# Optimal fourth-order iterative method free from derivative

*F. Soleymani*

# OPTIMAL FOURTH-ORDER ITERATIVE METHODS FREE FROM DERIVATIVES

F. SOLEYMANI

*Abstract.* The construction of optimal fourth-order iterative schemes for solving univariate non-linear equations is discussed. Per iteration, the methods consist of three evaluations of the function and they are free from any derivative calculation which property is so fruitful in engineering problems. We analytically show the fourth-order convergence. Numerical examples are considered to confirm the applicability and to justify the rapid convergence of the novel iterative algorithms.

## 1. INTRODUCTION

This paper is concerned with the numerical solution of single valued nonlinear equations of the form $f(x) = 0$. The exact and analytical solutions of such equations are not always at hand. That is why the accurate iterative methods, in which the total number of evaluations is appropriate, are required. Let $\alpha$ be the root of such an equation. This root can fall into two categories, i.e., it can be a simple zero [8] or a multiple zero [2]. In this study, we are concerned with simple zeros. Let $\alpha$ be in the open interval $D$, let it be the simple root the nonlinear equation $f(x) = 0$, then $f(\alpha) = 0$ and $f'(\alpha) \neq 0$. In engineering problems and/or in the real-world situations, when the calculation of the derivatives of the functions are not practical and/or cost so much time, we need root solvers that do not use derivative calculations per iteration in order to obtain an accurate approximation of the exact root. For example, consider the problem of finding the simple roots of the function $g(x) = \sin(2\cos(x)) - 1 - x^2 + e^{\sin(x^3)}$. Clearly, finding the derivatives of this function is hard and takes a great deal of time. Hence, we provide efficient fourth-order methods which are optimal and are based on the still unproved Kung-Traub hypothesis [4] concerning the "multi-point without memory" iteration schemes. According to their conjecture of optimality for multi-point iterations, a without memory iteration with $n$ evaluation per cycle can reach the optimal order of convergence $2^{(n-1)}$ and the

optimal efficiency index $2^{(n-1)/n}$. In fact, in 1964 Traub in the fundamental work [9] described that iterative (derivative-free or derivative-involved) methods for solving one variable nonlinear equations. He distinguished two main classes; one-point (one-step) with or without memory iterations and; multi-point (multi-step) iterations with or without memory. He also proved that the optimal order of convergence for a one-point (one-step) method, i.e. for Newton's or for Steffensen's, is two. After 10 years of that, in 1974 Kung and Traub [4] mentioned a hypothesis stating that the maximum convergence order for the multi-point without memory iterations cannot exceed $2^{(n-1)}$ where $n$ is the whole number of evaluations per full cycle. The reason of interest in multi-point iterations is that they possess better convergence rate and efficiency in solving real-world problems. Thus, in order to obtain an efficient method of order four in this work, we are allowed to use three evaluations of the function per iteration in the sense of Kung-Traub conjecture.

We here remark that if the sequence $\{x_n\}_{n=0}^{\infty}$ for a positive $\lambda$ and $p$ satisfies the following relation

$$\lim_{n \to \infty} \frac{|x_{n+1} - \alpha|}{|x_n - \alpha|^p} = \lambda, \tag{1.1}$$

then the iterative method which produces this sequence has local order of convergence $p$. We also remind that, the Ostrowski-Traub efficiency index could be provided by $p^{(1/\theta)}$, where $\theta$ is the total number of evaluations of the iterative scheme per full cycle.

The article is summarized as follows. After collecting some important derivative-free root solvers in Section 2, our optimal algorithms are developed in Section 3. In Section 4, comparisons are made between the existing methods and the new techniques to prove that our novel derivative-free techniques are effective and convenient. Finally in Section 5, our conclusion is presented.

## 2. SELECTIONS FROM THE LITERATURE

For a long time the Steffensen's method, which is given by

$$x_{n+1} = x_n - \frac{f(x_n)^2}{f(x_n + f(x_n)) - f(x_n)}, \tag{2.1}$$

was the only available quadratically derivative-free scheme. This method was obtained by replacing the forward finite-difference approximation for the first derivative in the well-known Newton's method.

In 2001, Wu et al. [10] presented another optimal derivative-free second-order method as follows

$$x_{n+1} = x_n - \frac{f(x_n)^2}{bf(x_n)^2 + f(x_n + f(x_n)) - f(x_n)}, \tag{2.2}$$

where the parameter $b$ should be chosen such that the denominator is non-zero; for example, $b = sign\{f(x_n + f(x_n)) - f(x_n)\}$. Motivated by these methods, two-step

iterative methods have been invented to increase the order of convergence and the efficiency indices.

In 2007, a derivative-free method of order three [3], in which we have three evaluations of the function, had been presented in the following form

$$\begin{cases} y_n = x_n - \frac{f(x_n)^2}{f(x_n+f(x_n))-f(x_n)}, \\ x_{n+1} = x_n - \frac{f^3(x_n)}{[f(x_n+f(x_n))-f(x_n)][f(x_n)-f(y_n)]}. \end{cases} \quad (2.3)$$

In 2010, another third-order iterative algorithm had been developed by Dehghan and Hajarian in [1]

$$\begin{cases} y_n = x_n - \frac{f(x_n)^2}{f(x_n+f(x_n))-f(x_n)}, \\ x_{n+1} = x_n - \frac{f(x_n)[f(y_n)+f(x_n)]}{f(x_n+f(x_n))-f(x_n)}. \end{cases} \quad (2.4)$$

As we can see, this algorithm also includes three evaluations of the function per iteration and therefore is not optimal with high efficiency index.

Recently, an accurate optimal fourth-order method [5] was proposed by Liu et al. as follows

$$\begin{cases} y_n = x_n - \frac{f(x_n)^2}{f(x_n+f(x_n))-f(x_n)}, \\ x_{n+1} = y_n - \frac{f[x_n,y_n]-f[y_n,z_n]+f[x_n,z_n]}{f[x_n,y_n]^2} f(y_n), \end{cases} \quad (2.5)$$

where $z_n = x_n + f(x_n)$. This method consists of three evaluations of the function per iteration in order to obtain fourth-order convergence. In this method $f[x_n, y_n], f[y_n, z_n], f[x_n, z_n]$ are divided differences of $f(x)$. We recall that they can be defined recursively via $f[x_i] = f(x_i); f[x_i, x_j] = \frac{f[x_i]-f[x_j]}{x_i-x_j}, x_i \neq x_j$.

In 2011, Zheng et al. in [11] extended the approach given by Liu et al. (2.5) to provide a three-parameter family of iterartions with optimal convergence rate four

$$\begin{cases} y_n = x_n - \frac{f(x_n)^2}{f(x_n+f(x_n))-f(x_n)}, \\ x_{n+1} = y_n - \frac{f[x_n,y_n]+(p-1)f[y_n,z_n]-(p-1)f[x_n,z_n]-\beta(y_n-x_n)(y_n-z_n)}{f[x_n,y_n]+pf[y_n,z_n]-pf[x_n,z_n]+a(y_n-x_n)(y_n-z_n)} \frac{f(y_n)}{f[x_n,y_n]}, \end{cases} \quad (2.6)$$

where $z_n = x_n + f(x_n)$ and $\beta, a, p$ are real valued parameters.

We remark here that Kung and Traub in the fundamental paper [4] provided the following derivative-free family ($\beta \in \mathbb{R} - \{0\}$) of methods by using the Inverse Interpolation

$$\begin{cases} y_n = x_n + \beta f(x_n), \\ z_n = y_n - \beta \frac{f(x_n)f(y_n)}{f(y_n)-f(x_n)}, \\ x_{n+1} = z_n - \frac{f(x_n)f(y_n)}{f(z_n)-f(x_n)}[\frac{1}{f[y_n,x_n]} - \frac{1}{f[z_n,y_n]}], \end{cases} \quad (2.7)$$

which have the following error equation $e_{n+1} = \frac{c_2(2c_2^2 - c_1 c_3)(1+c_1\beta)^2}{c_1^3} e_n^4 + O(e_n^5)$ where $c_j = \frac{f^{(j)}(\alpha)}{j!}$, $j \geq 1$.

For a more detailed description of this topic consult the papers [6,7] and the references therein.

## 3. THE PROPOSED METHODS

In this section, our contribution is based on the following two-step relation

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = y_n - \frac{f(y_n)}{f'(y_n)}. \end{cases} \tag{3.1}$$

Unfortunately, (3.1) includes two evaluations of the function and two evaluations of the first derivative per full cycle to the reach the convergence order 4 with $4^{1/4} \approx 1.4142$ as its index of efficiency. Hence, we must remedy two problems in (3.1). First, the derivative evaluations should disappear from this structure, and second, the number of evaluations must be decreased from four to three in such a way that the convergence rate does not decrease. Due to these requirements, the first thing that we could do is to replace $f'(y_n)$ with $f'(x_n)$ in the second sub-step. The second thing is to approximate $f'(x_n)$ as Steffensen did; see (2.1). This path reduces the number of evaluations from four to three, unfortunately the order decreases as well. Anyhow, we attain

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f[x_n, A_n]}, \\ x_{n+1} = y_n - \frac{f(y_n)}{f'(x_n)}. \end{cases} \tag{3.2}$$

where $A_n = x_n + f(x_n)$ and $n = 0, 1, 2, \cdots$. To remedy the existing problems in (3.2), first, $f'(x_n)$ should be estimated as efficiently as possible. Hopefully, we have the value of the function in the nodes $x_n, A_n$ and $y_n$ by now. By using the well-known method of undeterministic coefficients in the interpolating polynomial $p(t) = a_1 + a_2(t - x_n) + a_3(t - x_n)^2$, we obtain a robust approximation of $f'(x_n)$ in the second step of (3.2). Note that at the second step of (3.2), we have the values of the function in three nodes, but in the first step we have the values of the function only at two nodes. That is why $f$ at this step can be approximated by an interpolation polynomial of order two. In other words, $p'(x_n) = a_2$. Thus, by substituting the known data in the interpolation conditions $p(x_n) = f(x_n)$, $p(A_n) = f(A_n)$ and $p(y_n) = f(y_n)$; we obtain $f'(x_n) \approx \frac{x_n f[x_n, A_n] + A_n f[x_n, y_n] - x_n f[x_n, y_n] - f[x_n, A_n] y_n}{A_n - y_n}$. Therefore, we attain

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f[x_n, A_n]}, \\ x_{n+1} = y_n - \frac{(A_n - y_n)f(y_n)}{(x_n - y_n)f[x_n, A_n] + (A_n - x_n)f[x_n, y_n]}, \end{cases} \tag{3.3}$$

where the total number of evaluations are three and it is free from derivative.(3.3) satisfies the error equation $e_{n+1} = \frac{2(1+c_1)c_2^2}{c_1^2} e_n^3 + O(e_n^4)$, which shows that it is cubically

convergent and reaches the index of efficiency $3^{1/3} \approx 1.442$. Thus, our aim has not been achieved at this point. In order to reach the optimal order of convergence four with three evaluations of the function per iteration; we construct a weight function at the second step as follows:

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f[x_n, A_n]}, \\ x_{n+1} = y_n - \dfrac{(A_n - y_n)f(y_n)}{(x_n - y_n)f[x_n, A_n] + (A_n - x_n)f[x_n, y_n]}[1 + \gamma \dfrac{f(y_n)}{f(A_n)}]. \end{cases} \tag{3.4}$$

Its error equation is $e_{n+1} = -\dfrac{(1+c_1)c_2^2(-2+\gamma)}{c_1^2}e_n^3 + O(e_n^4)$ and $\gamma$ should be determined such that the order of convergence reaches four. That is to say, the coefficient of $e_n^3$ should vanish. Due to this, $\gamma = 2$ is taken into consideration. Finally, we obtain

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f[x_n, A_n]}, \\ x_{n+1} = y_n - \dfrac{(A_n - y_n)f(y_n)}{(x_n - y_n)f[x_n, A_n] + (A_n - x_n)f[x_n, y_n]}[1 + 2\dfrac{f(y_n)}{f(A_n)}]. \end{cases} \tag{3.5}$$

The order of convergence for the preceding method is analyzed in Theorem 1.

**Theorem 1.** *Assume that $f : D \subseteq \mathbb{R} \to \mathbb{R}$ is a sufficiently many times differentiable function with a simple zero $\alpha \subseteq D$, let $D$ be an open interval, let $x_0$ be a guess close enough to $\alpha$. In this case the new method (3.5) has fourth-order convergence.*

*Proof.* Let $e_n = x_n - \alpha$, $f(\alpha) = 0$ and $c_j = \dfrac{f^{(j)}(\alpha)}{j!}$, $j \geq 1$. Using Taylor expansion, we have

$$f(x_n) = c_1 e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + O(e_n^5), \tag{3.6}$$

and $f(A_n) = f(x_n + f(x_n)) = e_n(1 + c_1 + e_n(c_2 + e_n(c_3 + c_4 e_n)))(c_1 + c_2 e_n(1 + c_1 + e_n(c_2 + e_n(c_3 + c_4 e_n))) + c_3 e_n^2(1 + c_1 + e_n(c_2 + e_n(c_3 + c_4 e_n)))^2 + c_4 e_n^3(1 + c_1 + e_n(c_2 + e_n(c_3 + c_4 e_n)))^3)$. Furthermore, we obtain

$$x_n - \frac{f(x_n)^2}{f(x_n + f(x_n)) - f(x_n)} = \alpha + (1 + \frac{1}{c_1})c_2 e_n^2$$

$$+ \frac{(-(2 + (-2 + c_1)c_1)c_2^2 + c_1(1 + c_1)(2 + c_1)c_3)}{c_1^2}e_n^3 + O(e_n^4). \tag{3.7}$$

Now we should expand $f(y_n)$ around the simple root by using (3.7). We have

$$f(y_n) = (1 + c_1)c_2 e_n^2 + (-\frac{(2 + c_1(2 + c_1))c_2^2}{c_1} + (1 + c_1)(2 + c_1)c_3)e_n^3$$

$$+ \frac{1}{c_1^2}((5 + c_1(7 + c_1(4 + c_1)))c_2^3 - c_1(7 + c_1(10 + c_1(7 + 2c_1)))c_2 c_3$$

$$+ c_1^2(1 + c_1)(3 + c_1(3 + c_1))c_4)e_n^4 + O(e_n^5). \tag{3.8}$$

Using (3.8) and the Taylor expansion for the approximation function which is used in the second step, of our derivative-free method gives us

$$\frac{(A_n - y_n)f(y_n)}{(x_n - y_n)f[x_n, A_n] + (A_n - x_n)f[x_n, y_n]} = (1 + \frac{1}{c_1})c_2 e_n^2$$

$$+ \frac{(2 + c_1)(-(2 + c_1)c_2^2 + c_1(1 + c_1)c_3)}{c_1^2}e_n^3 + O(e_n^4). \tag{3.9}$$

For the weight function in the last step of (3.5), we obtain

$$1 + 2\frac{f(y_n)}{f(A_n)} = 1 + \frac{2c_2}{c_1}e_n^1 + \frac{(-2(3 + 2c_1)c_2^2 + 2c_1(2 + c_1)c_3)}{c_1^2}e_n^2 + O(e_n^3). \tag{3.10}$$

And finally, considering the above relations (3.7)-(3.10) in the last step of (3.5), we obtain

$$e_{n+1} = x_{n+1} - \alpha = \frac{(1 + c_1)c_2((5 + 3c_1)c_2^2 - c_1(1 + c_1)c_3)}{c_1^3}e_n^4 + O(e_n^5). \tag{3.11}$$

This completes the proof and shows that our proposed optimal derivative-free method is a fourth-order algorithm and agrees with the still un-proved conjecture of Kung-Traub on the optimality of multi-point iterations without memory. □

*Remark* 1. If one chooses the backward finite difference approximation in the Steffensen's method, i.e. $A_n = x_n - f(x_n)$, then another novel optimal fourth order method can be obtained as follows

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f[x_n, A_n]}, \\ x_{n+1} = y_n - \frac{(A_n - y_n)f(y_n)}{(x_n - y_n)f[x_n, A_n] + (A_n - x_n)f[x_n, y_n]}[1 + 2\frac{f(y_n)}{f(A_n)}], \end{cases} \tag{3.12}$$

where its error equation is

$$e_{n+1} = x_{n+1} - \alpha = \frac{(-1 + c_1)c_2((-5 + 3c_1)c_2^2 - (-1 + c_1)c_1c_3)}{c_1^3}e_n^4 + O(e_n^5). \tag{3.13}$$

## 4. NUMERICAL IMPLEMENTATIONS

In this section, numerical examples are furnished to re-verify the effectiveness of the proposed derivative-free methods. The comparison among the presented methods (**PM1**) (3.5) and (**PM2**) (3.12), the third-order method of Jain (**JM**), the third-order method of Dehghan and Hajarian (**DHM**), the fourth-order method of Liu et al. (**LM**), the fourth-order method of Kung and Traub with $\beta = 1$ (**KTM1**) and $\beta = 3$ (**KTM3**) are given. The test functions with their roots (up to 10 decimal places when they are not integers) are:

$$f_1(x) = (\sin x)^2 + x, \qquad\qquad\qquad\qquad\qquad \alpha = 0,$$
$$f_2(x) = (1 + x^3)\cos(\tfrac{\pi x}{2}) + \sqrt{1 - x^2} - \tfrac{2(9\sqrt{2} + 7\sqrt{3})}{27}, \qquad \alpha = 1/3,$$
$$f_3(x) = (\sin x)^2 - x^2 + 1, \qquad\qquad\qquad\qquad \alpha \approx 1.4044916482,$$
$$f_4(x) = e^{-x} + \sin(x) - 1, \qquad\qquad\qquad\qquad \alpha \approx 2.0768312745,$$
$$f_5(x) = xe^{-x} - 0.1, \qquad\qquad\qquad\qquad\qquad \alpha \approx 0.1118325591,$$
$$f_6(x) = x^2 + \sin(x) + x, \qquad\qquad\qquad\qquad \alpha = 0,$$
$$f_7(x) = \sin(2\cos(x)) - 1 - x^2 + e^{\sin(x^3)}, \qquad\qquad \alpha \approx 1.3061752018,$$
$$f_8(x) = \sin(2\cos(x)) - 1 - x^2 + e^{\sin(x^3)}, \qquad\qquad \alpha \approx -0.7848959876,$$
$$f_9(x) = \cos(x) + \sin(2x)\sqrt{1 - x^2} + \sin(x^2) + x^{14} + x^3 + \tfrac{1}{2x}, \quad \alpha \approx -0.9257722498,$$
$$f_{10}(x) = \tan(\ln(x)) + x^3 + \tfrac{1}{2x}, \qquad\qquad\qquad \alpha \approx 0.3974036712.$$

The results are provided in Table 1 based on the same Total Number of Evaluations (TNE) (12), i.e. after 4 iterations for JM, LM, KTM1, KTM2, PM1, and PM2. In fact, the absolute values of each test function for each starting point is listed in Table 1 after 4 full iterations. Note that Div. represents that the considered iterative scheme is divergent. All calculations were done with MATLAB 7.6 using 600 digit floating point (Digits: =600) with VPA Command. In the examples considered in this paper, the stopping criteria is the $|f(x_n)| \le \varepsilon$, where $\varepsilon = 10^{-600}$. In the implementation of the iterative methods, the appropriate choice of initial approximation value $x_0$ is very important since a badly chosen initial approximation produces a bad predictor and, consequently, destroys rapid convergence. Thus, we have selected the starting points close enough to the sought zero.

It could be said from these numerical experiments that the presented methods appear to be superior in compared to the other methods. Table 1 also reveals that the methods introduced in this study have similar performance compared to the other known methods of the same order for most of the numerical cases. From the results displayed in Table 1 and the number of numerical experiments, it can be concluded that the proposed multi-point derivative-free schemes possess quick convergence for good initial approximations. Moreover, after an extensive experimentation, we could mention that among all of the compared derivative-free schemes, it is impossible to select one as the best. Since, they behave similarly, i.e. for some initial guesses one is better while for other starting points another one would be appropriate.

In one word, we can infer that the convergence behavior of the considered multi-point methods strongly depends on the structure of the tested functions and the accuracy of the starting points. To compare the efficiency indices of different derivative-free methods, we provide Table 2. The computational order of convergence (namely, COC), which can be defined by

$$COC = \frac{\ln(|x_{n+1} - \alpha| / |x_n - \alpha|)}{\ln(|x_n - \alpha| / |x_{n-1} - \alpha|)}, \qquad\qquad (4.1)$$

TABLE 1. Comparison of different derivative-free methods with the same TNE (12)

| Fun. | Guess. | JM | DHM | LM | KTM1 | KTM3 | PM1 | PM2 |
|------|--------|-----|------|-----|------|------|-----|-----|
| $f_1$ | 0.6 | 0.3e-40 | 0.2e-32 | 0.1e-90 | 0.7e-88 | 0.4e-6 | 0.1e-81 | 0.1e-204 |
| $f_1$ | 0.8 | 0.6e-15 | 0.2e-9 | 0.2e-25 | 0.2e-22 | 0.5e-6 | 0.2e-10 | 0.8e-137 |
| $f_1$ | -0.2 | 0.1e-36 | 0.2 | 0.2e-83 | 0.3e-72 | 0.1e-22 | 0.4e-27 | 0.4e-318 |
| | | | | | | | | |
| $f_2$ | 0.5 | 0.4e-100 | 0.5e-100 | 0.9e-276 | 0.5e-344 | 0.5e-34 | 0.4e-255 | 0.3e-106 |
| $f_2$ | 0.4 | 0.7e-137 | 0.9e-136 | 0.1e-384 | 0.1e-468 | 0.6e-200 | 0.6e-360 | 0.5e-195 |
| $f_2$ | 0.2 | 0.3e-85 | 0.2e-80 | 0.2e-228 | 0.5e-268 | 0.4e-154 | 0.2e-190 | 0.3e-40 |
| | | | | | | | | |
| $f_3$ | 1.7 | 0.2e-39 | 0.1e-51 | 0.2e-172 | 0.1e-72 | 0.3e-21 | 0.1e-106 | 0.9e-84 |
| $f_3$ | 1.2 | 0.1e-58 | 0.1e-79 | 0.1e-278 | 0.8e-168 | 0.2e-83 | 0.9e-171 | 0.2e-22 |
| $f_3$ | 1.5 | 0.2e-82 | 0.4e-93 | 0.6e-308 | 0.7e-226 | 0.6e-94 | 0.7e-238 | 0.6e-171 |
| | | | | | | | | |
| $f_4$ | 1.9 | 0.3e-88 | 0.1e-80 | 0.7e-246 | 0.1e-263 | 0.6e-242 | 0.1e-209 | 0.1e-122 |
| $f_4$ | 2.3 | 0.3e-93 | 0.6e-89 | 0.5e-265 | 0.1e-286 | 0.8e-188 | 0.3e-241 | 0.1e-162 |
| $f_4$ | 2.1 | 0.1e-166 | 0.6e-161 | 0.3e-495 | 0.9e-515 | 0.1e-454 | 0.8e-467 | 0.1e-385 |
| | | | | | | | | |
| $f_5$ | 0.3 | 0.8e-43 | 0.2e-18 | 0.1e-117 | 0.3e-105 | 0.2e-53 | 0.1e-54 | 0.8e-163 |
| $f_5$ | 0 | 0.9e-67 | 0.8e-51 | 0.7e-200 | 0.2e-187 | 0.6e-146 | 0.7e-162 | 0.2e-291 |
| $f_5$ | 0.4 | 0.4e-26 | Div. | 0.1e-63 | 0.1e-52 | 0.1e-10 | 0.4e-3 | 0.4e-95 |
| | | | | | | | | |
| $f_6$ | 0.3 | 0.3e-56 | 0.5e-37 | 0.2e-146 | 0.9e-138 | 0.9e-102 | 0.2e-125 | 0.1e-183 |
| $f_6$ | -0.2 | 0.2e-53 | 0.5e-20 | 0.1e-122 | 0.4e-107 | 0.5e-8 | 0.1e-72 | 0.1e-258 |
| $f_6$ | 0.1 | 0.8e-89 | 0.9e-67 | 0.1e-245 | 0.1e-239 | 0.2e-185 | 0.2e-218 | 0.3e-316 |
| | | | | | | | | |
| $f_7$ | 1.29 | 0.3e-95 | 0.4e-72 | 0.9e-218 | 0.2e-213 | 0.3e-135 | 0.1e-210 | 0.1e-90 |
| $f_7$ | 1.33 | 0.1e-38 | 1.1 | 0.2e-106 | 0.3e-64 | 41.1 | 0.1e-60 | 0.1e-180 |
| $f_7$ | 1.32 | 0.8e-83 | 0.1e-35 | 0.2e-179 | 0.3e-155 | Div. | 0.4e-147 | 0.5e-231 |
| | | | | | | | | |
| $f_8$ | -0.6 | 0.5e-35 | Div. | 0.3e-58 | 0.4e-45 | 281 | 0.1e-18 | 0.1e-155 |
| $f_8$ | -0.9 | 0.3e-75 | 0.6e-55 | 0.2e-181 | 0.1e-176 | 0.3e-133 | 0.1e-169 | 0.2e-181 |
| $f_8$ | -0.7 | 0.4e-67 | 0.1e-33 | 0.2e-160 | 0.3e-144 | 0.4e-56 | 0.1e-117 | 0.2e-234 |
| | | | | | | | | |
| $f_9$ | -0.91 | 0.6-53 | 0.3e-29 | 0.5e-249 | 0.4e-136 | 0.5e-78 | 0.3e-114 | 0.6e-59 |
| $f_9$ | -0.93 | 0.4e-97 | 0.2e-72 | 0.2e-400 | 0.5e-267 | 0.6e173 | 0.2e-254 | 0.1e-224 |
| $f_9$ | -0.9 | 0.2e-38 | Div. | 0.2e-170 | 0.2e-93 | 0.5e-43 | 0.8e-73 | 0.1e-9 |
| | | | | | | | | |
| $f_{10}$ | 0.4 | 0.4e-122 | 0.2e-90 | 0.9e-498 | 0.8e-365 | 0.2e-297 | 0.4e-326 | 0.2e-382 |
| $f_{10}$ | 0.42 | 0.1e-49 | 0.3e-18 | 0.3e-293 | 0.9e-151 | 0.4e-147 | 0.3e-98 | 0.3e-137 |
| $f_{10}$ | 0.36 | 0.1e-25 | 0.6e-2 | 0.2e-124 | 0.2e-50 | 0.4e-51 | 0.8e-73 | 0.9e-30 |

is very close to 4 (at least to the fifth decimal place) for the fourth-order methods and to 3 (at least to the fifth decimal place) for the third-order methods which were considered in Table 1.

TABLE 2. Comparison of efficiency indices for some famous derivative-free algorithms

| Methods | SM | (2.2) | JM | DHM | LM | KTM | PM1 | PM2 |
|---|---|---|---|---|---|---|---|---|
| Efficiency Index | 1.4142 | 1.4142 | 1.4421 | 1.4421 | 1.5874 | 1.5874 | 1.5874 | 1.5874 |

## 5. Concluding remarks and discussions on future works

In this contribution, accurate optimal fourth-order derivative-free algorithms for solving single variable nonlinear equations are developed and their advantages with respect to the other existed well-known methods are illustrated by numerical examples. The algorithms consist of three function evaluations per iteration and therefore their efficiency index is 1.5874 which is bigger than lot of the existing derivative-free methods. In addition, these derivative-free methods can be applied for finding the multiple zeros of nonlinear equations by applying a suitable transformation and converting the multiple zeros of the nonlinear function to a simple one.

To suggest future works, we direct the attention toward two directions, first to multi-point schemes with memory, and to higher-order multi-point iterations without memory. For the first case, one should consider more complicated versions of forward/backward finite difference approximations of order one, e.g. $f'(x_n) = [f(x_n + \beta f(x_n)) - f(x_n)]/\beta f(x_n) + O(\beta f(x_n)))$. In the first case, this would provide families depending on one parameter $\beta \neq 0$, and then one can estimate $\beta$ by an iteration per step using only the known data of the first step. For the latter one, we should take into account the three steps cycle, in which the first two steps are PM1 or PM2, and the subsequent estimate of the newly appeared first derivatives of the function is obtained by a combination of already known values or by using a weight function.

## References

[1] M. Dehghan and M. Hajarian, "Some derivative free quadratic and cubic convergence iterative formulas for solving nonlinear equations," *Comput. Appl. Math.*, vol. 29, no. 1, pp. 19–30, 2010.

[2] A. Galántai and C. J. Hegedŭs, "A study of accelerated Newton methods for multiple polynomial roots," *Numer. Algorithms*, vol. 54, no. 2, pp. 219–243, 2010.

[3] P. Jain, "Steffensen type methods for solving non-linear equations," *Appl. Math. Comput.*, vol. 194, no. 2, pp. 527–533, 2007.

[4] H. T. Kung and J. F. Traub, "Optimal order of one-point and multipoint iteration," *J. Assoc. Comput. Mach.*, vol. 21, pp. 643–651, 1974.

[5] Z. Liu, Q. Zheng, and P. Zhao, "A variant of Steffensen's method of fourth-order convergence and its applications," *Appl. Math. Comput.*, vol. 216, no. 7, pp. 1978–1983, 2010.

[6] J. R. Sharma and R. K. Goyal, "Fourth-order derivative-free methods for solving non-linear equations," *Int. J. Comput. Math.*, vol. 83, no. 1, pp. 101–106, 2006.

[7] F. Soleymani, "Regarding the accuracy of optimal eighth-order methods," *Math. Comput. Modelling*, vol. 53, no. 5-6, pp. 1351–1357, 2011.

[8] F. Soleymani, "Revisit of Jarratt method for solving nonlinear equations," *Numer. Algorithms*, vol. 57, no. 3, pp. 377–388, 2011.

[9] J. F. Traub, *Iterative methods for the solution of equations*, ser. Prentice-Hall Series in Automatic Computation. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1964.

[10] X.-Y. Wu, J.-L. Xia, and R. Shao, "Quadratically convergent multiple roots finding method without derivatives," *Comput. Math. Appl.*, vol. 42, no. 1-2, pp. 115–119, 2001.

[11] Q. Zheng, P. Zhao, and F. Huang, "A family of fourth-order Steffensen-type methods with the applications on solving nonlinear ODEs," *Appl. Math. Comput.*, vol. 217, no. 21, pp. 8196–8203, 2011.

*Author's address*

**F. Soleymani**
Department of Mathematics, Islamic Azad University, Zahedan Branch, Zahedan, Iran
*E-mail address:* `fazl_soley_bsb@yahoo.com`