Applied Mathematics Letters 25 (2012) 847–853

Contents lists available at SciVerse ScienceDirect

Applied Mathematics Letters

journal homepage: www.elsevier.com/locate/aml

Two new classes of optimal Jarratt-type fourth-order methods

F. Soleymani [a,*], S.K. Khattri [b], S. Karimi Vanani [a]

[a] Department of Mathematics, Islamic Azad University, Zahedan Branch, Zahedan, Iran
[b] Department of Engineering, Stord Haugesund University College, Hordaland, Norway

ARTICLE INFO

Article history:
Received 17 July 2011
Received in revised form 20 October 2011
Accepted 26 October 2011

Keywords:
Jarratt-type methods
Multi-point iterations
Kung–Traub conjecture
Optimal order
Efficiency index

ABSTRACT

In this paper, we investigate the construction of some two-step without memory iterative classes of methods for finding simple roots of nonlinear scalar equations. The classes are built through the approach of weight functions and these obtained classes reach the optimal order four using one function and two first derivative evaluations per full cycle. This shows that our classes can be considered as Jarratt-type schemes. The accuracy of the classes is tested on a number of numerical examples. And eventually, it is observed that our contributions take less number of iterations than the compared existing methods of the same type to find more accurate approximate solutions of the nonlinear equations.

1. Introduction

No doubt that the quadratic Newton's iteration is one of the best root-finding methods for solving nonlinear scalar equations. Recent results improving the classical formula of Newton at the expense of an additional evaluation of the function, an additional evaluation of the first derivative or a change in the point of evaluation can be found in the literature on the subject; see e.g. [1–3]. In these works the order of convergence and the classical efficiency index – defined as $p^{1/n}$ where n is the total number of evaluation and p is the order of convergence – have been improved in the neighborhood of a simple root.

To discuss more, some modifications of Newton's method to achieve higher order two-step methods with better efficiency have been suggested and analyzed using several different techniques; see [4] for more details and the references therein. A third-order variant of Newton's method appeared in [5] where trapezoidal approximation to the integral in Newton's theorem

$$f(x) = f(x_n) + \int_{x_n}^{x} f'(t)dt, \tag{1}$$

was taken into account to obtain the cubically convergent method

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

where $y_n = x_n - \frac{f(x_n)}{f'(x_n)}$. Another improvement of Newton's method was suggested in [6], where the authors considered the midpoint rule for the integral of (1) and obtained the following third-order without memory method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'\left(\frac{x_n+y_n}{2}\right)}. \tag{3}$$

* Corresponding author.
E-mail address: fazl_soley_bsb@yahoo.com (F. Soleymani).

0893-9659/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.aml.2011.10.030

In [7], Homeier derived the following cubically convergent iteration scheme

$$x_{n+1} = x_n - \frac{f(x_n)}{2}\left(\frac{1}{f'(x_n)} + \frac{1}{f'(y_n)}\right). \tag{4}$$

The fourth-order Jarratt method [8] in which we use one evaluation of the function and two evaluations of the first derivative (the same evaluations as (2)–(4)) is defined by

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{3f'(y_n) + f'(x_n)}{6f'(y_n) - 2f'(x_n)}\dfrac{f(x_n)}{f'(x_n)}, \end{cases} \tag{5}$$

where its error equations is $e_{n+1} = \left(c_2^3 - c_2c_3 + \frac{c_4}{9}\right)e_n^4 + O\left(e_n^5\right)$. Recently, an efficient fourth-order technique in which we have two evaluations of the first derivative and one evaluation of the function had been presented by Khattri and Abbasbandy in [9] as comes next

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \left[1 + \left(\dfrac{21}{8}\right)\left(\dfrac{f'(y_n)}{f'(x_n)}\right)^1 + \left(-\dfrac{9}{2}\right)\left(\dfrac{f'(y_n)}{f'(x_n)}\right)^2 + \left(\dfrac{15}{8}\right)\left(\dfrac{f'(y_n)}{f'(x_n)}\right)^3\right]\dfrac{f(x_n)}{f'(x_n)}, \end{cases} \tag{6}$$

where its error equations is $e_{n+1} = \frac{1}{9}(85c_2^3 - 9c_2c_3 + c_4)e_n^4 + O\left(e_n^5\right)$. It is worth noting that (5) and (6) are optimal root solvers without memory in viewpoint of Kung–Traub conjecture [10]. Since, such schemes reach the highest possible order with only three evaluations. It is also remarked that methods in which there are two evaluations of the first-order derivative and one of evaluation of the function (with the fourth-order convergence) are called as optimal Jarratt-type methods in literature. For further reading on this topic, one may consult [11,12].

In this paper, using the technique that consists in applying of weight functions on the existing iterative methods, we attain two root-finders for solving nonlinear equations with improved order of convergence and efficiency index than their origins. The key idea to improve is to use only some more additional arithmetic (addition, subtraction, multiplication etc.) of the available (function, derivative) values in order to boost up the convergence speed. As a matter of fact, we suggest two efficient two-step two-point Jarratt-type classes based on the well-cited cubically methods (2) and (4) without using any more evaluations of the function or its derivatives. We also discuss that our general classes of methods are consistent with the optimality of Kung and Traub [10] for building high-order optimal without memory iterations. Since, our classes achieve the fourth-order convergence by consuming only three evaluations per computing step.

The rest of the paper unfolds the contents as follows. Section 2 provides some efficient general fourth-order Jarratt-type schemes as the generalizations of (2) and (4). The analysis of convergence is also given to manifest that our classes are optimal according the un-proved hypothesis of Kung and Traub. Numerical examples will be employed in Section 3 in order to put on show the effectiveness of our new members from the suggested two-point classes. Finally, a conclusion has been drawn and the outline of future studies will be given.

2. New classes of Jarratt-type methods

Throughout this paper, we consider iterative methods to find a simple root α, i.e., $f(\alpha) = 0$ and $f'(\alpha) \neq 0$, of the nonlinear equation $f(x) = 0$, where $f : D \subseteq \mathbb{R} \rightarrow \mathbb{R}$ is a scalar function in an open interval D. Let us take into account (2) which its order is three with three evaluations per full iteration. Clearly its index of efficiency ($3^{1/3} \approx 1.442$) is not high (optimal). We now make use of weight function approach to build our first optimal class based on (2) by a simple change in its first step. Thus, we consider

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{2f(x_n)}{f'(x_n) + f'(y_n)}\left[G(t) \times H(\tau)\right], \end{cases} \tag{7}$$

wherein $G(t)$, $H(\tau)$ are two real-valued weight functions (or two factors of the whole weight function) that should be chosen such that the order of convergence arrives at the optimal level four without using more evaluations of the function. In (7), $t = \frac{f(x)}{f'(x)}$ and $\tau = \frac{f'(y)}{f'(x)}$, without the index n. Theorem 1 indicates that under what conditions on the weight functions in (7), the order of convergence will arrive at the optimal level four.

Theorem 1. *Let a sufficiently smooth function $f : D \subseteq \mathbb{R} \to \mathbb{R}$ has a simple root α in the open interval D. Then the class of methods without memory (7) is of fourth-order convergence when*

$$\begin{cases} G(0) = 1, \quad G'(0) = G''(0) = 0, \quad |G^{(3)}(0)| \leq +\infty, \\ H(1) = 1, \quad H'(1) = -\dfrac{1}{4}, \quad H''(1) = \dfrac{3}{2}, \quad |H^{(3)}(1)| \leq +\infty, \end{cases} \tag{8}$$

and it satisfies the error equation below

$$e_{n+1} = \left(-c_2c_3 + \frac{c_4}{9} - \frac{1}{6}G^{(3)}(0) + \frac{1}{81}c_2^3(297 + 32H^{(3)}(1))\right) e_n^4 + O\left(e_n^5\right). \tag{9}$$

Proof. Let $e_n = x_n - \alpha$ be the error in the nth iterate. By using the symbolic computation and writing the Taylor's series expansion for any term of (7), we have

$$f(x_n) = f'(\alpha)\left[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + O(e_n^5)\right], \tag{10}$$

wherein $c_k = \left(\frac{1}{k!}\right)\frac{f^{(k)}(\alpha)}{f'(\alpha)}$, $k \geq 2$. Also for the first derivative of the function in the first step of our cycle, we have

$$f'(x_n) = f'(\alpha)\left[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + O(e_n^4)\right]. \tag{11}$$

Using (10), (11) and the first step of (7) gives us

$$x_n - \frac{2}{3}\frac{f(x_n)}{f'(x_n)} = \alpha + \frac{e_n}{3} + \frac{2c_2e_n^2}{3} - \frac{4}{3}\left(c_2^2 - c_3\right)e_n^3 + \frac{2}{3}\left(4c_2^3 - 7c_2c_3 + 3c_4\right)e_n^4 + O(e_n^5). \tag{12}$$

Applying Taylor series expansion, we also have

$$f'(y_n) = f'(\alpha) + \frac{2c_2f'(\alpha)\,e_n^1}{3} + \frac{1}{3}\left(4c_2^2 + c_3\right)f'(\alpha)\,e_n^2 + \frac{4}{27}\left(-18c_2^3 + 27c_2c_3 + c_4\right)f'(\alpha)\,e_n^3 + O\left(e_n^4\right).$$

And $f'(x_n) + f'(y_n) = 2f'(\alpha) + \frac{8c_2f'(\alpha)e_n^1}{3} + \frac{2}{3}(2c_2^2 + 5c_3)f'(\alpha)\,e_n^2 + \frac{4}{27}(-18c_2^3 + 27c_2c_3 + 28c_4)f'(\alpha)\,e_n^3 + O\left(e_n^4\right)$. In the same way, for the second step of (12), we attain

$$\frac{2f(x_n)}{f'(x_n) + f'(y_n)} = e_n^1 - \frac{c_2e_n^2}{3} - \frac{2}{9}\left(c_2^2 + 3c_3\right)e_n^3 + \frac{1}{27}\left(50c_2^3 - 15c_2c_3 - 29c_4\right)e_n^4 + O\left(e_n^5\right). \tag{13}$$

Furthermore, we have

$$x_n - \frac{2f(x_n)}{f'(x_n) + f'(y_n)} = \frac{c_2e_n^2}{3} + \frac{2}{9}\left(c_2^2 + 3c_3\right)e_n^3 + \frac{1}{27}\left(-50c_2^3 + 15c_2c_3 + 29c_4\right)e_n^4 + O\left(e_n^5\right). \tag{14}$$

Using (14) and (8), we attain

$$\frac{2f(x_n)}{f'(x_n) + f'(y_n)}\left[G(t) \times H(\tau)\right] = e_n^1 + \left(c_2c_3 - \frac{c_4}{9} + \frac{1}{6}G^{(3)}(0) - \frac{1}{81}c_2^3(297 + 32H^{(3)}(1))\right)e_n^4 + O\left(e_n^5\right). \tag{15}$$

Finally, by using (15) and (7), we have the follow-up error equation

$$\begin{aligned} e_{n+1} &= x_{n+1} - \alpha = x_n - \frac{2f(x_n)}{f'(x_n) + f'(y_n)}\left[G(t) \times H(\tau)\right] - \alpha \\ &= \left(-c_2c_3 + \frac{c_4}{9} - \frac{1}{6}G^{(3)}(0) + \frac{1}{81}c_2^3(297 + 32H^{(3)}(1))\right)e_n^4 + O\left(e_n^5\right). \end{aligned} \tag{16}$$

This reveals that the general two-step class of methods (7)–(8) reaches the convergence order four by using only three evaluations per full iteration. The proof is complete now. □

It should be remarked that each member from our proposed class (7)–(8) includes two evaluations of the first-order derivative and one evaluation of the function, thus (7)–(8) is an optimal Jarratt-type scheme. Choosing any desired weight functions based on (8) will end in a new optimal two-point method.

Some typical examples from our proposed class can be the followings

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{2f(x_n)}{f'(x_n) + f'(y_n)}\left[\left(1 + \left(\dfrac{f(x_n)}{f'(x_n)}\right)^3\right)\left(2 - \dfrac{7}{4}\dfrac{f'(y_n)}{f'(x_n)} + \left(\dfrac{3}{4}\right)\left(\dfrac{f'(y_n)}{f'(x_n)}\right)^2\right)\right], \end{cases} \tag{17}$$

where its error equation is

$$e_{n+1} = x_{n+1} - \alpha = \frac{1}{9}\left(-9 + 33c_2^3 - 9c_2c_3 + c_4\right)e_n^4 + O\left(e_n^5\right), \tag{18}$$

and

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{2f(x_n)}{f'(x_n)+f'(y_n)}\left[\left(1+\left(\dfrac{f(x_n)}{f'(x_n)}\right)^4\right)\left(2-\dfrac{7}{4}\dfrac{f'(y_n)}{f'(x_n)}+\left(\dfrac{3}{4}\right)\left(\dfrac{f'(y_n)}{f'(x_n)}\right)^2\right)\right], \end{cases} \tag{19}$$

wherein its error equation reads

$$e_{n+1} = x_{n+1} - \alpha = \frac{1}{9}\left(33c_2^3 - 9c_2c_3 + c_4\right)e_n^4 + O\left(e_n^5\right). \tag{20}$$

In addition to (7)–(8), it will then be easy to construct another novel two-point class of Jarratt-type methods based on (4). In order to do this, we similarly suggest the following without memory general class as the second contribution of this study

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{f(x_n)}{2}\left(\dfrac{1}{f'(x_n)}+\dfrac{1}{f'(y_n)}\right)\left[G(t)\times H(\tau)\right]. \end{cases} \tag{21}$$

Theorem 2. *Let a sufficiently smooth function $f : D \subseteq \mathbb{R} \to \mathbb{R}$ has a simple root α in the open interval D. Then the class of methods (21) is of fourth-order convergence when*

$$\begin{cases} G(0) = 1, \quad G'(0) = G''(0) = 0, \quad |G^{(3)}(0)| \leq +\infty, \\ H(1) = H''(1) = 1, \quad H'(1) = -\dfrac{1}{4}, \quad |H^{(3)}(1)| \leq +\infty, \end{cases} \tag{22}$$

and it satisfies

$$e_{n+1} = \left(-c_2c_3 + \frac{c_4}{9} - \frac{1}{6}G^{(3)}(0) + \frac{1}{81}c_2^3(237 + 32H^{(3)}(1))\right)e_n^4 + O\left(e_n^5\right). \tag{23}$$

Proof. The proof of this theorem is completely similar to the proof of Theorem 1. It is hence omitted. □

As some examples from this new optimal Jarratt-type class of iterations without memory, we can produce the followings using (22)

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{f(x_n)}{2}\left(\dfrac{1}{f'(x_n)}+\dfrac{1}{f'(y_n)}\right)\left[\left(1+\left(\dfrac{f(x_n)}{f'(x_n)}\right)^4\right)\left(1-\dfrac{1}{4}\left(\dfrac{f'(y_n)}{f'(x_n)}-1\right)+\dfrac{1}{2}\left(\dfrac{f'(y_n)}{f'(x_n)}-1\right)^2\right)\right], \end{cases} \tag{24}$$

with $e_{n+1} = x_{n+1} - \alpha = \left(\frac{79c_2^3}{27} - c_2c_3 + \frac{c_4}{9}\right)e_n^4 + O\left(e_n^5\right)$, as the error relation and also

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{f(x_n)}{2}\left(\dfrac{1}{f'(x_n)}+\dfrac{1}{f'(y_n)}\right) \\ \qquad \times\left[\left(1+\left(\dfrac{f(x_n)}{f'(x_n)}\right)^4\right)\left(1-\dfrac{1}{4}\left(\dfrac{f'(y_n)}{f'(x_n)}-1\right)+\dfrac{1}{2}\left(\dfrac{f'(y_n)}{f'(x_n)}-1\right)^2-\left(\dfrac{f'(y_n)}{f'(x_n)}-1\right)^3\right)\right]. \end{cases} \tag{25}$$

where $e_{n+1} = x_{n+1} - \alpha = \frac{1}{9}(5c_2^3 - 9c_2c_3 + c_4)e_n^4 + O\left(e_n^5\right)$, is its error equation. Note that (21)–(22) is somehow similar to (7)–(8).

In what follows; we give a comparison over the efficiency index and the informational efficiency $\left(\mathrm{EI} = \frac{p}{n}\right)$ of some well-known methods with our developed classes (7)–(8) and (21)–(22). This comparison is provided in Table 1.

Table 1
Comparison of different methods in terms of orders and efficiencies.

Techniques	Order of convergence	Number of evaluations	Informational index	Efficiency index
Newton	2	2	1	1.4142
(2)	3	3	1	1.4422
(3)	3	3	1	1.4422
(4)	3	3	1	1.4422
(5)	4	3	1.3333	1.5874
(6)	4	3	1.3333	1.5874
(7)–(8)	4	3	1.3333	1.5874
(21)–(22)	4	3	1.3333	1.5874

3. Numerical reports

Now, we employ the new methods (17) and (19) form our suggested class (7)–(8) to solve some nonlinear equations and compare them with the Newton's method (NM), (2), (4) and (6). All computations were carried out with the use of VPA command in MATLAB 7.6. All problems were solved with a given initial value x_0, which is necessary to start the iterative process. Currently, IEEE 64-bit floating-point arithmetic is sufficient for the most commonly applications in order to attain the accuracy desired. But, it is increasing the number of applications where it is required to use a higher level of numeric precision. Namely, evaluating orthogonal polynomials, high-precision solution of ODEs, divergent asymptotic series, experimental mathematics, and nonlinear oscillator theory among others. Therefore, adaptive multi-precision arithmetic facilities are most appropriate in a modern large-scale scientific computing environment. Consequently, we consider the number of decimal places as follows: 800 digits floating point (VPA = 800) with VPA Command. In examples considered in this article, the stopping criterion is the $|f(x_n)| \leq \varepsilon$, where $\varepsilon = 10^{-800}$. The test functions and their simple zeros are as comes next.

- $f_1 = (\sin x)^2 + x,$ $\alpha = 0,$
- $f_2 = \left(1 + x^3\right)\cos\left(\frac{\pi x}{2}\right) + \sqrt{1 - x^2} - \frac{2(9\sqrt{2}+7\sqrt{3})}{27},$ $\alpha = 1/3,$
- $f_3 = (\sin x)^2 - x^2 + 1,$ $\alpha \approx 1.404491648215341226035086817786,$
- $f_4 = e^{-x} + \sin(x) - 1,$ $\alpha \approx 2.076831274533112613070044244750,$
- $f_5 = xe^{-x} - 0.1,$ $\alpha \approx 0.111832559158962964833569456820,$
- $f_6 = x^2 + \sin(x) + x,$ $\alpha = 0,$
- $f_7 = \sin(2\cos(x)) - 1 - x^2 + e^{\sin(x^3)},$ $\alpha \approx 1.306175201846827825014842909066,$
- $f_8 = \sin(2\cos(x)) - 1 - x^2 + e^{\sin(x^3)},$ $\alpha \approx -0.784895987661212535224856018448,$
- $f_9 = \cos(x) + \sin(2x)\sqrt{1 - x^2} + \sin\left(x^2\right) + x^{14} + x^3 + \frac{1}{2x},$ $\alpha \approx -0.925772249827561423326931990067,$
- $f_{10} = \tan(\ln x) + \cos\left(x^3\right) \times \sqrt{1/(2x)},$ $\alpha \approx 0.443260783556767073513472596321,$
- $f_{11} = \tan^{-1}(x) - 1,$ $\alpha \approx 1.557407724654902230506974807458,$
- $f_{12} = x^6 - 10x^3 + x^2 - x + 3,$ $\alpha \approx 0.658604847118140436763860014710,$
- $f_{13} = x^4 - x^3 + 11x - 7,$ $\alpha \approx 0.803511199110777688978137660293,$
- $f_{14} = x^3 - \cos x + 2,$ $\alpha \approx -1.172577964753970012673332714868,$
- $f_{15} = \sqrt{x} - \cos x,$ $\alpha \approx 0.641714370872882658398565300316,$
- $f_{16} = \ln(x) - x^3 + 2\sin x,$ $\alpha \approx 1.297997743280371847164479238286.$

The results of comparisons are provided in Table 2. In fact, the absolute value of the given test functions after four full iterations for (2), (4), (6), (17) and (19); and after six iterations for NM are listed there. That is to say with the same Total Number of Evaluations (TNE = 12). Numerical results are in concordance with the theory developed in this paper. In most of the cases, the results obtained from our new methods are similar or better than the existing methods of the same type. The proposed fourth-order classes (7)–(8) and (21)–(22) of methods for finding simple real roots of nonlinear equations are free from second-order derivative of the given function. Their methods can also be used to find local maxima and minima of functions as these extrema are the roots of the derivative function.

Our classes require evaluations of one function and two first order derivatives per full iteration. The high order convergence is also corroborated by numerical tests in this section. We should remark that Jarratt-type methods are so convenient for predictor–corrector structures (three-step methods); since they provide better predictions for starting approximations, which are in the vicinity of the roots but not so close [13].

4. Concluding remarks and future works

In science and engineering, many of the nonlinear and transcendental problems of the form $f(x) = 0$, are complex. Since it is not always possible to obtain their exact solution by usual algebraic processes, therefore numerical iterative methods such as Newton's method are often used to obtain the approximate solution. Though these methods are very effective, but there are some limitations that they do not give the result as fast as one wants, and take several iterations. As a remedy, a

Table 2
Comparison of different methods with the same Total Number of Evaluation (TNE = 12).

Test functions	Guess	NM	(2)	(4)	(6)	(17)	(19)
$\|f_1\|$	0.3	0.2e−41	0.4e−54	0.5e−198	0.2e−107	0.2e−172	0.1e−136
	0.2	0.5e−50	0.2e−64	0.7e−224	0.6e−136	0.1e−186	0.5e−165
	−0.1	0.5e−61	0.7e−77	0.3e−257	0.7e−147	0.1e−198	0.7e−189
$\|f_2\|$	0.3	0.4e−87	0.1e−109	0.6e−170	0.4e−260	0.2e−302	0.3e−297
	0.2	0.8e−44	0.3e−56	0.6e−87	0.1e−63	0.5e−115	0.3e−111
	0.4	0.1e−70	0.1e−87	0.1e−122	0.7e−209	0.2e−247	0.6e−241
$\|f_3\|$	1.3	0.7e−67	0.2e−83	0.1e−157	0.2e−176	0.1e−244	0.2e−216
	1	0.1e−24	0.9e−29	0.5e−61	0.1e−3	0.2e−63	0.1e−35
	1.5	0.4e−73	0.9e−92	0.1e−132	0.2e−220	0.1e−296	0.2e−254
$\|f_4\|$	2	0.6e−83	0.3e−107	0.8e−138	0.9e−242	0.8e−362	0.8e−277
	2.3	0.1e−59	0.6e−79	0.1e−115	0.1e−169	0.1e−215	0.5e−211
	2.1	0.2e−118	0.1e−152	0.3e−186	0.1e−392	0.6e−493	0.5e−426
$\|f_5\|$	0.3	0.1e−41	0.1e−48	0.2e−93	0.2e−60	0.1e−103	0.1e−109
	0.2	0.2e−64	0.1e−77	0.6e−111	0.2e−165	0.4e−201	0.1e−208
	0	0.9e−61	0.2e−73	0.5e−98	0.2e−173	0.6e−197	0.6e−207
$\|f_6\|$	0.3	0.5e−57	0.6e−76	0.4e−113	0.1e−161	0.4e−157	0.2e−261
	0.1	0.3e−84	0.4e−110	0.6e−141	0.2e−261	0.2e−279	0.3e−299
	−0.2	0.3e−59	0.4e−78	0.1e−100	0.9e−138	0.7e−223	0.1e−172
$\|f_7\|$	1.35	0.3e−78	0.7e−112	0.1e−88	0.2e−234	0.1e−254	0.7e−252
	1.31	0.9e−139	0.1e−217	0.2e−174	0.5e−470	0.3e−497	0.8e−496
	1.29	0.8e−96	0.7e−159	0.6e−123	0.7e−292	0.3e−323	0.4e−322
$\|f_8\|$	−0.8	0.7e−125	0.2e−168	0.1e−172	0.5e−418	0.1e−430	0.2e−448
	−0.75	0.1e−99	0.5e−135	0.4e−142	0.7e−312	0.6e−330	0.1e−344
	−0.72	0.2e−81	0.4e−111	0.2e−119	0.1e−234	0.5e−256	0.3e−269
$\|f_9\|$	−0.9	0.2e−48	0.4e−55	0.1e−81	0.2e−96	0.5e−146	0.4e−146
	−0.88	0.2e−48	0.9e−34	0.1e−81	0.7e−20	0.2e−73	0.2e−73
	−0.96	0.8e−43	0.3e−49	0.6e−67	0.8e−110	0.2e−140	0.2e−140
$\|f_{10}\|$	0.5	0.2e−39	0.5e−42	0.5e−65	0.1e−57	0.3e−111	0.2e−111
	0.4	0.8e−47	0.6e−53	0.4e−69	0.7e−123	0.4e−156	0.3e−156
	0.45	0.1e−98	0.1e−117	0.6e−137	0.4e−313	0.6e−356	0.3e−356
$\|f_{11}\|$	1.55	0.3e−158	0.5e−194	0.8e−215	0.1e−553	0.4e−537	0.5e−595
	1.54	0.1e−134	0.5e−164	0.1e−184	0.4e−459	0.1e−442	0.2e−501
	1.6	0.1e−109	0.2e−132	0.1e−153	0.9e−357	0.7e−341	0.2e−394
$\|f_{12}\|$	0.7	0.1e−80	0.9e−101	0.1e−138	0.1e−249	0.1e−288	0.2e−283
	0.6	0.6e−68	0.2e−84	0.5e−137	0.5e−183	0.9e−229	0.2e−224
	0.5	0.1e−37	0.1e−45	0.6e−89	0.1e−41	0.3e−95	0.5e−91
$\|f_{13}\|$	0.65	0.5e−58	0.8e−72	0.7e−151	0.7e−140	0.7e−231	0.1e−180
	0.75	0.7e−89	0.7e−111	0.2e−153	0.8e−274	0.5e−372	0.3e−311
	0.85	0.3e−94	0.1e−117	0.9e−153	0.1e−303	0.1e−423	0.1e−339
$\|f_{14}\|$	−1	0.1e−43	0.1e−52	0.2e−147	0.2e−70	0.1e−112	0.1e−118
	−1.1	0.1e−69	0.1e−85	0.2e−125	0.4e−190	0.6e−224	0.2e−231
	−1.2	0.1e−98	0.1e−122	0.1e−154	0.1e−318	0.3e−345	0.2e−354
$\|f_{15}\|$	0.9	0.3e−93	0.2e−110	0.9e−138	0.6e−326	0.1e−152	0.1e−226
	0.7	0.3e−134	0.7e−153	0.7e−161	0.2e−530	0.1e−315	0.7e−424
	0.6	0.1e−144	0.8e−161	0.2e−166	0.1e−511	0.1e−351	0.1e−502
$\|f_{16}\|$	1.2	0.1e−51	0.2e−64	0.4e−110	0.3e−105	0.3e−154	0.1e−150
	2	0.1e−17	0.1e−21	0.3e−41	0.1e−35	0.2e−75	0.2e−58
	1.5	0.6e−41	0.5e−51	0.7e−82	0.5e−106	0.4e−139	0.1e−132

technique for accelerating the order of convergence of two given iterative processes without any additional evaluation of the function or derivative was given in this article by using weight functions approach. Furthermore, we have analyzed the new two-step classes and it was confirmed that each member from our classes (7)–(8) and (21)–(22) supports the conjecture of Kung and Traub for constructing optimal multi-point methods without memory. Order of convergence and efficiency index have been improved in contrast to the origin cubically methods (2) and (4). The results have been computationally tested on a set of functions. Due to the fact that when the order of convergence of any iterative method is high, we need to carry out the computations for testing it with an enlarged mantissa. A multi-precision with low computing time must be used in all the calculations, as we have done in this work. Kindly note that development of our fourth-order classes for multiple roots is under investigation in our research group and this can be done as the future works in this active field of study.

References

[1] S.K. Khattri, Optimal eighth order iterative methods, Math. Comput. Sci. (2011) doi:10.1007/s11786-011-0064-7.
[2] P. Sargolzaei, F. Soleymani, Accurate fourteenth-order methods for solving nonlinear equations, Numer. Algorithms (2011) doi:10.1007/s11075-011-9467-4.

[3] F. Soleymani, M. Sharifi, B.S. Mousavi, An improvement of Ostrowski's and King's techniques with optimal convergence order eight, J. Optim. Theory Appl. (2011) doi:10.1007/s10957-011-9929-9.
[4] C. Chun, B. Neta, Certain improvements of Newton's method with fourth-order convergence, Appl. Math. Comput. 215 (2009) 821–828.
[5] S. Weerakoon, G.I. Fernando, A variant of Newton's method with accelerated third-order convergence, Appl. Math. Lett. 17 (2000) 87–93.
[6] M. Frontini, E. Sormani, Some variants of Newton's method with third-order convergence, Appl. Math. Comput. 140 (2003) 419–426.
[7] H.H.H. Homeier, On Newton-type methods with cubic convergence, J. Comput. Appl. Math. 176 (2005) 425–432.
[8] P. Jarratt, Some efficient fourth order multipoint methods for solving equations, BIT 9 (1969) 119–124.
[9] S.K. Khattri, S. Abbasbandy, Optimal fourth order family of iterative methods, Mat. Vesnik 63 (2011) 67–72.
[10] H.T. Kung, J.F. Traub, Optimal order of one-point and multipoint iteration, J. Assoc. Comput. Mach. 21 (1974) 634–651.
[11] F. Soleymani, S. Karimi Vanani, M. Khan, M. Sharifi, Some modifications of King's family with optimal eighth order of convergence, Math. Comput. Modelling (2011) doi:10.1016/j.mcm.2011.10.016.
[12] F. Soleymani, S. Karimi Vanani, Optimal Steffensen-type methods with eighth order of convergence, Comput. Math. Appl. (2011) doi:10.1016/j.camwa.2011.10.047.
[13] F. Soleymani, A novel and precise sixth-order method for solving nonlinear equations, Int. J. Math. Models Methods Appl. Sci. 5 (2011) 730–737.