Applied Mathematics and Computation 216 (2010) 671–676

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

A class of three-point root-solvers of optimal order of convergence

Ljiljana D. Petković [a,*], Miodrag S. Petković [b], Jovana Džunić [b]

[a] Faculty of Mechanical Engineering, Department of Mathematics, University of Niš, 18000 Niš, Serbia
[b] Faculty of Electronic Engineering, Department of Mathematics, University of Niš, 18000 Niš, Serbia

ARTICLE INFO

Keywords:
Multipoint iterative methods
Nonlinear equations
Optimal order of convergence
Computational efficiency
Kung–Traub's conjecture

ABSTRACT

The construction of a class of three-point methods for solving nonlinear equations of the eighth order is presented. These methods are developed by combining fourth order methods from the class of optimal two-point methods and a modified Newton's method in the third step, obtained by a suitable approximation of the first derivative based on interpolation by a nonlinear fraction. It is proved that the new three-step methods reach the eighth order of convergence using only four function evaluations, which supports the Kung–Traub conjecture on the optimal order of convergence. Numerical examples for the selected special cases of two-step methods are given to demonstrate very fast convergence and a high computational efficiency of the proposed multipoint methods. Some computational aspects and the comparison with existing methods are also included.

1. Introduction

Let f be a real sufficiently smooth function, defined on an interval $I_f \subset \boldsymbol{R}$ which contains a simple root α of f. Although extensively investigated in Traub's book [14], multipoint iterative methods for solving a nonlinear equation $f(x) = 0$ have drawn a considerable attention in the first decade of the 21st century, which led to the construction of many methods of this type. These methods are primarily introduced with the aim to achieve as high as possible order of convergence using a fixed number of function evaluations. Namely, as Traub proved in [14], an one-point iterative method can reach the order of convergence at most p if it depends explicitly on the first $p-1$ derivatives of f. However, multipoint methods do not use derivatives of high order and overcome theoretical limits of one-point methods concerning the convergence order and computational efficiency. Studying the optimal convergence rate of multipoint methods, Kung and Traub [9] stated the hypothesis that *multipoint methods without memory for solving nonlinear equations, based on* $m+1$ *function evaluations per iteration, have the order of convergence at most* 2^m.

The main goal of this paper is to develop a general class of very efficient three-point methods for solving nonlinear equations. A new class of three-point methods has the order of convergence eight and uses only four function evaluations. In this way, the optimal order of convergence and optimal computational efficiency in the sense of the Kung–Traub conjecture are attained.

The paper is organized as follows. A new family of three-point methods of the optimal order eight is constructed in Section 2 by combining optimal two-point fourth order methods and a modified Newton's method in the third step. The modified Newton method is obtained using a suitable approximation to the first derivative of a function f in order to reduce the number of function evaluations. The total number of function evaluations of the proposed three-point family is four so that the optimal computational efficiency is $2^{3/4} \approx 1.682$. Numerical examples for the selected special cases of the proposed

* Corresponding author.
E-mail address: ljiljana@masfak.ni.ac.rs (L.D. Petković).

0096-3003/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2010.01.123

three-step methods are presented in Section 3. In this section, some computational aspects of the considered multipoint methods and the comparison with existing methods are also given.

2. A new family of optimal three-step methods

Let us consider an m-point iterative method (IM) without memory which requires $\theta(m) = m + 1$ function evaluations per iterative step (including derivatives of f, if they appear). According to the Kung–Traub conjecture, the order of convergence r of such a method cannot exceed 2^m. This bound is usually called *optimal order* of convergence and the corresponding multipoint method *optimal method*. Calculating the computational efficiency by the formula $E(\mathrm{IM}) = r^{1/\theta(m)}$ (see [11,14]), the *optimal* computational efficiency is $E_m^{(o)}(\mathrm{IM}) = 2^{m/(m+1)}$. For example, Newton's method

$$x_{k+1} = N(x_k) := x_k - u(x_k), \quad u(x) = \frac{f(x)}{f'(x)},$$

is the optimal second order method with two function evaluations and the efficiency $E_1^{(o)} = 2^{1/2} \approx 1.414$.

Let f be defined on an interval $I_f \subset \boldsymbol{R}$ and f' does not vanish on I_f, and assume that a simple root α of f is isolated in the interval I_f. Let $\Phi_{2^m}(m \geqslant 1)$ be a class of optimal m-point methods; it requires $\theta(m) = m + 1$ function evaluations and has the optimal order $r = 2^m$, giving the optimal computational efficiency $E_m^{(o)}(\Phi_{2^m}) = 2^{m/(m+1)}$. In this paper we construct a class Φ_8 of optimal three-point methods with the optimal order eight requiring four function evaluations. This class relies on optimal two-point methods belonging to the class Φ_4.

The first optimal two-point methods were developed by Ostrowski [11], Jarratt [6,7] and Kung and Traub [9]. Their good convergence properties have led to enormously rapid development of new methods of this type at the beginning of this century, see e.g., [1,3–5,8,12].

Assume that a real function p and its derivatives p' and p'' are continues in the neighborhood of 0. A rather wide class of optimal two-point methods can be obtained starting from the two-step iterative scheme

$$\begin{cases} y_k = N(x_k) = x_k - \frac{f(x_k)}{f'(x_k)}, \\ x_{k+1} = N(y_k) = y_k - \frac{f(y_k)}{f'(y_k)}, \end{cases} \qquad (k = 0, 1, \ldots),$$

and substituting the derivative $f'(y_k)$ by its approximation $f'(x_k)/p(t_k)$, where $t_k = f(y_k)/f(x_k)$ and p is chosen so that it satisfies the conditions $p(0) = 1,\ p'(0) = 2,\ |p''(0)| < \infty$. In this way we obtain a family of two-point methods

$$\begin{cases} y_k = N(x_k) = x_k - \frac{f(x_k)}{f'(x_k)}, \\ x_{k+1} = y_k - p(t_k)\frac{f(y_k)}{f'(x_k)}, \quad t_k = \frac{f(y_k)}{f(x_k)}, \end{cases} \qquad (k = 0, 1, \ldots). \tag{1}$$

Using the symbolic computation in the programming package *Mathematica* (*Maple* and *Matlab* are also convenient), it is easy to prove that the family (1) has the optimal order four. We note that Chun [4] approximated $f'(y_k)$ by $f'(x_k)h(t_k)$, but the approximation applied in (1) is slightly better since it directly (without any expansion of p) produces several existing optimal two-point methods as follows:

(**I**) For p given by

$$p(t) = \frac{1 + \beta t}{1 + (\beta - 2)t} \quad (\beta \in \boldsymbol{R}),$$

we obtain King's fourth order family of two-point methods [7] (omitting the iterative indices)

$$K(x;\beta) = x - u(x) - \frac{f(x - u(x))}{f'(x)} \cdot \frac{f(x) + \beta f(x - u(x))}{f(x) + (\beta - 2)f(x - u(x))}. \tag{2}$$

King's family produces the following special cases:

Ostrowski's method [11], $\beta = 0$:

$$K(x;0) = x - u(x) - \frac{u(x)f(x - u(x))}{f(x) - 2f(x - u(x))}; \tag{3}$$

Kou's method [8], $\beta = 1$:

$$K(x;1) = x - \frac{f(x)^2 + [f(x - u(x))]^2}{f'(x)[f(x) - f(x - u(x))]}; \tag{4}$$

Chun's method [4], $\beta = 2$:

$$K(x;2) = x - u(x)\left\{1 + \frac{f(x-u(x))}{f(x)} + \frac{2[f(x-u(x))]^2}{f(x)^2}\right\}. \tag{5}$$

(**II**) Choosing

$$p(t) = \frac{t^2 + (c-2)t - 1}{ct - 1} \quad (c \in \boldsymbol{R}),$$

we construct the iterative function $\omega_c(x, y)$,

$$\begin{cases} y = x - \frac{f(x)}{f'(x)}, \\ \omega_c(x,y) = y - \frac{f(y)}{f'(x)}\left\{1 + \frac{f(y)[f(y)-2f(x)]}{f(x)[cf(y)-f(x)]}\right\}, \end{cases} \tag{6}$$

which defines a one parameter family of two-point methods $x_{k+1} = \omega_c(x_k, y_k)$ of the order four. Taking $c = 1$ in (6), we obtain Maheshvari's method [10] as a special case,

$$M(x) = x - u(x)\left\{\frac{[f(x-u(x))]^2}{f(x)^2} - \frac{f(x)}{f(x-u(x)) - f(x)}\right\}. \tag{7}$$

(**III**) In a limit case when $t \to 0$ the function

$$p(t) = \frac{1}{t}\left(\frac{2}{1+\sqrt{1-4t}} - 1\right),$$

provides the fourth order method proposed in [12]

$$E(x) = x - \frac{2u(x)}{1 + \sqrt{1 - \frac{4f(x-u(x))}{f(x)}}}. \tag{8}$$

Remark 1. The computational efficiency of optimal two-point methods is $E_2^{(o)}(\Phi_4) = 2^{2/3} \approx 1.587$.

Now we will construct a new family of three-step iterative methods having optimal order of convergence equal to eight. For simplicity, we will sometimes omit iterative indices and denote a new approximation x_{k+1} with $\hat{x}$. Let $\varphi_f \in \Phi_4$ denote an iterative function from the class of optimal two-point iterative methods. Then the improved approximation $\hat{x}$ to the root α can be found by the following three-point iterative scheme:

$$\begin{cases} (1) \quad y = x - \frac{f(x)}{f'(x)}, \\ (2) \quad z = \varphi_f(x,y), \quad \varphi_f \in \Phi_4, \\ (3) \quad \hat{x} = z - \frac{f(z)}{f'(z)}. \end{cases} \tag{9}$$

We note that the first two steps define an optimal two-point method from the class Φ_4 with the order $r_1 = 4$ using the Newton method in the first step, while the third step is Newton's method of the order $r_2 = 2$. The presented scheme is simple and its convergence rate is equal to eight, which is a consequence of the following Traub's theorem [14, Theorem 2.4]:

Theorem 1. *Let g_1 and g_2 be iterative functions with the order of convergence r_1 and r_2, respectively. Then iterative function $G(x) = g_2(g_1(x))$ defines a composite iterative method of the order $r_1 \cdot r_2$.*

However, the three-point method (9) requires five function evaluations per iterative step so that it is *not* optimal in the sense of Kung–Traub's conjecture. To reduce the number of function evaluations and thus increase the computational efficiency, we will approximate $f'(z)$ using available data. Since we have four values $f(x),\ f'(x),\ f(y)$ and $f(z)$, one of the ways is to approximate f by the Hermite interpolation polynomial h of degree 3 in the nodes $x,\ y,\ z$ and use the approximation $f'(z) \approx h'(z)$ in the third step of the iterative scheme (9). The described approach was applied in [13] where a general class of optimal n-point methods is constructed for arbitrary $n \geqslant 3$.

In this paper we will approximate $f'(z)$ by applying the interpolation of f by a nonlinear fraction

$$w(t) = \frac{a_1 + a_2(t-x) + a_3(t-x)^2}{1 + a_4(t-x)} \quad (a_2 - a_1a_4 \neq 0). \tag{10}$$

From (10) we have

$$w'(t) = \frac{a_2 - a_1a_4 + a_3(t-x)(2 + a_4(t-x))}{(1 + a_4(t-x))^2}. \tag{11}$$

The unknown coefficients $a_1, \ldots, a_4$ will be determined from the conditions:

$$\text{(i)}\ w(x) = f(x), \quad \text{(ii)}\ w(y) = f(y), \quad \text{(iii)}\ w(z) = f(z), \quad \text{(iv)}\ w'(x) = f'(x). \tag{12}$$

Putting $t = x$ into (10) and (11) and using (12-i) and (12-iv), we get $a_1 = f(x)$ and $a_2 = f'(x) + a_4 a_1$ $(= w'(x))$. The coefficients a_3 and a_4 are obtained from the system of two linear equations formed by using the remaining two conditions (12-ii) and (12-iii), and putting y and z into (10). We get

$$a_3 = \frac{f'(x)f[y,z] - f[x,y]f[x,z]}{xf[y,z] + \frac{yf(z)-zf(y)}{y-z} - f(x)}, \quad a_4 = \frac{a_3}{f[x,y]} + \frac{f'(x) - f[x,y]}{(y-x)f[x,y]}, \tag{13}$$

where $f[x,y] = (f(y) - f(x))/(y-x)$ denotes a divided difference. Finally, we find

$$a_2 = f'(x) + a_4 a_1 = f'(x) + a_4 f(x), \quad \text{recalling that} \quad a_1 = f(x). \tag{14}$$

Replacing the obtained coefficients into (11) and putting $t = z$, we get the explicit formula for $w'(z)$ which uses only already calculated quantities $f(x),\ f'(x),\ f(y)$, and $f(z)$. In this way, the nonlinear fraction w and its derivative w' are completely determined by (10)–(14).

Setting $w'(z)$ (calculated by (11) putting $t = z$) into (9) instead of $f'(z)$, we state a new family of three-point methods: Given an initial approximation x_0, the improved approximations x_k $(k = 1, 2, \ldots)$ are calculated by the three-step procedure

$$\begin{cases} y_k = x_k - \frac{f(x_k)}{f'(x_k)}, \\ z_k = \varphi_f(x_k, y_k), \quad (k = 0, 1, \ldots). \\ x_{k+1} = z_k - \frac{f(z_k)}{w'(z_k)}, \end{cases} \tag{15}$$

The proposed class of root-solvers requires only four function evaluations. We will see later that its order of convergence is eight, that is, the family of three-point methods (15) is optimal.

Remark 2. There are other ways to approximate $f'(z)$ using available data. For example, Bi et al. [2] used an approximation of the second derivative f'' and divided differences. However, in order to provide the optimal order equal to eight, this approach must use only King's method (2) in the first two step, that is, $\varphi_f = K(x;\beta)$. On the other hand, any optimal two-point method $\varphi_f \in \Phi_4$ can be applied in our algorithm (15), which is more general.

Many authors determine the order of convergence using a standard technique which mainly relies on the Taylor series. In the case of multipoint methods, such an approach deals with rather cumbersome expressions so that their development and manipulations with them most often require the use of computer. If a computer is employed anyway, it is reasonable to determine the convergence rate using a symbolic computation, as done in this paper. If necessary, intermediate expressions can be always printed during this computation, although their presentation is most frequently of academic interest.

Using the Taylor series and symbolic computation in the programming package *Mathematica* (or *Maple*, *Matlab*), we can find the order of convergence and the asymptotic error constant of the three-point methods (15). The following abbreviations are used in the program given below.

ck $= f^{(k)}(\alpha)/(k!f'(\alpha))$, e $= x - \alpha$, e1 $= \hat{x} - \alpha$,
fx $= f(x)$, fy $= f(y)$, fz $= f(z)$, f1x $= f'(x)$, f1a $= f'(\alpha)$,
w1z $= w'(z)$ (calculated by (11)–(14)).

Program (written in *Mathematica*):

```
fx = f1a * (e + c2 * e^2 + c3 * e^3 + c4 * e^4); f1x = D[fx, e];
u = e - Series[fx/f1x, e, 0, 7 ]; fy = f1a * (u + c2 * u^2 + c3 * u^3 + c4 * u^4);
v = q * e^4; fz = f1a * (v + c2 * v^2 + c3 * v^3 + c4 * v^4); fxy = (fx - fy)/(e - u);
fxz = (fx - fz)/(e - v); fyz = (fy - fz)/(u - v); a1 = fx;
a3 = ((f1x * fyz - fxy * fxz))/((e * (fy - fz) + u * fz - v * fy)/(u - v) - fx);
a4 = a3/fxy + (f1x - fxy)/((u - e) * fxy); a2 = f1x + fx * a4 ;
w1z = (a2 - a1 * a4 + a3(v - e) * (2 + a4(v - e)))/(1 + a4(v - e))^2; e1 = v - fz/w1z//Simplify
```

$$\boxed{\texttt{Out}[e1] = q(c_2(c_4 + q) - c_3^2)e^8 + O[e^9]} \tag{16}$$

The output (16) of the above program points to the eighth order of convergence of the family of three-point methods (15). Taylor's expansions used in the program assume sufficiently small $e = x - \alpha$, which means that the initial approximation should be reasonably close to the root α. Altogether, we can state the following theorem.

Theorem 2. *If an initial approximation x_0 is sufficiently close to the root α of a function f, then the convergence order of the family of three-point methods (15) is equal to eight.*

Remark 3. Since the number of function evaluations is $\theta(3) = 4$ and the convergence order is $2^3 = 8$ for the considered family of three-point methods (15), we conclude that the Kung–Traub conjecture is supported for $m = 3$.

From (16) we observe that the asymptotic error constant (AEC) of the family of methods (15) is given by

$$\text{AEC}(15) = \lim_{k\to\infty} \frac{e_{k+1}}{e_k^8} = q\left[c_2(c_4 + q) - c_3^2\right].$$

The AEC q should be determined for each particular two-point method φ_f applied in the iterative scheme (15). For example, $q = \text{AEC}(2) = c_2^3(1 + 2\beta) - c_2c_3$ for King's two-point method (2) so that the AEC of the three-point method (15)–(2) in this particular case is

$$\text{AEC}((15) - (2)) = \left[c_2^3(1 + 2\beta) - c_2c_3\right]\left[c_2\left(c_4 + c_2^3(1 + 2\beta) - c_2c_3\right) - c_3^2\right].$$

3. Computational aspects

In this section we demonstrate the convergence behavior of the proposed class of three-point methods (15) by presenting numerical results for different two-point methods from the class Φ_4. We have chosen the Ostrowski method (3), the King family for particular values of parameter $\beta = 1$ (Kou's method (4)) and $\beta = 2$ (Chun's method (5)), the Maheshwari method (7), and the method (8) developed in [12].

For comparison, we also tested two methods from the family of the eighth order methods recently proposed by Bi et al. in [2], referred to as Method 1 and Method 2 in [2], a general class of n-point methods (for $n = 3$) proposed by Petković [13], taking King's method for $\beta = 0$ and $\beta = 1$ (referred to as P-1 and P-2), and two almost forgotten families of Kung and Traub stated in 1974, see [9]. These latter families are denoted in Tables 1 and 2 with K-T-version 1 (family without derivatives), and K-T-version 2 (family with the first derivative).

For demonstration, among many numerical examples we selected two examples implemented in the programming package *Mathematica* by the use of multi-precision arithmetic. The tables of results also contain the computational order of convergence, evaluated by the following formula (see [15])

$$\tilde{r} \approx \frac{\log|(x_{k+1} - \alpha)/(x_k - \alpha)|}{\log|(x_k - \alpha)/(x_{k-1} - \alpha)|}. \tag{17}$$

Table 1
$f(x) = e^{-x^2+x+2} - \cos(x+1) + x^3 + 1,\ \alpha = -1,\ x_0 = -0.7.$

Three-point methods	$\lvert x_1 - \alpha\rvert$	$\lvert x_2 - \alpha\rvert$	$\lvert x_3 - \alpha\rvert$	$\tilde{r}(17)$
(15)–(3)	5.64(−7)	1.35(−52)	1.42(−417)	8.00023
(15)–(4)	2.85(−9)	8.32(−72)	2.57(−571)	7.98773
(15)–(5)	3.96(−7)	5.57(−54)	8.54(−429)	7.99999
(15)–(7)	2.04(−7)	3.11(−56)	8.92(−447)	8.00015
(15)–(8)	9.97(−7)	1.38(−50)	1.86(−401)	8.00000
K-T-version 1 [9]	2.82(−7)	2.18(−55)	2.81(−440)	7.99990
K-T-version 2 [9]	2.45(−7)	5.73(−56)	5.07(−445)	8.00010
Bi-Method 1 [2]	7.87(−7)	4.47(−52)	4.86(−414)	7.99996
Bi-Method 2 [2]	1.19(−6)	1.69(−50)	2.92(−401)	7.99957
P-(1) [13]	2.92(−7)	1.02(−55)	2.16(−443)	8.00041
P-(2) [13]	1.11(−9)	3.67(−75)	5.05(−599)	8.00025

Table 2
$f(x) = \ln(x^2 + x + 2) - x + 1,\ \alpha = 4.1525907367\ldots,\ x_0 = 3.$

Three-point methods	$\lvert x_1 - \alpha\rvert$	$\lvert x_2 - \alpha\rvert$	$\lvert x_3 - \alpha\rvert$	$\tilde{r}(17)$
(15)–(3)	3.29(−7)	2.92(−60)	1.12(−484)	8.00003
(15)–(4)	1.80(−6)	7.10(−54)	4.16(−433)	8.00000
(15)–(5)	5.24(−6)	7.52(−50)	1.35(−400)	8.00002
(15)–(7)	3.30(−6)	1.33(−51)	9.29(−415)	7.99997
(15)–(8)	8.20(−8)	1.74(−65)	7.06(−527)	8.00001
K-T-version 1 [9]	4.27(−6)	2.04(−50)	5.63(−405)	7.99984
K-T-version 2 [9]	4.39(−6)	2.62(−50)	4.29(−404)	7.99983
Bi-Method 1 [2]	9.18(−8)	9.85(−65)	1.73(−520)	8.00000
Bi-Method 2 [2]	4.72(−6)	2.51(−50)	1.63(−404)	7.99985
P-(1) [13]	6.85(−7)	2.34(−57)	4.39(−461)	7.99999
P-(2) [13]	2.58(−6)	2.18(−52)	5.66(−421)	7.99999

Example 1. We applied the aforementioned methods to the test function

$$f(x) = e^{-x^2+x+2} - \cos(x+1) + x^3 + 1,$$

for finding its root which is near to $x_0 = -0.7$. The exact root is $\alpha = -1$. The absolute values of the errors of the approximations x_k in the first three iterations are displayed in Table 1, where $A(-h)$ means $A \times 10^{-h}$.

Example 2. The three-point methods employed in Example 1 were also applied to the function

$$f(x) = \ln(x^2 + x + 2) - x + 1,$$

to approximate its root $\alpha = 4.15259073675715827499\ldots$. In this test we used the common initial approximation $x_0 = 3$. The obtained results are presented in Table 2.

Regarding the results given in Tables 1 and 2 and a number of numerical examples, we have concluded that the proposed optimal three-point methods (15) are extremely fast and very efficient. Their computational efficiency

$$E_3^{(o)}(\Phi_8) = 2^{3/4} \approx 1.682 > E_2^{(o)}(\Phi_4) = 2^{2/3} \approx 1.587,$$

is higher than the efficiency of optimal two-point methods (see Remark 1). Two iterative steps are usually sufficient in solving most practical problems at present. The third iteration is given only to demonstrate remarkably fast convergence of the considered root-solvers. From a comparison study, we have also concluded that the proposed family (15) and the eight-order methods given in the papers [2,9,13] produce results of the approximately same quality.

Solving a number of nonlinear equations, we did not find a specific iterative function $\varphi_f \in \Phi_4$ which would be approximately best for all tested nonlinear equations. All methods from the family (15) show good convergence behavior if initial approximations are close to the roots, and the tested functions are not "pathological." In such situations, the computational order of convergence $\tilde{r}$, given by (17), perfectly matches the theoretical result given in Theorem 2.

Contrary, bad initial approximations can cause slower convergence of multipoint methods at the beginning of iterative process. For this reason, the problem of finding good initial approximations is as equally important as the convergence rate of root-finding methods. The determination of a good initial approximation is often more profitable than the implementation of a very fast multipoint method with a bad starting approximation. Indeed, since multipoint methods have the form of a predictor-corrector method, the predictor (usually Newton's method) will not deliver a good approximation to the corrector if the chosen initial approximation is not good enough. We draw a reader's attention to an efficient method for determining initial approximations of great accuracy, given by Yun [16] and used in our numerical experimentations. For illustration, let us consider the function $f(x) = \ln(x^2 + x + 2) - x + 1$ (tested in Example 2) and a rather wide interval [0,6] containing a simple root of f. Using Yun's approach [16] based on numerical integration and simple statements in the programming package *Mathematica*

```
f[x] = Log[x^2 + x + 2] - x + 1; a = 0; b = 6; n = 20 ;
x0 = 0.5 * (a + b + Sign[f[a]] * NIntegrate[Tanh[n * f[x]], {x, a, b}])
```

we find considerably good initial approximation $x_0 = 4.15225$ with the error $|x_0 - \alpha| \approx 3.4 \times 10^{-4}$.

Acknowledgement

This work was supported by the Serbian Ministry of Science under grant 144024.

The authors wish to thank Dr. Melvin Scott and an anonymous referee for stimulating discussions on the topic and for contributing to the improvement of the presentation of this paper.

References

[1] S. Amat, S. Busquier, S. Plaza, Review of some iterative root-finding methods from a dynamical point of view, Sci. Ser. A Math. 10 (2004) 3–35.
[2] W. Bi, H. Ren, Q. Wu, A new family of eight-order iterative methods for solving nonlinear equations, Appl. Math. Comput. 214 (2009) 236–245.
[3] C. Chun, A family of composite fourth-order iterative methods for solving nonlinear equations, Appl. Math. Comput. 187 (2007) 951–956.
[4] C. Chun, Some fourth-order iterative methods for solving nonlinear equations, Appl. Math. Comput. 195 (2008) 454–459.
[5] C. Chun, Y. Ham, A one-parameter fourth-order family of iterative methods for nonlinear equations, Appl. Math. Comput. 189 (2007) 610–614.
[6] P. Jarratt, Some fourth order multipoint methods for solving equations, Math. Comput. 20 (1966) 434–437.
[7] R. King, A family of fourth order methods for nonlinear equations, SIAM J. Numer. Anal. 10 (1973) 876–879.
[8] J. Kou, Y. Li, X. Wang, A composite fourth-order iterative method, Appl. Math. Comput. 184 (2007) 471–475.
[9] H.T. Kung, J.F. Traub, Optimal order of one-point and multipoint iteration, J. ACM 21 (1974) 643–651.
[10] A.K. Maheshwari, A fourth-order iterative method for solving nonlinear equations, Appl. Math. Comput. 211 (2009) 383–391.
[11] A.M. Ostrowski, Solution of Equations and Systems of Equations, Academic Press, New York, 1960.
[12] M.S. Petković, L.D. Petković, A one parameter square root family of two-step root-finders, Appl. Math. Comput. 188 (2007) 339–344.
[13] M.S. Petković, On a general class of multipoint root-finding methods of high computational efficiency, SIAM J. Numer. Anal. 47 (2010) 4402–4414.
[14] J.F. Traub, Iterative Methods for the Solution of Equations, Prentice-Hall, Englewood Cliffs, New Jersey, 1964.
[15] S. Weerakoon, T.G.I. Fernando, A variant of Newton's method with accelerated third-order convergence, Appl. Math. Lett. 13 (2000) 87–93.
[16] B.I. Yun, A non-iterative method for solving non-linear equations, Appl. Math. Comput. 198 (2008) 691–699.