frac{f(w_n)}{f'(x_n)} \left(\left(\frac{f(x_n)-f(y_n)}{f(x_n)-2f(y_n)} \right)^2 + \frac{f(w_n)}{f(y_n)-f(w_n)} + \frac{4f(w_n)}{f(x_n)+f(w_n)} \right). \end{cases} \quad (3)$$

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, w_n = y_n - \left(3 - 2 \frac{f[y_n, x_n]}{f'(x_n)} \right) \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \frac{f(w_n)}{f'(x_n)} \left(\frac{f'(x_n)-f[y_n, x_n]+f[w_n, y_n]}{2f[w_n, y_n]-f[w_n, x_n]} \right). \end{cases} \quad (4)$$

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, w_n = y_n - \frac{f(y_n)}{f'(x_n)} \frac{1}{1-2t+t^2-t^3/2}, \\ x_{n+1} = w_n - \frac{1+3r}{1+r} \frac{f(w_n)}{f[w_n, y_n]+f[w_n, x_n](w_n-y_n)}, t = \frac{f(y_n)}{f(x_n)}, r = \frac{f(w_n)}{f(x_n)}. \end{cases} \quad (5)$$

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, w_n = x_n - \frac{1-t}{1-2t} \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \left(\frac{1-t}{1-2t} - v \right)^2 \frac{1}{1-3v} \frac{f(w_n)}{f'(x_n)}, v = \frac{f(w_n)}{f(y_n)}. \end{cases} \quad (6)$$

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, w_n = y_n - \frac{1}{(1-t)^2} \frac{f(y_n)}{f'(x_n)}, x_{n+1} = w_n - \frac{f(w_n)}{H'_3(w_n)}, \\ H'_3(w_n) = 2(f[x_n, w_n] - f[x_n, y_n]) + f[y_n, w_n] + \frac{y_n-w_n}{y_n-x_n} (f[x_n, y_n] - f'(x_n)) \end{cases} \quad (7)$$

In this work, we contribute a little more in the theory of iterative methods by developing an optimal formula of order four from third order iterative method (2) with weight function for computing simple roots of a nonlinear equation which uses three function evaluations. Also, we develop a class of optimal eighth order methods from proposed fourth order method by using finite difference techniques. On the other hand, we analyze the behavior of eighth order method in the complex plane. Several authors have used these techniques on different iterative methods, viz Curry et al. [20] and Vrscay and Gilbert [44, 45] described the dynamical behavior of some

well-known iterative methods. The complex analysis of various other known iterative methods, such as King's and Chebyshev-Halley's families, Jarratt method have also been analyzed by various researchers, example, see [2, 3, 10, 18, 33, 22]. Further, we develop an optimal sixteenth order method from proposed a member of optimal eighth order method with five functional evaluation by using finite difference techniques.

This paper is organized as follows. In section 2, a class of optimal eighth-order and sixteenth-order and its proof of convergence are stated and proved for scalar equations. We compare the presented methods with some previously available eighth order methods on some test functions in section 3. In section 4, the proposed eighth order methods are studied in the complex plane using basins of attraction. Some real world problems are discussed in section 5 where new eighth and sixteenth order methods are applied on this problem. In section 6, we further developed a sixteenth order method for solving system of nonlinear equation and and its proof of convergence are stated. Also, we tested the performance of the proposed method with some academic problems. Section 7 gives concluding remarks.

2 Development of methods and its Convergence

Traub's method (2) having convergence order is three with three function evaluations per full iteration and having $EI = 1.442$ and it is not an optimal method. In this section, we propose new eighth and sixteenth order iterative method without memory namely three and fourth-step methods respectively. To improve the convergence order and efficiency of the Traub's method with three function evaluations, we used a weight function in the second step. To get an optimal eighth and sixteenth order iterative method, we used a divided difference technique in the third and fourth step. First, we trying to get an optimal fourth-order method in the following way

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)}, w_n = y_n - \frac{1}{H(\tau)} \frac{f(y_n)}{f'(x_n)}, \tau = \frac{f(y_n)}{f(x_n)}. \quad (8)$$

Next, we stated the convergence proof for fourth order, which can easily proved with help of MATHEMATICA.

Theorem 2.1. *For sufficiently smooth function $f : D \subset \mathbb{R} \rightarrow \mathbb{R}$ having a simple root x^* in the open interval D , then the family of method (8) is of fourth order convergence, when*

$$H(0) = 1, H'(0) = -2, |H''(0)| < \infty \quad (9)$$

and it satisfies the error equation:

$$e_{n+1} = d_0 e_n^4 + O(e_n^5), \quad d_0 = (1 + H''(0))c_2^3 - c_2 c_3, \quad (10)$$

$$c_j = \frac{f^{(j)}(x^*)}{j!f'(x^*)}, \quad j = 2, 3, 4, \dots \text{ and } e_n = x_n - x^*.$$

In this section, our main aim is to develop an eighth and sixteenth-order method without memory. Here, we develop the class of optimal eighth order method with help of fourth order method (8) with following expressions

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \quad w_n = y_n - \frac{1}{H(\tau)} \frac{f(y_n)}{f'(x_n)}, \quad x_{n+1} = w_n - \frac{f(w_n)}{f'(w_n)}, \quad (11)$$

where this method have convergence order eight with 5 function evaluations and it is not an optimal. To obtain an optimal scheme, so we estimate $f'(w_n)$ by the following polynomial

$$\chi(t) = a_0 + a_1(t - x) + a_2(t - x)^2 + a_3(t - x)^3, \quad (12)$$

which satisfies these condition

$$\chi(x_n) = f(x_n), \chi'(x_n) = f'(x_n), \chi(y_n) = f(y_n), \chi(w_n) = f(w_n).$$

Let us define the divided differences

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

By using above conditions on equation (12), we get system of four linear equations with four unknowns a_0 , a_1 , a_2 and a_3 . From $\chi(x_n) = f(x_n)$, $\chi'(x_n) = f'(x_n)$, we get $a_0 = f(x_n)$ and $a_1 = f'(x_n)$. To find a_2 and a_3 , we solve the following equations:

$$\begin{aligned} f(y_n) &= f(x_n) + f'(x_n)(y_n - x_n) + a_2(y_n - x_n)^2 + a_3(y_n - x_n)^3 \\ f(w_n) &= f(x_n) + f'(x_n)(w_n - x_n) + a_2(w_n - x_n)^2 + a_3(w_n - x_n)^3. \end{aligned}$$

Solving above equations by using divided difference, we have

$$a_2 = \frac{f[y_n, x_n, x_n](w_n - x_n) - f[w_n, x_n, x_n](y_n - x_n)}{w_n - y_n}, \quad a_3 = \frac{f[w_n, x_n, x_n] - f[y_n, x_n, x_n]}{w_n - y_n}. \quad (13)$$

Further, using eq. (13), we have the estimation

$$f'(w_n) \approx \chi'(w_n) = f'(x_n) + 2a_2(w_n - x_n) + 3a_3(w_n - x_n)^2.$$

Finally, we obtain a new class of optimal eighth order method

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ w_n = y_n - \frac{1}{H(\tau)} \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \frac{f(w_n)}{f'(x_n) + 2a_2(w_n - x_n) + 3a_3(w_n - x_n)^2}, \end{cases} \quad (14)$$

where a_2 and a_3 are given in (13).

Theorem 2.2. *For sufficiently smooth function $f : D \subset \mathbb{R} \rightarrow \mathbb{R}$ having a simple root x^* in the open interval D , then the method (14) is of local eighth order convergence and it satisfies the error equation*

$$e_{n+1} = c_2 d_0 (c_4 + d_0) e_n^8 + O(e_n^9)$$

Proof. Let $\tilde{e}_n = y_n - x^*$, $\hat{e}_n = w_n - x^*$ and $c_j = \frac{f^{(j)}(x^*)}{j!f'(x^*)}$, $j = 2, 3, 4, \dots$. Expansion of $f(x_n)$ and $f'(x_n)$ around x^* by using Taylor's series, we have

$$f(x_n) = f'(x^*) \left(e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 + c_6 e_n^6 + c_7 e_n^7 + c_8 e_n^8 + O(e_n^9) \right) \quad (15)$$

and

$$f'(x_n) = f'(x^*) \left(1 + 2c_2 e_n + 3c_3 e_n^2 + 4c_4 e_n^3 + 5c_5 e_n^4 + 6c_6 e_n^5 + 7c_7 e_n^6 + 8c_8 e_n^7 + 9c_9 e_n^8 + O(e_n^9) \right) \quad (16)$$

Thus,

$$\begin{aligned}
\tilde{e}_n = & c_2 e_n^2 + \left(-2c_2^2 + 2c_3\right) e_n^3 + \left(4c_2^3 - 7c_2 c_3 + 3c_4\right) e^4 + \left(-8c_2^4 \right. \\
& + 20c_2^2 c_3 - 6c_3^2 - 10c_2 c_4 + 4c_5\left.) e^5 + \left(16c_2^5 - 52c_2^3 c_3 + 28c_2^2 c_4 - 17c_3 c_4 \right. \right. \\
& + c_2(33c_3^2 - 13c_5) + 5c_6\left.) e^6 - 2\left(16c_2^6 - 64c_2^4 c_3 - 9c_3^3 + 36c_2^3 c_4 + 6c_4^2 + 9c_2^2(7c_3^2 \right. \right. \\
& - 2c_5) + 11c_3 c_5 + c_2(-46c_3 c_4 + 8c_6) - 3c_7\left.) e^7 + \left(64c_2^7 - 304c_2^5 c_3 \right. \right. \\
& + 176c_2^4 c_4 + 75c_3^2 c_4 + c_2^3(408c_3^2 - 92c_5) - 31c_4 c_5 - 27c_3 c_6 \\
& \left. + c_2^2(-348c_3 c_4 + 44c_6) + c_2(-135c_3^3 + 64c_4^2 + 118c_3 c_5 - 19c_7) + 7c_8\right) e^8 + \dots
\end{aligned} \tag{17}$$

Expanding $f(y_n)$ about x^* by using Taylor's series, we have

$$f(y_n) = f'(x^*) \left(\tilde{e}_n + c_2 \tilde{e}_n^2 + c_3 \tilde{e}_n^3 + c_4 \tilde{e}_n^4 + O(\tilde{e}_n^5) \right) \tag{18}$$

Also, expanding $f(w_n)$ about x^* by using Taylor's series, we have

$$f(w_n) = f'(x^*) \left(\hat{e}_n + c_2 \hat{e}_n^2 + O(\hat{e}_n^3) \right) \tag{19}$$

Substituting equations (33)-(19) in the third step of (14) and simplifying, we obtain

$$e_{n+1} = c_2 d_0 \left(c_4 + d_0 \right) e_n^8 + O(e_n^9).$$

This reveals that the proposed family of methods attains eighth-order convergence. The efficiency of the method (14) is $EI = 1.682$. $\square$

By choice of any value of $H''(0)$ in (9), we getting a new eighth order iterative method. Some members of the class (14) are as follows. Proposed method (VTM1): By choosing $H''(0) = 0$, we obtain as

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ w_n = y_n - \left(\frac{1}{1-2\tau} \right) \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \frac{f(w_n)}{f'(x_n) + 2a_2(w_n - x_n) + 3a_3(w_n - x_n)^2}. \end{cases} \tag{20}$$

This method has the following error equation $e_{n+1} = c_2^2 \left(c_2^2 - c_3 \right) \left(c_2^3 - c_2 c_3 + c_4 \right) e_n^8 + O(e_n^9)$. Proposed method (VTM2): By choosing $H''(0) = -2$, we obtain as

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ w_n = y_n - \left(\frac{1}{1-2\tau-\tau^2} \right) \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \frac{f(w_n)}{f'(x_n) + 2a_2(w_n - x_n) + 3a_3(w_n - x_n)^2}. \end{cases} \tag{21}$$

This method has the following error equation $e_{n+1} = c_2^2 c_3 \left(c_2 c_3 - c_4 \right) e_n^8 + O(e_n^9)$. Proposed method (VTM3): By choosing $H''(0) = -4$, we obtain as

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ w_n = y_n - \left(\frac{1}{1-2\tau-2\tau^2} \right) \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = w_n - \frac{f(w_n)}{f'(x_n) + 2a_2(w_n - x_n) + 3a_3(w_n - x_n)^2}. \end{cases} \tag{22}$$

This method has the following error equation $e_{n+1} = c_2^2(c_2^2 + c_3)(c_2^3 + c_2c_3 - c_4)e_n^8 + O(e_n^9)$.

Note that based on the analysis done on numerical results, we find that *VTM2* is marginally better or equal than other ones, therefore we are considering *VTM2* to develop further to sixteenth order method, namely *VTM4*. In the similar way, we are trying to get a new sixteenth order iterative method (*VTM4*) as following way

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ w_n = y_n - \left(\frac{1}{1-2\tau-\tau^2} \right) \frac{f(y_n)}{f'(y_n)}, \\ z_n = w_n - \frac{f(w_n)}{f'(x_n) + 2a_2(w_n - x_n) + 3a_3(w_n - x_n)^2}, \\ x_{n+1} = z_n - \frac{f(z_n)}{f'(z_n)}. \end{cases} \quad (23)$$

The above method is having convergence order is 16 with six evaluations. However, this is not an optimal method. To get an optimal, we estimate $f'(z_n)$ by the following polynomial

$$\chi(t) = b_0 + b_1(t - x) + b_2(t - x)^2 + b_3(t - x)^3 + b_4(t - x)^4, \quad (24)$$

which satisfies these conditions

$$\chi(x_n) = f(x_n), \chi'(x_n) = f'(x_n), \chi(y_n) = f(y_n), \chi(w_n) = f(w_n), \chi(z_n) = f(z_n).$$

Using the above conditions on equation (24), we obtain system of five linear equations with five unknowns b_0 , b_1 , b_2 and b_3 . From conditions, we get $b_0 = f(x_n)$ and $b_1 = f'(x_n)$. To find b_2 , b_3 and b_4 , we solve the following equations:

$$\begin{aligned} f(y_n) &= f(x_n) + f'(x_n)(y_n - x_n) + b_2(y_n - x_n)^2 + b_3(y_n - x_n)^3 + b_4(y_n - x_n)^4 \\ f(w_n) &= f(x_n) + f'(x_n)(w_n - x_n) + b_2(w_n - x_n)^2 + b_3(w_n - x_n)^3 + b_4(w_n - x_n)^4 \\ f(z_n) &= f(x_n) + f'(x_n)(z_n - x_n) + b_2(z_n - x_n)^2 + b_3(z_n - x_n)^3 + b_4(z_n - x_n)^4. \end{aligned}$$

Thus by using divided differences, the above equations reduced to

$$\begin{aligned} b_2 + b_3(y_n - x_n) + b_4(y_n - x_n)^2 &= f[y_n, x_n, x_n] \\ b_2 + b_3(w_n - x_n) + b_4(w_n - x_n)^2 &= f[w_n, x_n, x_n] \\ b_2 + b_3(z_n - x_n) + b_4(z_n - x_n)^2 &= f[z_n, x_n, x_n] \end{aligned} \quad (25)$$

Solving the equation (25), we have

$$\begin{aligned} b_2 &= \frac{(f[y_n, x_n, x_n](-S_2^2S_3 + S_2S_3^2) + f[w_n, x_n, x_n](S_1^2S_3 - S_1S_3^2) + f[z_n, x_n, x_n](-S_1^2S_2 + S_1S_2^2))}{-S_1^2S_2 + S_1S_2^2 + S_1^2S_3 - S_2^2S_3 - S_1S_3^2 + S_2S_3^2}, \\ b_3 &= \frac{(f[y_n, x_n, x_n](S_2^2 - S_3^2) + f[w_n, x_n, x_n](-S_1^2 + S_3^2) + f[z_n, x_n, x_n](S_1^2 - S_2^2))}{-S_1^2S_2 + S_1S_2^2 + S_1^2S_3 - S_2^2S_3 - S_1S_3^2 + S_2S_3^2}, \\ b_4 &= \frac{(f[y_n, x_n, x_n](-S_2 + S_3) + f[w_n, x_n, x_n](S_1 - S_3) + f[z_n, x_n, x_n](-S_1 + S_2))}{-S_1^2S_2 + S_1S_2^2 + S_1^2S_3 - S_2^2S_3 - S_1S_3^2 + S_2S_3^2}, \\ S_1 &= y_n - x, \quad S_2 = w_n - x, \quad S_3 = z_n - x. \end{aligned} \quad (26)$$

Using equation (26), we have

$$f'(z_n) \approx q'(z_n) = b_1 + 2b_2(z_n - x_n) + 3b_3(z_n - x_n)^2 + 4b_4(z_n - x_n)^3.$$

Finally, we obtain a new optimal sixteenth order method (*VTM4*)

$$x_{n+1} = z_n - \frac{f(z_n)}{f'(x) + 2b_2(z_n - x_n) + 3b_3(z_n - x_n)^2 + 4b_4(z_n - x_n)^3}. \quad (27)$$

where b_2 , b_3 and b_4 are given in (26). The efficiency of the method (27) is $EI = 1.741$.

The following theorem is given without proof, which can be worked out with the help of MATHEMATICA.

Theorem 2.3. *For sufficiently smooth function $f : D \subset \mathbb{R} \rightarrow \mathbb{R}$ having a simple root x^* in the open interval D , then the method (27) is of local sixteenth order convergence and it satisfies the error equation*

$$e_{n+1} = c_2^4 c_3^2 (c_2 c_3 - c_4) (c_2^2 c_3 - c_2 c_4 + c_5) e_n^{16} + O(e_n^{17}).$$

3 Numerical examples

Here, we testing the performance and effectiveness of new methods, classical Newton's method (NM), and these eighth order methods (3)-(7). Let us consider the following test functions to test,

$$\begin{cases} f_1(x) = \sin(2 \cos x) - 1 - x^2 + e^{\sin(x^3)}, & x^* = -0.7848959876612125352... \\ f_2(x) = x^3 + 4x^2 - 10, & x^* = 1.3652300134140968457... \\ f_3(x) = \sin(x) + \cos(x) + x, & x^* = -0.4566247045676308244... \\ f_4(x) = x^2 + \sin(\frac{x}{5}) - \frac{1}{4}, & x^* = 0.4099920179891371316... \end{cases}$$

Numerical results are carried out in the MATLAB software and we have used the following stopping criteria for satisfying the iterative process $error = |x_N - x_{N-1}| < \epsilon$, where $\epsilon = 10^{-50}$ and N is the number of iterations required for convergence. The computational order of convergence is given by ([17])

$$\rho = \frac{\ln |(x_N - x_{N-1}) / (x_{N-1} - x_{N-2})|}{\ln |(x_{N-1} - x_{N-2}) / (x_{N-2} - x_{N-3})|}.$$

From Table 1, we observe that $VTM1$, $VTM2$, $VTM3$, and $VTM4$ converges with lesser number of iterations or with least error than compared methods (3)-(7). We conform that the theoretical order of convergence and computational order of convergence are approximately equal. Note that the initial guess are near to the root, then we will get converge with least iteration and error. Concluding that the proposed method $VTM4$ having good efficiency in all the test function as compared to other methods. Hence, the $VTM4$ can be considered competent enough to existing other compared methods.

Remark 1: We are trying to compare the test function with “ $fzero$ ” command in MATLAB software and the results are given in table 2. Here N_1 is the number of iterations to converge the interval containing the root and $f(x_n)$ is the error after converging N number of iterations. For the $fzero$ command, the zeros are consider to be location where the function actually cuts, not just meet the x-axis. It is observed that the new methods converge with a lesser number of iteration and total function evaluations than the $fzero$ solver. Also, we conclude that the Newton-type iterative methods are better than $fzero$ command.

4 Applications

4.1 Projectile Motion Problem

We consider the classical projectile problem in which a shot is propelled from a tower of tallness $h > 0$, with beginning speed v and at an point θ with regard to the horizontal onto a slope, which is characterized by the function ω , called the affect function which is subordinate on the horizontal distance, x . We wish to discover the ideal dispatch point θ_m which maximizes the even distance. In our calculations, we disregard air resistances. The path function $y = P(x)$ that depicts the movement of the shot is given by

$$P(x) = h + x \tan \theta - \frac{gx^2}{2v^2} \sec^2 \theta \quad (28)$$

I.F.	N	e_1	e_2	e_3	$error$	ρ	CPU_{time}
$f_1, x_0 = -1.0$							
NM	7	0.1967	0.0182	2.3636e-04	9.0696e-61	2.00	1.144847
LWM	4	0.2151	2.1060e-06	1.7919e-45	0	7.82	0.979315
SAM	4	0.2151	1.3780e-06	6.7387e-47	0	7.80	0.989816
$CFGT$	3	0.2151	1.8564e-07	3.3933e-55	3.3933e-55	7.87	0.811550
CTV	4	0.2151	2.1081e-06	7.2218e-46	0	7.90	1.015022
NCS	3	0.2151	2.6854e-07	1.4762e-53	1.4762e-53	7.83	0.777970
$VTM1$	3	0.2151	1.6751e-07	8.8559e-56	8.8559e-56	7.90	0.777329
$VTM2$	3	0.2151	8.7696e-08	1.1049e-58	1.1049e-58	7.96	0.730283
$VTM3$	3	0.2151	3.0044e-08	1.4996e-62	1.4996e-62	7.92	0.743219
$VTM4$	3	0.2151	3.2987e-15	1.8592e-235	1.8592e-235	15.94	0.781413
$f_2, x_0 = 0.9$							
NM	8	0.6263	0.1497	0.0113	1.3514e-72	2.00	1.023982
LWM	4	0.4660	7.3967e-04	3.9688e-27	2.7346e-213	7.99	0.740913
SAM	4	0.4492	0.0160	7.4896e-16	1.5396e-122	7.99	0.731669
$CFGT$	4	0.4654	1.8025e-04	6.6893e-33	2.4091e-260	7.99	0.810392
CTV	4	0.4652	6.6285e-05	2.2445e-36	3.8782e-288	7.99	0.746914
NCS	4	0.4655	2.8120e-04	8.1224e-31	3.9399e-243	7.99	0.764206
$VTM1$	4	0.4653	3.3164e-05	5.5734e-39	3.5460e-309	7.99	0.710881
$VTM2$	4	0.4653	6.9411e-05	2.3267e-37	3.7105e-297	7.99	0.724889
$VTM3$	4	0.4660	7.2943e-04	8.5164e-28	2.9522e-219	7.99	0.723621
$VTM4$	3	0.4652	2.6070e-07	2.5410e-110	2.5410e-110	16.47	0.702237
$f_3, x_0 = -0.9$							
NM	6	0.4415	0.0019	3.5152e-07	1.9715e-59	2.00	0.912788
LWM	3	0.4434	5.7199e-10	1.4223e-78	1.4223e-78	7.71	0.793040
SAM	3	0.4434	1.4115e-08	2.4782e-67	2.4782e-67	7.83	0.784606
$CFGT$	3	0.4434	2.0940e-10	2.5879e-83	2.5879e-83	7.81	0.819606
CTV	3	0.4434	4.2747e-08	3.6452e-63	3.6452e-63	7.84	0.830108
NCS	3	0.4434	8.0225e-11	5.6893e-87	5.6893e-87	7.81	0.791882
$VTM1$	3	0.4434	8.0034e-11	3.5499e-87	3.5499e-87	7.83	0.702888
$VTM2$	3	0.4434	7.9843e-11	1.7686e-87	1.7686e-87	7.86	0.787046
$VTM3$	3	0.4434	7.9653e-11	3.2750e-88	3.2750e-88	7.94	0.759142
$VTM4$	3	0.4434	4.7543e-21	0	0	16.07	0.671253
$f_4, x_0 = 1.5$							
NM	9	0.7194	0.2927	0.0727	2.7867e-74	2.00	1.335429
LWM	4	1.0743	0.0157	3.3096e-14	1.4980e-107	7.99	0.873192
SAM	4	1.0905	4.9099e-04	7.3633e-26	1.8691e-200	8.00	0.879429
$CFGT$	4	1.0899	8.8191e-05	1.4106e-35	5.5201e-282	8.00	0.936900
CTV	4	1.0943	0.0043	2.2678e-19	1.2146e-149	8.00	0.907344
NCS	4	1.0814	0.0086	9.2312e-17	1.8264e-128	7.99	0.902067
$VTM1$	4	1.0848	0.0052	4.2524e-19	9.2702e-148	7.99	0.807937
$VTM2$	4	1.0879	0.0021	1.0258e-26	1.9657e-214	8.05	0.878774
$VTM3$	4	1.0899	1.5396e-04	2.7156e-31	2.5507e-245	7.99	0.883730
$VTM4$	3	1.0900	2.3207e-08	5.1552e-123	5.1552e-123	14.99	0.784262

Table 1: Comparison of numerical results for test functions.

Table 2: Results for the *fzero* command in MATLAB.

f	x_0	N_1	N	d_1	$f(x_n)$	x^*	Zero found in the interval
f_1	-1.0	7	6	20	$2.2204e-16$	-0.7849	$[-0.77, -1.16]$
f_2	0.9	10	6	27	0	1.3652	$[0.32, 1.47]$
f_3	-0.9	10	6	26	$-1.6653e-16$	-0.4566	$[-0.32, -1.30]$
f_4	1.5	11	8	30	0	0.4100	$[0.14, 2.46]$

When the shot hits the slope, there's a value of x for which $P(x) = \omega(x)$ for each value of x . We wish to discover the value of θ that maximize x (See more [27]).

$$\omega(x) = P(x) = h + x \tan \theta - \frac{gx^2}{2v^2} \sec^2 \theta. \quad (29)$$

Table 3 appears that the method *VTM4* is converging with least number of iteration with least error than

IF	N	error	<i>cpu time</i> (s)	ρ
<i>NM</i>	7	4.3980e-76	1.074036	1.99
<i>LWM</i>	3	1.5610e-66	0.658235	8.03
<i>SAM</i>	3	1.2092e-61	0.754623	8.06
<i>CFGT</i>	3	3.3018e-89	0.731083	9.03
<i>CTV</i>	3	3.5871e-73	0.689627	8.02
<i>NCS</i>	3	3.0839e-70	0.714955	8.03
<i>VTM1</i>	3	4.3980e-76	0.683299	8.02
<i>VTM2</i>	3	4.1159e-112	0.639101	10.03
<i>VTM3</i>	3	2.3793e-74	0.673341	8.05
<i>VTM4</i>	3	0	0.623352	15.99

Table 3: Numerical results on projectile motion problem

other compared methods. Hence, the method *VTM4* can be considered competent sufficient to existing other compared methods. Also, we accommodate that the hypothetical arrange of convergence and computational arrange of merging are roughly break even with.

4.2 Planck's Radiation Law Problem

We consider the following Planck's radiation law problem found in [7]:

$$\varphi(\lambda) = \frac{8\pi ch\lambda^{-5}}{e^{ch/\lambda kT} - 1}, \quad (30)$$

which calculates the vitality thickness inside an isothermal blackbody. Here, λ is the wavelength of the radiation, T is the supreme temperature of the blackbody, k is Boltzmann's steady, h is the Planck's consistent and c is the speed of light. Assume, we would like to decide wavelength λ which compares to greatest vitality thickness $\varphi(\lambda)$. From (30), we get

$$\varphi'(\lambda) = \left(\frac{8\pi ch\lambda^{-6}}{e^{ch/\lambda kT} - 1} \right) \left(\frac{(ch/\lambda kT)e^{ch/\lambda kT}}{e^{ch/\lambda kT} - 1} - 5 \right) = A \cdot B.$$

It can be checked that a maxima for φ occurs when $B = 0$ that is, when

$$\left(\frac{(ch/\lambda kT)e^{ch/\lambda kT}}{e^{ch/\lambda kT} - 1} \right) = 5.$$

IF	N	error	$cpu\ time(s)$	ρ
<i>NM</i>	7	1.8205e-70	1.043458	2.00
<i>LWM</i>	4	3.1188e-268	0.872883	7.99
<i>SAM</i>	4	1.9335e-298	0.869115	8.00
<i>CFGT</i>	4	0	0.937835	7.99
<i>CTV</i>	4	5.8673e-282	0.890461	8.00
<i>NCS</i>	4	5.9695e-314	0.888809	7.99
<i>VTM1</i>	4	2.5197e-322	0.813324	8.00
<i>VTM2</i>	4	0	0.866063	8.00
<i>VTM3</i>	4	0	0.857505	7.99
<i>VTM4</i>	3	2.8513e-170	0.888935	16.36

Table 4: Numerical results on Planck's radiation law problem

Table 4 appears that the method *VTM4* is converging with least number of iteration with least error than other compared methods. Hence, the method *VTM4* can be considered competent sufficient to existing other compared methods. Also, we accommodate that the hypothetical arrange of convergence and computational arrange of merging are roughly break even with.

5 Basins of attraction

The consider on basin of attractions of the rational function related to an iterative procedure gives basic information around blending and strength of the procedure. To start with, we grant underneath a number of essential definitions of rational function in organize to think about capacities inside the complex space with complex zeros as found in [3, 39].

The Fatou set is characterized as the set of focuses whose circles tend to an pulling in settled point z_0 . The closure of the set comprising of repulsing settled points is called Julia set which is nothing but the complement of Fatou set, which builds up the borders between the basins of attraction. That suggests, the basin of fascination of any settled point incorporates a put to the Fatou set and the boundaries of these basins of fascination have a place to the Julia set.

Consider a square region having the boundaries $\mathbb{R} \times \mathbb{R} = [-3, 3] \times [-3, 3]$ of 90000 lattice points. These points are gotten from 300 columns and 300 lines which see just like the pixels of a computer show and this speak to a locale of the complex plane. The iterative strategy endeavors a zero z_j^* of the condition with a condition $|f(z^{(k)})| < 1e - 4$ and a most extreme of 100 iteration, we conclude that $z^{(0)}$ is within the basin of attraction of this zero. In the event that the iterative strategy beginning in $z^{(0)}$ comes to a zero in N iterations ($N \leq 100$), at that point we check this point $z^{(0)}$ with colors in case $|z^{(N)} - z_j^*| < 1e - 4$. On the off chance that $N > 50$, we conclude that the starting point has diverged and we assign a dark blue color. Let N_D be number of diverging points and we check the number of beginning points which converge in 1, 2, 3, 4, 5 or over 5 iterations. In this way, we recognize the basin attractors by distinctive colors for diverse roots and diverse behaviors like converging or diverging. We analyze the basins of attraction for new eighth order methods and a few equivalent methods for the three polynomials $p_1(z) = z^2 - 1$, $p_2(z) = z^3 - 1$ and $p_3(z) = z^4 - 1$.

Figure 1: Basins of attraction for $p_1(z) = z^2 - 1$

5.1 Polynomiographs of $p_1(z) = z^2 - 1$

The roots for polynomials of $p_1(z)$ are given by $\alpha_1 = 1$, $\alpha_2 = -1$. The polynomiographs of $p_1(z)$ are displayed in Fig. 1, though Table 5 presents the number of meeting and wandering mesh points for each iterative method. We observe that the proposed methods $VTM1$, $VTM2$, and $VTM3$ has no chaotic behaviour, no divergent (N_D) points, and have less mean (μ) number of iteration then other compared methods.

5.2 Polynomiographs of $p_2(z) = z^3 - 1$

The roots for polynomials of $p_2(z)$ are given by $\alpha_1 = 1$, $\alpha_2 = -0.5000 - 0.8660i$ and $\alpha_3 = -0.5000 + 0.8660i$. The polynomiographs of $p_2(z)$ are displayed in Fig. 2, though Table 6 presents the number of meeting and wandering mesh points for each iterative method. We observe that the proposed methods $VTM1$, $VTM2$, and $VTM3$ has less chaotic behaviour, no divergent points, and have less mean number of iteration then other compared methods.

5.3 Polynomiographs of $p_3(z) = z^4 - 1$

The roots for polynomials of $p_3(z)$ are given by $\alpha_1 = 1$, $\alpha_2 = -1$, $\alpha_3 = i$ and $\alpha_4 = -i$. The polynomiographs of $p_3(z)$ are displayed in Fig. 3, though Table 7 presents the number of meeting and wandering mesh points for each iterative method. We observe that the proposed methods $VTM1$, $VTM2$, and $VTM3$ has less chaotic behaviour, less divergent points, and have less mean number of iteration then other compared methods.

Remark 2: We acclimate that a point z_0 containing to the Julia set at whatever point the elements in a neighborhood of point displays touchy on the conditions based. Hence, adjacent introductory conditions driving to the marginally diverse behavior afterward in a few number of iterations. Subsequently, a few compared

I.F.	$N = 1$	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N > 5$	N_D	μ
<i>LWM</i>	2724	63304	16396	4112	1800	1664	0	2.3949
<i>SAM</i>	2636	69312	12476	2396	1068	2112	0	2.3459
<i>CFG</i>	5884	70336	9956	2572	844	408	28	2.1660
<i>CTV</i>	4360	72248	10320	2220	664	188	0	2.1465
<i>NCS</i>	3672	71536	12056	2004	588	144	0	2.1642
<i>VTM1</i>	5104	74608	8988	1220	80	0	0	2.0729
<i>VTM2</i>	7720	74172	7020	1056	32	0	0	2.0168
<i>VTM3</i>	11480	69576	7680	1172	92	0	0	1.9869

Table 5: Results of the polynomials $p_1(z) = z^2 - 1$ Figure 2: Basins of attraction for $p_2(z) = z^3 - 1$

I.F.	$N = 1$	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N > 5$	N_D	μ
<i>LWM</i>	1284	37232	27774	8004	4760	10946	0	3.4772
<i>SAM</i>	986	47510	17100	4912	2404	17088	880	5.5056
<i>CFG</i>	2258	45258	30130	7758	2466	2130	250	2.8863
<i>CTV</i>	2158	59938	18684	4586	1858	2776	0	2.5273
<i>NCS</i>	1492	44222	26260	8480	4886	4660	0	2.8833
<i>VTM1</i>	2330	56398	22710	5500	1896	1166	0	2.4713
<i>VTM2</i>	3868	59206	17626	5988	2122	1190	0	2.4177
<i>VTM3</i>	1918	60026	19696	5622	1810	928	0	2.4296

Table 6: Results of the polynomials $p_2(z) = z^3 - 1$

Figure 3: Basins of attraction for $p_3(z) = z^4 - 1$

I.F.	$N = 1$	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N > 5$	N_D	μ
<i>LWM</i>	840	18128	32584	10312	5456	22680	670	5.7375
<i>SAM</i>	592	28468	23172	9368	4972	23428	5008	8.2523
<i>CFGT</i>	1196	17660	40368	11300	6036	13440	1296	3.8561
<i>CTV</i>	1464	43824	22728	8080	4152	9752	536	3.6437
<i>NCS</i>	916	20372	34108	11880	6296	16428	416	4.1533
<i>VTM1</i>	1464	30528	34440	11752	5168	6648	490	3.4518
<i>VTM2</i>	2332	38404	28732	11064	4576	4892	341	3.1929
<i>VTM3</i>	1036	32120	22768	15556	8332	10188	288	3.6422

Table 7: Results of the polynomials $p_3(z) = z^4 - 1$

methods are getting many divergent initial points. The boundaries of the basins of attraction are Julia set of the iteration function. Note that the proposed methods are less chaotic and more reliable than the other compared methods.

6 Further Development

In this section, we are considering third order method (2) to develop new sixteenth order iterative methods by using weight function for solving system of nonlinear equations, whereas the method required only two function, two derivative and only one inverse of Jacobian needed per cycle. Hence the new method having the sense that, Kung-Traub conjecture is fails in scalar equations. Recently, the similar work done by Ahmad [1], Babajee [4], Babajee and Madhu [5] and they proved their methods are fails in Kung-Traub conjecture for quadratic equations.

Let us modified the method (2) for solving system of nonlinear equation as given below

$$\begin{aligned} \mathbf{y} &= \mathbf{x} - \mathbf{f}'(\mathbf{x})^{-1}\mathbf{f}(\mathbf{x}), \\ \psi_{(s+3)^{th}TM}(\mathbf{x}) &= \mathbf{y} - \mathbf{f}'(\mathbf{x})^{-1}\mathbf{f}(\mathbf{y}) \times T(\eta(\mathbf{x}), s), \end{aligned} \quad (31)$$

where $T(\eta(\mathbf{x}), s) = I + \sum_{i=1}^s \alpha_i(\eta - I)^i$, $\eta = \mathbf{f}'(\mathbf{x})^{-1}\mathbf{f}'(\frac{\mathbf{x}+\mathbf{y}}{2})$, $s \geq 1$, α_i 's are constants, and I is the identity matrix. Here, the system of nonlinear equations $\mathbf{f}(\mathbf{x}) = \mathbf{0}$, where $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x}))^T$, $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $\forall i = 1, 2, \dots, n$ defined as

$$f_i(\mathbf{x}) = b_i + \sum_{l=1}^n \sum_{m=1}^n b_{l,m} x_l x_m, \quad b_i, b_{l,m}, i, l, m = 1, \dots, n, \text{ are constants.}$$

and $\mathbf{f} : \mathbf{D} \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a smooth map and $\mathbf{D}$ is an open and convex set, where we assume that

$\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)^T$ is a zero of the system and $\mathbf{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})^T$ is an initial guess sufficiently close to $\mathbf{x}^*$.

Let us define

$$\mathbf{c}_2 = \frac{1}{2}[\mathbf{f}'(\mathbf{x}^*)]^{-1}\mathbf{f}^{(2)}(\mathbf{x}^*), \quad \mathbf{e}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$$

Using the notations in [14], it is noted that $\mathbf{c}_2\mathbf{e}^{(k)} \in \mathcal{L}(\mathbb{R}^n)$. The error at the $(k+1)$ th iteration is $\mathbf{e}^{(k+1)} = \mathbf{L}(\mathbf{e}^{(k)})^p + O((\mathbf{e}^{(k)})^{p+1})$, where $\mathbf{L}$ is a p -linear function $\mathbf{L} \in \mathcal{L}(\mathbb{R}^n \times \dots \times \mathbb{R}^n, \mathbb{R}^n)$, is called the *error equation* and p is the *order of convergence*. Observe that $(\mathbf{e}^{(k)})^p$ is $(\mathbf{e}^{(k)}, \mathbf{e}^{(k)}, \dots, \mathbf{e}^{(k)})$.

This following theorem can be proved with help of MATHEMATICA software.

Theorem 6.1. *Let $\mathbf{f} : \mathbf{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be twice Frechet differentiable at each point of an open convex neighborhood $\mathbf{D}$ of $\mathbf{x}^* \in \mathbb{R}^n$, that is a solution of the quadratic system $\mathbf{f}(\mathbf{x}) = \mathbf{0}$. Let us suppose that $\mathbf{f}'(\mathbf{x})$ is continuous and nonsingular in $\mathbf{x}^*$, and $\mathbf{x}^{(0)}$ is close enough to $\mathbf{x}^*$. Then the sequence $\{\mathbf{x}^{(k)}\}_{k \geq 0}$ obtained using the iterative expressions (31), $s = 13$ converge to $\mathbf{x}^*$ with order 16, when*

$$\begin{aligned} \alpha_1 &= -2, \alpha_2 = 5, \alpha_3 = -14, \alpha_4 = 42, \alpha_5 = -132, \alpha_6 = 429, \alpha_7 = -1430, \alpha_8 = 4862, \alpha_9 = -16796, \\ \alpha_{10} &= 58786, \alpha_{11} = -208012, \alpha_{12} = 742900, \alpha_{13} = -2674440. \end{aligned} \quad (32)$$

and it satisfies the error equation

$$\mathbf{e}^{(k+1)} = 9694845\mathbf{c}_2^{15}(\mathbf{e}^{(k)})^{16} + O((\mathbf{e}^{(k)})^{17}). \quad (33)$$

Remark 3: Generally in scalar equation, Kung-Traub conjecture says that four function evaluation reaches maximum eighth order convergence. But, this new method getting sixteenth order convergence.

Remark 4: This new method fails in Kung-Traub conjecture only for quadratic equation.

Note 1: Researchers can develop higher order methods in similar way for cubic, quartile and higher degree equations.

6.1 Numerical examples

This section deals with numerical comparisons in the MATLAB computer code rounding to 1000 significant digits. The criteria to stopping used for the iterative process

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_2 < 10^{-100}. \quad (34)$$

The approximated computational order of convergence p_c given by (see [17])

$$p_c \approx \frac{\log(\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_2 / \|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|_2)}{\log(\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|_2 / \|\mathbf{x}^{(k-1)} - \mathbf{x}^{(k-2)}\|_2)}. \quad (35)$$

Test Problem 1 (TP1) We consider the following nonlinear system:

$$\begin{aligned} x_1^2 + x_2^2 - 7 &= 0, \\ x_1 - x_2 + 1 &= 0. \end{aligned} \quad (36)$$

Whose positive root is given by $x_2^* = \frac{1+\sqrt{13}}{2} = 2.302775638$. Therefore $x_1^* = x_2^* - 1 = \frac{\sqrt{13}}{2} = 1.302775638$. We use $\mathbf{x}^{(0)} = (1, 2)^T$ as initial vector.

Test Problem 2 (TP2) We consider the following nonlinear system:

$$\begin{aligned} x_1^2 + x_2^2 - 1 &= 0, \\ x_1^2 - x_2^2 - 0.5 &= 0. \end{aligned} \quad (37)$$

Whose root is given by $x_2^* = \frac{\sqrt{3}}{2} = 0.866025403$ and therefore $x_1^* = \frac{1}{2}$. We choose initial point far from the root, $\mathbf{x}^{(0)} = (2, 3)^T$.

Test Problem 3 (TP3) We consider the following nonlinear system:

$$\begin{cases} x_2x_3 + x_4(x_2 + x_3) = 0, \\ x_1x_3 + x_4(x_1 + x_3) = 0, \\ x_1x_2 + x_4(x_1 + x_2) = 0, \\ x_1x_2 + x_1x_3 + x_2x_3 = 1. \end{cases}$$

We solve this system using the initial approximation $x^{(0)} = (0.5, 0.5, 0.5, -0.2)^T$. The solution of this system is $\alpha \approx (0.577350, 0.577350, 0.577350, -0.288675)^T$.

In table 8, we investigated the number of iterations N required to converge to the solutions, the total number of function evaluations n_{total} , the total number of inverse evaluations n_{inv} , computational order of convergence p_c and the residual minimum error err_{min} . The n_{total} is counted as sum of the total number of function evaluations in F and F' at point x^k . For example, here we shall calculate n_{total} for the TP3. In this case, requires four function evaluations in F and twelve function evaluations in F' . Per iteration NM method uses one F and one F' , which implies that n_{total} in 8 iterations is 128. Here, we can observe that the computational order of convergence is supports the theoretical order of convergence. The proposed method (31) requires less iterations than other compared methods. Also, the proposed method requires less n_{total} and n_{inv} than other compared methods in all the test problems.

Table 8: Numerical results for test problems (TP)

TP	Methods	N	p_c	err_{min}	F	F'	n_{total}	n_{inv}
TP1	NM (1)	8	1.99	7.4894e-133	8	8	32	8
	TM (2)	6	3.00	2.9040e-210	12	6	36	6
	MJ Eq. (3) in [34]	5	3.99	1.9385e-197	5	10	30	5
	HM Eq. (28) in [24]	5	3.99	1.1000e-265	5	10	30	10
	SGS Eq. (3) in [41]	5	3.99	2.7055e-230	5	10	30	10
	BMJ Eq. (6) in [6]	5	4.00	4.7688e-220	5	10	30	5
	CM Eq. (1) in [15]	4	5.99	8.8869e-177	8	8	32	8
	MBJ Eq. (6) in [32]	5	5.00	0	10	10	40	5
	Proposed method (31)	3	16.23	3.5829e-129	3	6	18	3
TP2	NM (1)	11	1.99	2.3168e-150	11	11	66	11
	TM (2)	8	2.99	4.0114e-223	16	8	64	8
	MJ Eq. (3) in [34]	7	3.99	0	7	14	70	7
	HM Eq. (28) in [24]	6	3.99	2.3168e-150	6	12	60	12
	SGS Eq. (3) in [41]	6	3.99	3.0148e-113	6	12	60	12
	BMJ Eq. (6) in [6]	6	3.99	1.4366e-105	6	12	60	6
	CM Eq. (1) in [15]	5	5.99	1.9241e-115	10	10	60	10
	MBJ Eq. (6) in [32]	6	4.99	5.1132e-221	12	12	72	6
	Proposed method (31)	4	15.82	1.0015e-161	4	8	40	4
TP3	NM (1)	8	1.99	7.9403e-147	8	8	128	8
	TM (2)	6	3.00	9.0798e-238	12	6	120	6
	MJ Eq. (3) in [34]	5	4.00	4.9944e-227	5	10	140	5
	HM Eq. (28) in [24]	5	3.99	3.0288e-293	5	10	140	10
	SGS Eq. (3) in [41]	5	3.99	8.7557e-259	5	10	140	10
	BMJ Eq. (6) in [6]	5	4.00	5.3612e-249	5	10	140	5
	CM Eq. (1) in [15]	4	5.99	5.3809e-201	8	8	128	8
	MBJ Eq. (6) in [32]	4	5.00	9.7403e-104	8	8	128	8
	Proposed method (31)	3	16.14	9.5006e-161	3	6	84	3

6.2 Application on One-dimensional Bratu Problem

The 1-D Bratu problem [8] is given by

$$\frac{d^2U}{dx^2} + \lambda \exp U(x) = 0, \lambda > 0, 0 < x < 1, \quad (38)$$

with the boundary value conditions $U(0) = U(1) = 0$. The one-dimensional Bratu problem has bifurcated, two known, actual solutions for the values of $\lambda < \lambda_c$, no solution for $\lambda > \lambda_c$, and one solution for $\lambda = \lambda_c$.

The value of λ_c is $8(\eta^2 - 1)$, where η is the fixed point of function $\coth(x)$. The exact solution to the problem (38) is

$$U(x) = -2 \ln \left[\frac{\cosh(x - \frac{1}{2})^{\frac{\theta}{2}}}{\cosh(\frac{\theta}{4})} \right], \quad (39)$$

where θ is a constant to be find, that satisfies the boundary value conditions and is fastidiously chosen and assuming the answer of the problem (38). Similar way as in [36], we trying to show the way to get the crucial

worth of λ . Substitute the equation (39) in the equation (38), simplify and collocate at the point $x = \frac{1}{2}$ between in the interval. Selected some other point, but low order of approximations are possibly to being better think the collocation factors are dispensed extremely similarly for the duration of the region. Then, we have

$$\theta^2 = 2\lambda \cosh^2 \left(\frac{\theta}{4} \right). \quad (40)$$

Differentiating equation (40) with respect to θ and setting $\frac{d\lambda}{d\theta} = 0$, the critical value λ_c satisfies

$$\theta = \frac{1}{2} \lambda_c \cosh \left(\frac{\theta}{4} \right) \sinh \left(\frac{\theta}{4} \right). \quad (41)$$

By eliminating λ from equations (40) and (41), we have the value of θ_c for the critical λ_c satisfying

$$\frac{\theta_c}{4} = \coth \left(\frac{\theta_c}{4} \right) \quad (42)$$

for $\theta_c = 4.798714560$ can be found by using an iterative method. Then, we will get $\lambda_c = 3.513830720$ from (40). The one dimensional problem by using standard finite difference scheme is given below

$$F_j(U_j) = \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} + \lambda \exp U_j = 0, \quad j = 1..M-1 \quad (43)$$

with boundary conditions $U_0 = U_M = 0$ and having step size $h = 1/M$. There are square measure $M-1$ unknowns ($n = M-1$). The Jacobian may be a sparse matrix and its typical range of nonzero per row is 3. Its noted that the finite difference scheme converges to the lower solution of the 1-D Bratu using the initial vector $U^{(0)} = (0, 0, \dots, 0)^T$.

We use $M = 101$ ($m = 100$) and check for 350 λ 's in the interval $(0, 3.5]$ (interval breadth = 0.01). For every λ , we let N_λ be the minimum number of iterations for which $\|U_j^{(k+1)} - U_j^{(k)}\|_2 < 1e-13$, where the approximation $U_j^{(k)}$ is calculated correct to fourteen decimal places. Let $\overline{N}_\lambda$ be the mean of iteration number for the 350 λ 's.

Table 9: Comparison of methods for number of λ 's converging for 1-D Bratu problem

Method	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N > 5$	$\overline{N}_\lambda$
NM (1)	0	12	115	142	81	4.9314
TM (2)	1	136	171	37	5	3.7571
MJ Eq. (3) in [34]	4	227	107	11	1	3.3714
HM Eq. (28) in [24]	0	140	206	2	2	3.6257
SGS Eq. (3) in [41]	4	237	100	8	1	3.3314
BMJ Eq. (6) in [6]	4	234	102	7	3	3.3914
CM Eq. (1) in [15]	3	215	122	8	2	3.4029
MBJ Eq. (6) in [32]	0	31	209	97	13	4.2714
Proposed method (31)	6	247	93	2	2	3.2800

Table 9 give the results for the 1-D Bratu problem, where N denoting number of iterations for convergence. The proposed method (31) is the most efficient method among the other compared methods because it has the lowest mean number of iteration and its converging many initial points are 2 and 3 iterations. Thus, the proposed method is converging faster than other compared methods.

7 Conclusions

This paper has developed a classes of optimal eighth order methods and sixteenth order method without memory to solve nonlinear scalar equations numerically. The advantage of the proposed methods were high efficiency index, which do not require second derivative, high accuracy in numerical examples and also consistency with the conjecture of Kung-Traub. We have tested some examples using the proposed schemes and some known schemes, which illustrate the superiority of the proposed methods. Also, Projectile motion problem and Planck's radiation law problem are used to validate our proposed methods. The results obtained are interesting and encouraging for the new method. We have also compared the basins of attraction of various eighth order methods in the complex plane.

Further, we have developed sixteenth order iterative method with four functional evaluation namely two F and two F' for solving system of quadratic equation. We have tested some examples using the proposed schemes and some known schemes, which illustrate the superiority of the proposed methods. Also, we test new method and some existing methods on the 1-D bratu problem and the results obtained are interesting and encouraging for the new methods. Concluding that the proposed method having good efficiency in all the test function as compared to other methods. Hence, the new method can be considered competent enough to existing other compared methods.

8 Declarations

Availability of data and materials: Not applicable.

Competing interests: The authors declare no competing of interest.

Funding: This work was supported by the Khalifa University internal funding (Grant Numbers:FSU-2019-07), Abu Dhabi, UAE.

Author contributions: The contributions of all the authors have been similar. All of them have worked together to develop the present manuscript.

Acknowledgment: Not applicable.

References

- [1] F Ahmad. Comment on: On the Kung-Traub conjecture for iterative methods for solving quadratic equations. *Algorithms*, 9:30, 2016.
- [2] S Amat, S Busquier, and S Plaza. Dynamics of a family of third-order iterative methods that do not require using second derivatives. *Appl. Math. Comp.*, 154:735–746, 2004.
- [3] S. Amat, S. Busquier, and S. Plaza. Review of some iterative root-finding methods from a dynamical point of view. *SCIENTIA Series A: Mathematical Sciences*, 10:3–35, 2004.
- [4] D K R Babajee. On the Kung-Traub conjecture for iterative methods for solving quadratic equations. *Algorithms*, 9:1, 2016.
- [5] D K R Babajee and K Madhu. Comparing two techniques for developing higher order two-point iterative methods for solving quadratic equations. *SeMA Journal*, 76:227–248, 2019.

- [6] D K R Babajee, K Madhu, and J Jayaraman. On some improved harmonic mean newton-like methods for solving systems of nonlinear equations. *Algorithms*, 8:895–909, 2015.
- [7] B. Bradie. *A Friendly Introduction to Numerical Analysis*. Pearson Education Inc, 2006.
- [8] R. Buckmire. Investigations of nonstandard Mickens-type finite-difference schemes for singular boundary value problems in cylindrical or spherical coordinates. *Num. Meth. P. Diff. Eqns.*, 19(2):380–398, 2003.
- [9] S.C. Chapra and R.P. Canale. *Numerical Methods for Engineers*. McGraw-Hill, New York, 1988.
- [10] F. Chicharro, A. Cordero, J.M. Gutiérrez, and J.R. Torregrosa. Complex dynamics of derivative-free methods for nonlinear equations. *Applied Mathematics and Computation*.
- [11] C Chun. A family of composite fourth-order iterative methods for solving nonlinear equations. *Applied Mathematics and Computation*, 187:951–956, 2007.
- [12] C. Chun, M. Y. Lee, B. Neta, and J. Dzunic. On optimal fourth-order iterative methods free from second derivative and their dynamics. *Applied Mathematics and Computation*, 218:6427–6438, 2012.
- [13] A. Cordero, M. Fardi, M. Ghasemi, and J.R. Torregrosa. Accelerated iterative methods for finding solutions of nonlinear equations and their dynamical behavior. *Calcolo*, 51:17–30, 2014.
- [14] A Cordero, J L Hueso, E Martinez, and J R Torregrosa. A modified newton-jarratts composition. *Numer. Algor.*, 55:87–99, 2010.
- [15] A. Cordero, J. L. Hueso, E. Martinez, and J. R. Torregrosa. Increasing the convergence order of an iterative method for nonlinear systems. *Applied Mathematics Letters*, 25:2369–2374, 2012.
- [16] A. Cordero, J. L. Hueso, E. Martinez, and J.R. Torregrosa. New modifications of potra-ptak’s method with optimal fourth and eighth orders of convergence. *J. Comput. Appl. Math.*, 234:2969–2976, 2010.
- [17] A Cordero and J R Torregrosa. Variants of Newton’s method using fifth-order quadrature formulas. *Appl. Math. Comp.*, 190:686–698, 2007.
- [18] A. Cordero, J. R. Torregrosa, and P. Vindel. Dynamics of a family of chebyshevhalley type methods. *Applied Mathematics and Computation*.
- [19] A. Cordero, J.R. Torregrosa, and M.P. Vasileva. A family of modified ostrowski’s methods with optimal eighth order of convergence. *Appl. Math. Lett.*, 24:2082–2086, 2011.
- [20] J. H. Curry, L. Garnett, and D. Sullivan. On the iteration of a rational function: computer experiments with newton’s method. *Communications in Mathematical Physics*, 91:267–277, 1983.
- [21] J. A. Ezquerro, M. A. Hernandez, and M. A. Salanova. A new family of mulitipoint methods of second order. *Numer. Funct. Anal. And Optimiz.*, 19(5-6):499–512, 1998.
- [22] K. Gdawiec, W. Kotarski, and Lisowska. Newton’s method with fractional derivatives and various iteration processes via visual analysis. *Numer. Algor. Doi:10.1007/s11075-020-00919-4*, 2020.
- [23] E Halley. A new, exact and easy method for finding the roots of equations generally and that without any previous reduction. *Phil. Trans. R. Soc. London*, 18:136–148, 1694.
- [24] H H H Homeier. On Newton-type methods with cubic convergence. *Comp. Appl. Math.*, 176(2):425–432, 2005.

- [25] P Jarratt. Some fourth order multipoint iterative methods for solving equations. *Math. Comp.*, 20:434–437, 1966.
- [26] P. Jarratt. Some efficient fourth order multipoint methods for solving equations. *BIT*, 9:119–124, 1969.
- [27] Robert Kantrowitz and Michael M. Neumann. Some real analysis behind optimization of projectile motion. *Mediterranean Journal of Mathematics*, 11(4):1081–1097, 2014.
- [28] R F King. A family of fourth-order methods for solving nonlinear equations. *SIAM J. Numer. Anal.*, 10:876–879, 1973.
- [29] J Kou, Y Li, and X Wang. A composite fourth-order iterative method for solving non-linear equations. *Appl. Math. Comp.*, 184:471–475, 2007.
- [30] H.T. Kung and J.F. Traub. Optimal order of one-point and multi-point iteration. *J. Assoc. Comput. Math.*, 21:643–651, 1974.
- [31] L Liu and X Wang. Eighth-order methods with high efficiency index for solving nonlinear equations. *Appl. Math. Comp.*, 215:3449–3454, 2010.
- [32] K Madhu, D K R Babajee, and J Jayaraman. An improvement to double-step newton method and its multi-step version for solving system of nonlinear equations and its applications. *Numerical Algorithms*, 74:593–607, 2017.
- [33] K. Madhu and J. Jayaraman. Higher order methods for nonlinear equations and their basins of attraction. *Mathematics*, 4(22), 2016.
- [34] K Madhu and J Jayaraman. Some higher order newton-like methods for solving system of nonlinear equations and its applications. *Int. J. Appl. Comput. Math*, 3:2213–2230, 2017.
- [35] Beny Neta, Changbum Chun, and Melvin Scott. Basins of attraction for optimal eighth order methods to find simple roots of nonlinear equations. *Applied Mathematics and Computation*, 227:567–592, 2014.
- [36] S A Odejide and Y A S Aregbesola. A note on two dimensional Bratu problem. *Kragujevac J. Math.*, 29:49–56, 2006.
- [37] A M Ostrowski. *Solutions of Equations and System of equations*. Academic Press, New York, 1960.
- [38] M. S. Petkovic, B. Neta, L. D. Petkovic, and J. Dzunic. *Multipoint Methods for Solving Nonlinear Equations*. Academic Press, 2013.
- [39] M. Scott, B. Neta, and C. Chun. Basin attractors for various methods. *Applied Mathematics and Computation*, 218:2584–2599, 2011.
- [40] J R Sharma and H Arora. An efficient family of weighted-newton methods with optimal eighth order convergence. *Appl. Math. Lett.*, 29:1–6, 2014.
- [41] J. R. Sharma, R. K. Guha, and R. Sharma. An efficient fourth order weighted-newton method for systems of nonlinear equations. *Numer. Algor*, 62:307–323, 2013.
- [42] J F Traub. *Iterative Methods for the Solution of Equations*. Prentice-Hall, New Jersey, 1964.
- [43] J. F. Traub. *Iterative Methods for the Solution of Equations*. Chelsea Publishing Company, New York, NY, USA, 1977.

- [44] E. R. Vrscay. Julia sets and mandelbrot-like sets associated with higher order schroder rational iteration functions: a computer assisted study. *Mathematics of Computation*, 46:151–169, 1986.
- [45] E. R. Vrscay and W. J. Gilbert. Extraneous fixed points, basin boundaries and chaotic dynamics for schroder and konig rational iteration functions. *Numerische Mathematik*, 52:1–16, 1987.

Figures

Figure 1

Basins of attraction for $p_1(z) = z^2 - 1$

Figure 2

Basins of attraction for $p_2(z) = z^3 - 1$

Figure 3

Basins of attraction for $p_3(z) = z^4 - 1$