Generating optimal derivative free iterative methods for nonlinear equations by using polynomial interpolation ☆

Alicia Cordero[a], José L. Hueso[a,*], Eulalia Martínez[b], Juan R. Torregrosa[a]

[a] *Instituto Universitario de Matemática Multidisciplinar*
Universitat Politècnica de València
Camino de Vera, s/n, 46022 València, Spain
[b] *Instituto de Matemática Pura y Aplicada*
Universitat Politècnica de València
Camino de Vera, s/n, 46022 València, Spain

Abstract

In this work we show a general procedure to obtain optimal derivative free iterative methods [4] for nonlinear equations $f(x) = 0$, applying polynomial interpolation to a generic optimal derivative free iterative method of lower order.

Let us consider an optimal method of order $q = 2^{n-1}$, $v = \phi_n(x)$, that uses n functional evaluations. Performing a Newton step $w = v - \frac{f(v)}{f'(v)}$ one obtains a method of order 2^n, that is not optimal because it introduces two new functional evaluations. Instead, we approximate the derivative by using a polynomial of degree n that interpolates $n + 1$ already known functional values and keeps the order 2^n.

We have applied this idea to Steffensen's method, [5], obtaining a family of optimal derivative free iterative methods of arbitrary high order.

We provide different numerical tests, that confirm the theoretical results and compare the new family with other well known family of similar characteristics.

☆ This research was supported by Ministerio de Ciencia e Innovación MTM2011-28636-C02-02 and by Vicerrectorado de Investigación. Universitat Politècnica de València PAID-06-2010-2285

* Corresponding author

Email addresses: acordero@mat.upv.es (Alicia Cordero), jlhueso@mat.upv.es (José L. Hueso), eumarti@mat.upv.es (Eulalia Martínez), jrtorre@mat.upv.es (Juan R. Torregrosa)

Preprint submitted to Mathematical and Computer Modelling *December 27, 2011*

1. Introduction

A variety of problems in different fields of science and technology require to find the solution of a nonlinear equation. Iterative methods for approximating solutions of nonlinear equation are the most used technique. The efficiency index, introduced by Ostrowski in [3], establishes the effectiveness of the iterative method. In this sense, Kung and Traub conjectured in [4] that a method is optimal if it reaches an order of convergence $q = 2^n$, using only $n + 1$ function evaluations per step.

In recent years a number of high order iterative methods have appeared in the literature, some of them optimal, reaching the maximal efficiency among the methods of the same order. The classical Newton's and Steffensen's methods, [5], are optimal for $n = 2$; Ostrowski and Jarrat's methods, [3, 6], for $n = 3$. Several optimal eighth order methods have been proposed (see for example [7, 8, 9, 10, 11, 12]). Even optimal sixteenth order methods have been published, such as [2].

In this paper we introduce a new family of optimal order derivative free iterative methods for approximating the solution of nonlinear equations, alternative to the families in [4, 13], proving a convergence result that shows the optimality of the methods. The interest of high order iterative methods is mainly theoretical because most of the applications do not need such a high precision.

The outline of the paper is as follows. In Section 2 we describe the idea of the procedure and show how to obtain optimal methods of increasing order, starting from an optimal second order derivative free iterative method. In Section 3 we prove that, under usual requirements, the procedure doubles the order of a given optimal method, maintaining the optimality. Finally, in Section 4, different numerical tests confirm the theoretical results and allow us to compare the obtained methods with other known optimal methods.

2. Optimal methods of increasing order

Newton's and Steffensen's iterative methods for approximating the solution of the nonlinear equation $f(x) = 0$ are both of second order and they are optimal, because they only need 2 functional evaluations per step. Steffensen's method has the advantage that it does not need the derivative. In this paper we focus on derivative free iterative methods, but our procedure could also be applied starting from Newton's method.

Let us start from Steffensen's method and explain the procedure to get optimal methods of increasing order. The idea is to compose a Steffensen's iteration,

$$\begin{aligned} y_0 &= x_k, \\ y_1 &= y_0 + f(y_0), \\ y_2 &= y_0 - \frac{f(y_0)^2}{f(y_1) - f(y_0)}, \end{aligned}$$

with a Newton's step

$$y_3 = y_2 - \frac{f(y_2)}{f'(y_2)}.$$

The resulting iteration has convergence order 4, being the composition of two second order methods (see [15], theorem 2.4), but the method is not optimal because it uses 4 function evaluations.

In order to get optimality, the value of $f'(y_2)$ can be approximated by the derivative $p_2'(y_2)$ of a suitable second degree polynomial obtained from already computed function values. It is natural to consider the interpolating polynomial of the points $(y_0, f(y_0))$, $(y_1, f(y_1))$ and $(y_2, f(y_2))$. This polynomial can be written as

$$p_2(t) = a_0^{(2)} + a_1^{(2)}(t - y_2) + a_2^{(2)}(t - y_2)^2$$

and its coefficients are the solutions of a linear system

$$p_2(y_j) = f(y_j), \quad j = 0, 1, 2.$$

In particular, $p_2'(y_2) = a_1^{(2)}$.

Therefore, we get a method, M_4, with $n + 1 = 3$ functional evaluations that will be optimal provided its order is $2^n = 4$.

$$y_0 = x_k, \tag{1}$$

$$y_1 = y_0 + f(y_0), \tag{2}$$

$$y_2 = y_0 - \frac{f(y_0)^2}{f(y_1) - f(y_0)}, \tag{3}$$

$$x_{k+1} = y_3 = y_2 - \frac{f(y_2)}{p_2'(y_2)}. \tag{4}$$

Incidentally, we notice that Steffensen's method can be derived in the same way from the first two steps (1-2) of the former iteration, substituting the derivative in a next Newton's step by the slope of the line through the points $(y_0, f(y_0))$ and $(y_1, f(y_1))$.

We can improve M_4 by the same procedure. Composing it with a Newton's step one obtains an order 8 method but loses optimality because there are two new function evaluations. Then, substitute the derivative of function f by the derivative $p_3'(y_3)$ of the third degree polynomial that interpolates f at the points y_0, y_1, y_2, y_3. The order is maintained whereas the method becomes optimal.

In theory, this procedure can be indefinitely repeated, giving a family of optimal methods of arbitrarily high order. The iteration of the generic method M_q, for $q = 2^n$ is $x_{k+1} = y_{n+1}$ where

$$y_0 = x_k, \tag{5}$$

$$y_1 = y_0 + f(y_0), \tag{6}$$

$$y_{j+1} = y_j - \frac{f(y_j)}{p_j'(y_j)}, \quad j = 1, 2, \ldots, n, \tag{7}$$

and p_j is the polynomial that interpolates f in $y_0, y_1, \ldots, y_j$.

The value $p_j'(y_j) = a_1^{(j)}$ can be explicitly expressed in terms of the interpolated points. We have to obtain the coefficient of the linear term of the polynomial

$$p_j(t) = a_0^{(j)} + a_1^{(j)}(t - y_j) + a_2^{(j)}(t - y_j)^2 + \ldots + a_j^{(j)}(t - y_j)^j.$$

This polynomial is determined by the interpolation conditions

$$p_j(y_i) = f(y_i), \quad i = 0, 1, \ldots, j.$$

From the last condition one obtains $a_0^{(j)} = f(y_j)$. The remaining coefficients are the solutions of a linear system that can be written as

$$\left.\begin{array}{r} a_1 + a_2(y_0 - y_j) + a_3(y_0 - y_j)^2 + \ldots + a_j(y_0 - y_j)^{j-1} = f[y_0, y_j] \\ a_1 + a_2(y_1 - y_j) + a_3(y_1 - y_j)^2 + \ldots + a_j(y_1 - y_j)^{j-1} = f[y_1, y_j] \\ \vdots \\ a_1 + a_2(y_{j-1} - y_j) + a_3(y_{j-1} - y_j)^2 + \ldots + a_j(y_{j-1} - y_j)^{j-1} = f[y_{j-1}, y_j] \end{array}\right\}$$

where, as usual, $f[y_i, y_j]$ denotes the divided difference $\dfrac{f(y_i) - f(y_j)}{y_i - y_j}$, and where, for simplicity, we have omitted the superindexes.

The coefficient $a_1^{(j)}$ can be obtained applying Cramer's rule

$$a_1^{(j)} = \frac{\Delta_1}{\Delta},$$

where Δ is the Vandermonde's determinant

$$\Delta = V(y_0 - y_j, y_1 - y_j, \ldots, y_{j-1} - y_j) = V(y_0, y_1, \ldots, y_{j-1})$$

and

$$\Delta_1 = \begin{vmatrix} f[y_0, y_j] & y_0 - y_j & (y_0 - y_j)^2 & \ldots & (y_0 - y_j)^{j-1} \\ f[y_1, y_j] & y_1 - y_j & (y_1 - y_j)^2 & \ldots & (y_1 - y_j)^{j-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f[y_{j-1}, y_j] & y_{j-1} - y_j & (y_{j-1} - y_j)^2 & \ldots & (y_{j-1} - y_j)^{j-1} \end{vmatrix}.$$

This determinant can be computed by cofactors of its first column.

$$\Delta_1 = \sum_{i=0}^{j-1} (-1)^i f[y_i, y_j] \prod_{k=0,\ k \neq i}^{j-1} (y_k - y_j) V_i(y_0, y_1, \ldots, y_{j-1}),$$

where $V_i(y_0, y_1, \ldots, y_{j-1})$ is the Vandermonde's determinant of the list of arguments where y_i is missing. It is not difficult to see that

$$a_1^{(j)} = \sum_{i=0}^{j-1} \left(\prod_{k=0,\ k \neq i}^{j-1} \frac{y_k - y_j}{y_k - y_i} \right) f[y_i, y_j].$$

For example, for $j = 2$, the derivative $f'(y_2)$ is approximated by

$$a_1^{(2)} = \frac{y_1 - y_2}{y_1 - y_0} f[y_0, y_2] + \frac{y_0 - y_2}{y_0 - y_1} f[y_1, y_2],$$

and for $j = 3$,

$$a_1^{(3)} = \frac{(y_1 - y_3)(y_2 - y_3)}{(y_1 - y_0)(y_2 - y_0)} f[y_0, y_3] + \frac{(y_0 - y_3)(y_2 - y_3)}{(y_0 - y_1)(y_2 - y_1)} f[y_1, y_3] + \frac{(y_0 - y_3)(y_1 - y_3)}{(y_0 - y_2)(y_1 - y_2)} f[y_2, y_3].$$

3. Order of convergence

Theorem 1. *Let $\alpha \in I$ be a simple zero of a function $f : I \subseteq \mathbb{R} \longrightarrow \mathbb{R}$ with bounded $(n+1)$-th derivative in an open interval I. If x is sufficiently close to α, then the method M_q defined by (5-7) has optimal convergence order 2^n.*

Proof: Let $\epsilon_{k,j}$ be the error of $y_j,\ j = 0, 1, \ldots, n+1$ in an iteration of the method M_q, that is $\epsilon_{k,j} = y_j - \alpha$. Then,

$$\begin{aligned} \epsilon_{k,0} &= x_k - \alpha = \epsilon_k, \\ \epsilon_{k,n+1} &= y_{n+1} - \alpha = x_{k+1} - \alpha = \epsilon_{k+1}. \end{aligned}$$

We have to prove that $\epsilon_{k+1} = O(\epsilon_k^{2^n})$. We will show, by induction on $j > 0$, that

$$\epsilon_{k,j} = O(\epsilon_k^{2^{j-1}}). \tag{8}$$

For $j = 1$, the result is true because

$$\begin{aligned} \epsilon_{k,1} &= y_1 - \alpha = y_0 - \alpha + f(y_0) \\ &= y_0 - \alpha + f'(\alpha)(y_0 - \alpha) + O((y_0 - \alpha)^2) \\ &= (1 + f'(\alpha))\epsilon_{k,0} + O(\epsilon_{k,0}^2) \\ &= (1 + f'(\alpha))\epsilon_k + O(\epsilon_k^2) = O(\epsilon_k). \end{aligned}$$

For $j = 2$ it derives from the fact that M_2 is the Steffensen's method. Suppose, by induction hypothesis that (8) is true for $j \leq n$. Let us show that it is also true for $j = n + 1$.

The interpolating polynomial $p_n(t)$ that appears in (7) satisfies the error equation

$$f(t) - p_n(t) = \frac{f^{(n+1)}(\xi(t))}{(n+1)!}(t - y_0)(t - y_1)\ldots(t - y_n).$$

Assuming $\xi(t)$ differentiable, and setting $t = y_n$, in the derivative, we have

$$f'(y_n) - p_n'(y_n) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(y_n - y_0)(y_n - y_1)\ldots(y_n - y_{n-1}). \tag{9}$$

Each factor in (9) with $j > 0$ satisfies

$$\begin{aligned} y_n - y_j &= y_n - \alpha - (y_j - \alpha) = \epsilon_{k,n} - \epsilon_{k,j} \\ &= O(\epsilon_k^{2^{n-1}}) + O(\epsilon_k^{2^{j-1}}) = O(\epsilon_k^{2^{j-1}}), \end{aligned}$$

by the induction hypothesis. Then

$$f'(y_n) - p'_n(y_n) = \frac{f^{(n+1)}(\xi)}{(n+1)!} O(\epsilon_k) O(\epsilon_k) O(\epsilon_k^2) \ldots O(\epsilon_k^{2^{n-2}}) = O(\epsilon_k^{2^{n-1}}).$$

Then we can write

$$f'(y_n) = p'_n(y_n)(1 + O(\epsilon_k^{2^{n-1}})), \tag{10}$$

So, the order of M_q can be established from:

$$\epsilon_{k,n+1} = y_{n+1} - \alpha = y_n - \alpha - \frac{f(y_n)}{p'_n(y_n)},$$

and by using (10) we have

$$\begin{aligned} \epsilon_{k,n+1} =& y_{n+1} - \alpha = y_n - \alpha - \frac{f(y_n)(1 + O(\epsilon_k^{2^{n-1}}))}{f'(y_n)}, \\ =& y_n - \alpha - \frac{f(y_n)}{f'(y_n)} - \frac{f(y_n)}{f'(y_n)} O(\epsilon_k^{2^{n-1}}). \end{aligned} \tag{11}$$

Dividing the Taylor's series of $f(y_n)$ and $f'(y_n)$ in α we have

$$\frac{f(y_n)}{f'(y_n)} = (y_n - \alpha) - \frac{c_2}{c_1}(y_n - \alpha)^2 + O((y_n - \alpha)^3),$$

with $c_1 = f'(\alpha)$ and $c_2 = \dfrac{f''(\alpha)}{2!}$. Then, using that $y_n - \alpha = \epsilon_{k,n+1} = O(\epsilon_k^{2^{n-1}})$, by the induction hypothesis we get

$$\frac{f(y_n)}{f'(y_n)} = O(\epsilon_k^{2^{n-1}}), \tag{12}$$

and

$$y_n - \alpha - \frac{f(y_n)}{f'(y_n)} = O(\epsilon_k^{2^n}). \tag{13}$$

Substituting (12) and (13) in (11) we get

$$\epsilon_{k,n+1} = O(\epsilon_k^{2^n}) + O(\epsilon_k^{2^{n-1}}) O(\epsilon_k^{2^{n-1}}) = O(\epsilon_k^{2^n}),$$

which completes the proof. □

4. Numerical results

In this section we check the effectiveness of the new optimal iterative methods M_q comparing them with the optimal family K_q, for $q = 2^n = 4, 8, 16$ introduced in [4],

$$\begin{aligned} y_0 &= x_k, \\ y_1 &= y_0 + \beta f(y_0), \quad \beta \neq 0, \\ y_{j+1} &= Q_j(0), \quad j = 1, 2, \ldots, n, \end{aligned}$$

where $Q_j(t)$ is a polynomial of degree at most j satisfying the interpolation conditions $Q_j(f(y_i)) = y_i, \quad i = 0, 1, \ldots, j$. The order 2 methods of both families are Steffensen's method, so that, they coincide in the comparison.

Firstly, we test the different iterative methods by using the following smooth functions:

a) $f(x) = xe^{x^2} - \sin^2(x) + 3\cos(x) + 5; \quad \alpha \approx -1.207647827130918927$
b) $f(x) = x^3 - 10; \quad \alpha \approx 2.15443469003188372 18$
c) $f(x) = \sin^2(x) - x^2 + 1; \quad \alpha \approx 1.404491648215341226$
d) $f(x) = (x+2)e^x - 1; \quad \alpha \approx -0.442854401002388583$
e) $f(x) = (x-1)^3 - 2; \quad \alpha \approx 2.259921049894873 1648$
f) Let us consider Kepler's equation $f(x) = x - e\sin(x) - M$; where $0 \leq e < 1$ and $0 \leq M \leq \pi$. A numerical study, for different values of M and e has been performed in [1]. We take values $M = 0.01$ and $e = 0.9995$. In this case the solution is $\alpha \approx 0.3899777749463621$.

Numerical computations have been carried out using variable precision arithmetic in MATLAB R2010b with 10000 significant digits.

Table 1 gives, for each test example an each method, the estimated error of the last iterate, $|x_{k+1} - x_k|$, the estimated order of convergence, ρ, (see [16])

$$q \approx \rho = \frac{\ln(|x_{k+1} - x_k| \,/\, |x_k - x_{k-1}|)}{\ln(|x_k - x_{k-1}| \,/\, |x_{k-1} - x_{k-2}|)}, \tag{14}$$

and the number of iterations, k, needed to satisfy the condition $|x_{k+1} - x_k| \leq 10^{-200}$. By means of (14), a vector is obtained by using the different iterations calculated in the process. The value of ρ that appears in Tables 1 to 3 is the last coordinate of this vector when the variation between its components is small. When these components are not stable, we will denote ρ by $'-'$.

		$\|x_{k+1} - x_k\|$	ρ	k		$\|x_{k+1} - x_k\|$	ρ	k
a) $x_0 = -1$	M2	1.18e−344	2	51	K2	1.18e−344	2	51
	M4	3.6e−395	4	7	K4	1.28e−320	4	7
	M8	9.57e−820	8.01	5	K8	4.36e−671	8	5
	M16	1.8e−944	14.84	4	K16	2.02e−872	14.08	4
b) $x_0 = 2$	M2	6.21e−296	2	16	K2	6.21e−296	2	16
	M4	2.67e−320	4	6	K4	1.81e−572	4	7
	M8	2.06e−211	8	4	K8	2.27e−739	7.99	5
	M16	1.67e−1853	16.27	4	K16	4.06e−826	17.10	4
c) $x_0 = 1$	M2	5.6e−250	2	10	K2	5.6e−250	2	10
	M4	1.06e−554	4	6	K4	2.37e−427	4	6
	M8	1.06e−295	8	4	K8	4.31e−204	8	4
	M16	7.79e−2367	15.76	4	K16	1.03e−1580	15.68	4
d) $x_0 = -1$	M2	1.93e−299	2	16	K2	1.93e−299	2	10
	M4	3.58e−260	4	6	K4	3.19e−250	4	6
	M8	8.38e−1016	8	5	K8	9.64e−279	8	5
	M16	1.23e−1074	16.03	4	K16	2.02e−285	15.99	4
e) $x_0 = 2$	M2	3.56e−291	2	19	K2	3.56e−291	2	19
	M4	4.06e−595	4	7	K4	7.44e−565	4	8
	M8	7.98e−816	7.99	5	K8	2.6e−1181	8	6
	M16	1.29e−918	16.50	4	K16	7.75e−2139	15.75	5
f) $x_0 = 1$	M2	2.04e−272	2	12	K2	2.04e−272	2	12
	M4	1.64e−671	4	7	K4	5.42e−483	4	7
	M8	1.72e−676	7.99	5	K8	1.65e−451	7.95	5
	M16	4.61e−667	14.16	4	K16	8.19e−434	12.64	4

Table 1: Numerical results for smooth functions

		$\lvert x_{k+1} - x_k \rvert$	ρ	k	α		$\lvert x_{k+1} - x_k \rvert$	ρ	k	α
$x_0 = 0.4$	M2	1.74e−254	2	11	1	K2	1.74e−254	2	11	1
	M4	7.23e−344	4	6	1	K4	5.04e−714	4	7	1
	M8	1.89e−1411	8	5	1	K8	1.27e−583	8.02	5	1
	M16	3.15e−1412	15.63	4	1	K16	1.94e−490	15.39	4	1
$x_0 = 0.2$	M2	1.62e−483	3	16	-1	K2	1.62e−483	3	16	-1
	M4	3.51e−247	2	10	0	K4	4.18e−224	2	11	0
	M8	2.99e−257	2	9	0	K8	2.85e−212	2	8	0
	M16	6.48e−223	2	8	0	K16	6.77e−234	2	8	0
$x_0 = -0.8$	M2	1.12e−481	3	7	-1	K2	1.12e−481	3	7	-1
	M4	4.63e−857	6	5	-1	K4	4.58e−766	6	5	-1
	M8	1.63e−1142	12	4	-1	K8	5.85e−963	12	4	-1
	M16	3.53e−381	24.06	3	-1	K16	5.07e−312	24	3	-1
$x_0 = 2$	M2	1.23e−288	3	8	-1	K2	1.23e−288	3	8	-1
	M4	2.44e−879	6	6	-1	K4	1.14e−791	6	6	-1
	M8	3.86e−1860	11.99	5	-1	K8	6.91e−1449	11.98	5	-1
	M16	5.16e−1239	18.82	4	-1	K16	1.28e−791	16.15	4	-1

Table 2: Numerical results for nonsmooth function (15)

The estimated convergence order of the proposed methods is always equal to or better than that of the classical family of Kung-Traub. The number of iterations is sometimes lower. When it is equal, the estimated error of the last iterate is also lower.

Now, we are going to test how the methods M_q and K_q behave on nonsmooth functions. The first test has been made on the function

$$f(x) = \begin{cases} x(x+1), & x < 0 \\ -2x(x-1), & x \geq 0. \end{cases} \tag{15}$$

In Table 2 we show the numerical results for different initial estimations. We observe that the behavior is similar in both families, but in some cases the best precision is obtained by K_q methods and in other cases the number of iterations needed by M_q schemes is lower.

Now, we consider, the nonsmooth function

$$f(x) = |x^2 - 9|. \tag{16}$$

		$\|x_{k+1}-x_k\|$	ρ	k	α		$\|x_{k+1}-x_k\|$	ρ	k	α
$x_0 = 2$	M2	-	-	$> 10^4$		K2	-	-	$> 10^4$	
	M4	-	-	$> 10^4$		K4	1.5e−348	4	7	3
	M8	2.44e−982	8	5	3	K8	-	-	$> 10^4$	
	M16	3.52e−389	18.96	4	3	K16	5.08e−454	-	16	3
$x_0 = 2.8$	M2	9.49e−294	2	30	3	K2	9.49e−294	2	30	3
	M4	-	-	$> 10^4$		K4	4.85e−760	4	7	3
	M8	6.10e−1270	8	5	3	K8	1.41e−343	-	11	3
	M16	2.12e−552	-	4	3	K16	3.18e−465	-	11	3
$x_0 = -2.8$	M2	-	-	$> 10^4$		K2	-	-	$> 10^4$	
	M4	4.21e−267	-	83	-3	K4	1.27e−314	4	11	3
	M8	1.30e−249	10.77	7	-3	K8	1.62e−1098	8	13	3
	M16	6.87e−296	19.6	5	-3	K16	9.22e−1587	15.3	8	3
$x_0 = -10$	M2	-	-	$> 10^4$		K2	-	-	$> 10^4$	
	M4	-	-	$> 10^4$		K4	1.03e−427	-	20	-3
	M8	5.65e−1318	-	10	-3	K8	8.35e−376	-	13	3
	M16	2.87e−1005	-	6	-3	K16	1.17e−251	-	7	3

Table 3: Numerical results for nonsmooth function (16)

It is of special interest as it has severe stability problems near the nonsmoothness.

The numerical experiments made on this function are summarized in Table 3. In this case, the advantages of the high order iterative schemes are evident when the initial estimation is far from the zero of the function. We can observe that in many cases the value ρ is not stable. Indeed, each family converges to a different solution in case of $x_0 = -2.8$, so the number of iterations and the difference $|x_{k+1} - x_k|$ gives no useful information. The methods analyzed have stability problems in function (16), but it can be observed that in many cases, the behavior of both families is complementary (see, for example, results from M8 and K8 in case $x_0 = 2$ and M4 vs K4 in case $x_0 = -10$), maybe because of they have been defined by means of direct and inverse interpolation techniques.

5. Conclusions

We have defined a family of optimal order derivative free iterative methods for the solution of nonlinear equations alternative to the family in [4], proving a convergence result that shows the optimality of the methods. We have also derived an explicit formula for the computation of the approximated derivative that avoids the solution of linear systems in each step of the iteration. The numerical results show that the new family has a slightly better performance than the classical one, so it can be competitive.

References

[1] J.M.A. Danby, T.M. Burkardt, *: The solution of Keplers equation, I*, Celestial Mechanics, 40 (1983) 95-107.

[2] Y.H.Geum, Y.I. Kim, *A biparametric family of optimally convergent sixteenth-order multipoint methods with their fourth-step weighting function as a sum of a rational and a generic two-variable function*, Journal of Computational and Applied Mathematics, 235 (2011) 3178-3188.

[3] A.M. Ostrowski, *Solutions of equations and systems of equations*, Academic Press, New York-London, 1966.

[4] H.T. Kung, J.F. Traub, *Optimal order of one-point and multi-point iteration*, Journal of the Association for Computing Machinery, 21 (1974) 643-651.

[5] J.M. Ortega, W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[6] P. Jarrat, *Some fourth order multipoint iterative methods for solving equations*, Math. Comp, 20 (1966) 434–437.

[7] W. Bi, H. Ren, Q. Wu, *Three-step iterative methods with eighth-order convergence for solving nonlinear equations*, Journal of Computational and Applied Mathematics 225 (2009) 105-112.

[8] J.R. Sharma, R. Sharma, *A new family of modified Ostrowski's methods with accelerated eighth order convergence*, Numerical Algoritms, 54 (2010) 445-458.

[9] W. Bi, Q. Wu, H. Ren, *A new family of eighth-order iterative methods for solving nonlinear equations*, Applied Mathematics and Computation, 214 (2009) 236-245.

[10] R. Thukral, M.S. Petkovic, *A family of three-point methods of optimal order for solving nonlinear equations*, Journal of Computational and Applied Mathematics, 233 (2010) 2278-2284

[11] L. Liu, X. Wang *Eighth-order methods with high efficiency index for solving nonlinear equations*, Applied Mathematics and Computation 215 (2010) 3449-3454.

[12] A. Cordero, J.L. Hueso, E. Martínez, J.R. Torregrosa, *New modifcations of Potra-Pták's method with optimal fourth and eighth order of convergence*, Journal of Computational and Applied Mathematics, 234 (2010) 2969–2976.

[13] B. Neta, *On a family of multipoint methods for non-linear equations*, International Journal of Computer Mathematics, 9 (1981) 353-361.

[14] L. Liu, X. Wang *New eighth-order methods for solving nonlinear equations*, Journal of Computational and Applied Mathematics, 234 (2010) 1611-1620.

[15] J.F. Traub, *Iterative methods for the solution of nonlinear equations*, Prentice Hall, New York, 1964.

[16] A. Cordero, J.R. Torregrosa, *Variants of Newton's method using fifth-order quadrature formulas*, Applied Mathematics and Computation, 190 (2007) 686- 698.