Journal of Computational and Applied Mathematics 235 (2011) 4199–4206

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Constructing higher-order methods for obtaining the multiple roots of nonlinear equations[☆]

Xiaojian Zhou[a,b], Xin Chen[a], Yongzhong Song[a,*]

[a] Jiangsu Key Laboratory for NSLSCS, School of Mathematical Sciences, Nanjing Normal University, Nanjing 210046, PR China

[b] School of Science, Nantong University, Nantong 226007, PR China

ARTICLE INFO

Article history:
Received 18 November 2010
Received in revised form 10 March 2011

MSC:
41A25
65D99

Keywords:
Nonlinear equations
Iterative method
Multiple roots
Convergence order
Efficiency index

ABSTRACT

This paper concentrates on iterative methods for obtaining the multiple roots of nonlinear equations. Using the computer algebra system Mathematica, we construct an iterative scheme and discuss the conditions to obtain fourth-order methods from it. All the presented fourth-order methods require one-function and two-derivative evaluation per iteration, and are optimal higher-order iterative methods for obtaining multiple roots. We present some special methods from the iterative scheme, including some known already. Numerical examples are also given to show their performance.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

One of the most important and challenging problems in scientific and engineering computations is to find the solutions to a nonlinear equation $f(x) = 0$. We concern ourselves with iterative methods to find the multiple roots $x^\star$ with multiplicity m of a nonlinear equation $f(x) = 0$, i.e., $f^{(i)}(x^\star) = 0$, $i = 0, 1, \ldots, m-1$, and $f^{(m)}(x^\star) \neq 0$.

A variant of Newton's method for obtaining multiple roots, given in [1], is quadratically convergent, and is given by

$$x_{n+1} = x_n - m\frac{f(x_n)}{f'(x_n)}. \tag{1}$$

In order to improve the convergence of iterative methods for multiple roots, some researchers, such as Dong [2,3], Neta et al. [4–7], and Li et al. [8,9], have developed some iterative methods with higher order of convergence. Some of these methods are of order three [2–4,6], while others are of order four [5,7–9]. All these methods require the knowledge of the multiplicity m. In this paper, we only concern ourselves with iterative methods of order four.

Based on the work of Jarratt [10], Neta et al. [5] have presented a fourth-order method requiring one-function and three-derivative evaluation per iteration, given by the iteration function

$$x_{n+1} = x_n - \frac{f(x_n)}{a_1 f'(x_n) + a_2 f'(y_n) + a_3 f'(\eta_n)}, \tag{2}$$

☆ This work is supported by the National Natural Science Foundation of China (10971252), and the Foundation for the Authors of the National Excellent Doctoral Thesis Award of China (200720).

* Corresponding author.
E-mail addresses: zxjntu@gmail.com (X. Zhou), xchen@njnu.edu.cn (X. Chen), yzsong@njnu.edu.cn (Y. Song).

0377-0427/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2011.03.014

where

$$y_n = x_n - a\frac{f(x_n)}{f'(x_n)}, \qquad \eta_n = x_n - b\frac{f(x_n)}{f'(x_n)} - c\frac{f(x_n)}{f'(y_n)}. \tag{3}$$

The values for the parameters a, b, c, a_1, a_2, and a_3 for several values of m are discussed by the authors.

Neta [7] has also developed another fourth-order method, requiring one-function and three-derivative evaluation per iteration:

$$x_{n+1} = x_n - a_1\frac{f(x_n)}{f'(x_n)} - a_2\frac{f(x_n)}{f'(y_n)} - a_3\frac{f(x_n)}{f'(\eta_n)} - \frac{f(x_n)}{b_1 f'(x_n) + b_2 f'(y_n)}, \tag{4}$$

where y_n and η_n are given by (3). A table of values for the parameters $a, b, c, a_1, a_2, a_3, b_1$, and b_2 for several values of m is given in [7].

Inspired by another work of Jarratt [11], Sharma and Sharma [12] present a variant of the Jarratt method for obtaining multiple roots, which has fourth order of convergence and requires one-function and two-derivative evaluation per iteration:

$$\begin{cases} y_n = x_n - \dfrac{2m}{2+m}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{m}{8}\left[(m^3 - 4m + 8) - (m+2)^2\left(\dfrac{m}{m+2}\right)^m \dfrac{f'(x_n)}{f'(y_n)}\right. \\ \quad \left. \times\left(2(m-1) - (m+2)\left(\dfrac{m}{m+2}\right)^m \dfrac{f'(x_n)}{f'(y_n)}\right)\right]\dfrac{f(x_n)}{f'(x_n)}. \end{cases} \tag{5}$$

In [9], Li et al. present six fourth-order methods with closed formulae for obtaining the multiple roots of nonlinear equations. Among them, the following two methods are more efficient, since they also only require one-function and two-derivative evaluation per iteration.

$$\begin{cases} y_n = x_n - \dfrac{2m}{m+2}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - a_3\dfrac{f(x_n)}{f'(y_n)} - \dfrac{f(x_n)}{b_1 f'(x_n) + b_2 f'(y_n)}, \end{cases} \tag{6}$$

where

$$a_3 = -\frac{1}{2}\frac{m(m-2)(m+2)^3\left(\frac{m}{m+2}\right)^m}{(m^3 - 4m + 8)},$$

$$b_1 = -\frac{(m^3 - 4m + 8)^2}{m(m^2 + 2m - 4)^3},$$

$$b_2 = \frac{m^2(m^3 - 4m + 8)\left(\frac{m+2}{m}\right)^m}{(m^2 + 2m - 4)^3},$$

and

$$\begin{cases} y_n = x_n - \dfrac{2m}{m+2}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - a_3\dfrac{f(x_n)}{f'(x_n)} - \dfrac{f(x_n)}{b_1 f'(x_n) + b_2 f'(y_n)}, \end{cases} \tag{7}$$

with

$$a_3 = -\frac{1}{2}m(m-2), \qquad b_1 = -\frac{1}{m}, \qquad b_2 = \frac{1}{m}\left(\frac{2+m}{m}\right)^m.$$

2. Development of a high-order method

Considering the following iterative method:

$$\begin{cases} y_n = x_n - t\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - Q\left(\dfrac{f'(y_n)}{f'(x_n)}\right)\dfrac{f(x_n)}{f'(x_n)}, \end{cases} \tag{8}$$

where t is a parameter and the function $Q(\cdot) \in C^2(\mathbb{R})$. Note that (5)–(7) are members of the family (8).

Let $e_n = x_n - x^\star$ and let $f(x)$ be a sufficiently differentiable function. Expanding $f(x_n)$ and $f'(x_n)$ at $x = x^\star$ with Taylor series, we then have

$$f(x_n) = \frac{f^{(m)}(x^\star)}{m!} e_n^m \left(1 + c_1 e_n + c_2 e_n^2 + c_3 e_n^3 + O(e_n^4)\right),$$

and

$$f'(x_n) = \frac{f^{(m)}(\alpha)}{(m-1)!} e_n^{m-1} \left(1 + \frac{m+1}{m} c_1 e_n + \frac{m+2}{m} c_2 e_n^2 + \frac{m+3}{m} c_3 e_n^3 + \cdots\right),$$

where $c_i = \frac{m!}{i!} \frac{f^{(i)}(x^\star)}{f^{(m)}(x^\star)}$ and $i \geq 1$.

Using a computer algebra system such as Mathematica, we can get

$$\frac{f'(y_n)}{f'(x_n)} = \mu^{m-1} + \frac{c_1 t\,(m(-2+t)+t)\,\mu^m}{m\,(m-t)^2} e_n + \frac{t\mu^m}{2m^2(m-t)^3}(h_1 c_1^2 + 2h_2 c_2) e_n^2 + \frac{t\mu^m}{6m^3(m-t)^4}(6h_3 c_3 + h_4 c_1^3 + 12 h_5 c_1 c_2) e_n^3 + O(e_n^4), \tag{9}$$

where

$$\mu = 1 - \frac{t}{m},$$
$$h_1 = 2(m+1)(2m+1)t^2 - 3m(m^2+5m+2)t + 6m^2(m+1),$$
$$h_2 = (2+m)t^3 - 4m(2+m)t^2 + 3m^2(4+m)t - 6m^3,$$
$$h_3 = (m-t)^2((m+3)t^3 - 4m(m+3)t^2 + 6m^2(m+3)t - 12m^3),$$
$$h_4 = 3(m+1)(7m^2+7m+2)t^3 - 4m(8m^3+33m^2+25m+6)t^2 + 12m^2(m+1)(m^2+8m+3)t - 24m^3(m+1)^2,$$
$$h_5 = (m+1)(m+2)t^4 - 2m(m+1)(3m+5)t^3 + 4m^2(2m^2+8m+5)t^2 - m^3(3m^2+25m+20)t + 2m^4(3m+4).$$

Let $\frac{f'(y_n)}{f'(x_n)} = u + v$, where $u = \mu^{m-1}$. Then, from (9), the remainder $v = \frac{f'(y_n)}{f'(x_n)} - u$ is infinitesimal with the same order of e_n. Thus, we can Taylor expand $Q(\frac{f'(y_n)}{f'(x_n)}) = Q(u+v)$ about u and then obtain

$$Q\left(\frac{f'(y_n)}{f'(x_n)}\right) = Q(u) + Q'(u)v + \frac{Q''(u)v^2}{2} + \frac{Q'''(u)v^3}{3!} + O(e_n^4).$$

Again by the help of Mathematica, we can obtain the error equation

$$\begin{aligned} e_{n+1} &= e_n - Q\left(\frac{f'(y_n)}{f'(x_n)}\right)\frac{f(x_n)}{f'(x_n)} \\ &= \left(1 - \frac{Q(u)}{m}\right)e_n + \left(\frac{1}{m^2}Q(u) - \frac{(mt+t-2m)t\mu^m}{m^2(m-t)^2}Q'(u)\right)c_1 e_n^2 + (p_1 c_1 + p_2 c_2)e_n^3 + O(e_n^4), \end{aligned} \tag{10}$$

where

$$p_1 = \frac{2}{m^2}Q(u) - \frac{(m+2)t^3 - 4m(m+2)t^2 + 3m^2(m+4)t - 6m^3}{m^3(m-t)^3} t\mu^m Q'(u),$$
$$p_2 = -\frac{t^2(m(-2+t)+t)^2\mu^{2m}}{2m^3(m-t)^4}Q''(u) - \frac{m+1}{m^3}Q(u) + \left(\frac{mt+t-2m}{m^3(m-t)^2} + \frac{(m+1)(2m+1)}{m^3(m-t)^4}t^3 + \frac{7m^2+21m+8}{2m^2(m-t)^4}t^2 + \frac{3m^2+21m+12}{2m(m-t)^4}t + \frac{3(m+1)}{(m-t)^4}\right)t\mu^m Q'(u).$$

Thus, to obtain an iterative method of order four, the coefficients of e_n, e_n^2, and e_n^3 in error equation (10) should all be zeros. Furthermore, to get a fourth-order method independent of the information of $f(x)$, we also should ensure that $p_1 = p_2 = 0$. So we have the following equations involving $Q(u)$, $Q'(u)$, $Q''(u)$, and t.

$$\begin{cases} \dfrac{Q(u)}{m} = 1, \\ Q(u) = \dfrac{Q'(u)t(mt+t-2m)\left(1-\frac{t}{m}\right)^m}{(m-t)^2}, \\ p_1 = 0, \\ p_2 = 0. \end{cases}$$

Solving these equations, we get

$$\begin{cases} t = \dfrac{2m}{2+m}, \\ Q(u) = m, \\ Q'(u) = -\dfrac{1}{4}m^{3-m}(2+m)^m, \\ Q''(u) = \dfrac{1}{4}m^4\left(\dfrac{m}{2+m}\right)^{-2m}, \end{cases}$$

where $u = (\frac{m}{2+m})^{m-1}$.

From the discussion above we can deduce the following conclusion.

Theorem 2.1. *Let $x^\star \in \mathbb{R}$ be a multiple root of multiplicity m of a sufficiently differentiable function $f : I \to \mathbb{R}$ for an open interval I. If the initial point x_0 is sufficiently close to $x^\star$, then the convergence order of the method defined by (8) is at least four, when the following equations hold:*

$$t = \frac{2m}{2+m},$$

and

$$\begin{cases} Q(u) = m, \\ Q'(u) = -\dfrac{1}{4}m^{3-m}(2+m)^m, \\ Q''(u) = \dfrac{1}{4}m^4\left(\dfrac{m}{2+m}\right)^{-2m}, \end{cases}$$

where $u = (\frac{m}{2+m})^{m-1}$.

Remark 1. From the error equation (10), we can find that the iterative method (8) contains a variant of Newton method (1) as a second-order method. It is also easy to find the conditions to obtain third-order methods.

Remark 2. One should note that, in (8), three new function evaluations for $f(x_n)$, $f'(x_n)$, and $f'(y_n)$ are required per iteration. So Theorem 2.1 shows that the method (8) is optimal with convergence order of four, as expected by the conjecture in [13].

Remark 3. Consider the definition of an efficiency index as $p^{1/q}$, where p is the order of the method and q is the number of function evaluations per iteration required by the method. The fourth-order methods (8) have the efficiency index $4^{1/3} \approx 1.587$, which is better than $2^{1/2} \approx 1.414$ of Newton method (1), and $4^{1/4} \approx 1.414$ of the fourth-order methods (2) and (4).

3. Some special cases of order four

In this section, we will give some special cases of order four of the presented method (8). According to Theorem 2.1, $t = \frac{2m}{2+m}$; then $u = (\frac{m}{2+m})^{m-1}$.

Case 1. First, we consider the simplest case. Suppose that

$$Q(x) = Ax^2 + Bx + C.$$

Then

$$Q'(x) = 2Ax + B, \qquad Q''(x) = 2A.$$

According to Theorem 2.1, we should solve the following equations:

$$\begin{cases} Au^2 + Bu + C = m, \\ 2Au + B = -\dfrac{1}{4}m^{3-m}(2+m)^m, \\ 2A = \dfrac{1}{4}m^4\left(\dfrac{m}{2+m}\right)^{-2m}. \end{cases}$$

The solution to the equations above is

$$\begin{cases} A = \dfrac{1}{8}m^4\left(\dfrac{m+2}{m}\right)^{2m}, \\ B = -\dfrac{1}{4}m^3(m+3)\left(\dfrac{2+m}{m}\right)^{m}, \\ C = \dfrac{1}{8}m(m^3+6m^2+8m+8), \end{cases}$$

and thus we obtain the following iterative method of order four:

$$\begin{cases} y_n = x_n - \dfrac{2m}{2+m}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{m}{8}\left[m^3\left(\dfrac{m+2}{m}\right)^{2m}\left(\dfrac{f'(y_n)}{f'(x_n)}\right)^2 - 2m^2(m+3)\left(\dfrac{2+m}{m}\right)^{m}\dfrac{f'(y_n)}{f'(x_n)}\right. \\ \quad \left. + (m^3+6m^2+8m+8)\right]\dfrac{f(x_n)}{f'(x_n)}. \end{cases} \tag{11}$$

Case 2. Let

$$Q(x) = Ax + \frac{B}{x} + C.$$

Then

$$Q'(x) = A - \frac{B}{x^2} \quad \text{and} \quad Q''(x) = \frac{2B}{x^3}.$$

So we have the following equations:

$$\begin{cases} Au + \dfrac{B}{u} + C = m, \\ A - \dfrac{B}{u^2} = -\dfrac{1}{4}m^{3-m}(2+m)^m, \\ \dfrac{2B}{u^3} = \dfrac{1}{4}m^4\left(\dfrac{m}{2+m}\right)^{-2m}. \end{cases}$$

Solving them, we get

$$A = \frac{1}{8}m^4\left(\frac{m+2}{m}\right)^{m}, \qquad B = \frac{1}{8}m(m+2)^3\left(\frac{m}{m+2}\right)^{m}, \qquad C = -\frac{1}{4}m(m^3+3m^2+2m-4),$$

and hence another fourth-order convergent iterative scheme:

$$\begin{cases} y_n = x_n - \dfrac{2m}{2+m}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \dfrac{m^4}{8}\left(\dfrac{m+2}{m}\right)^{m}\dfrac{f'(y_n)}{f'(x_n)}\dfrac{f(x_n)}{f'(x_n)} - \dfrac{m(m+2)^3}{8}\left(\dfrac{m}{m+2}\right)^{m}\dfrac{f(x_n)}{f'(y_n)} \\ \quad + \dfrac{1}{4}m(m^3+3m^2+2m-4)\dfrac{f(x_n)}{f'(x_n)}. \end{cases} \tag{12}$$

Case 3. Let

$$Q(x) = A + \frac{B}{x} + \frac{C}{x^2}.$$

Then

$$Q'(x) = -\frac{2C}{x^3} - \frac{B}{x^2}, \qquad Q''(x) = \frac{6C}{x^4} + \frac{2B}{x^3}.$$

From Theorem 2.1, we have the following equations:

$$\begin{cases} A + \dfrac{B}{u} + \dfrac{C}{u^2} = m, \\ -\dfrac{2C}{u^3} - \dfrac{B}{u^2} = -\dfrac{1}{4} m^{3-m} (2+m)^m, \\ \dfrac{6C}{u^4} + \dfrac{2B}{u^3} = \dfrac{1}{4} m^4 \left(\dfrac{m}{2+m} \right)^{-2m}. \end{cases}$$

The solutions are

$$A = \frac{m}{8}(m^3 - 4m + 8), B = -\frac{m}{4}(m-1)(m+2)^2 \left(\frac{m}{m+2} \right)^m, C = \frac{m}{8}(m+2)^3 \left(\frac{m}{m+2} \right)^{2m}.$$

Thus we have the fourth-order iterative method given by (5), proposed in [12].

Case 4. Let

$$Q(x) = \frac{A}{x} + \frac{1}{B + Cx}.$$

Then

$$Q'(x) = -\frac{A}{x^2} - \frac{C}{(B+Cx)^2}, \qquad Q''(x) = \frac{2A}{x^3} + \frac{2C^2}{(B+Cx)^3}.$$

Similarly, we have

$$\begin{cases} \dfrac{A}{u} + \dfrac{1}{B+Cu} = m, \\ -\dfrac{A}{u^2} - \dfrac{C}{(B+Cu)^2} = -\dfrac{1}{4} m^{3-m} (2+m)^m, \\ \dfrac{2A}{u^3} + \dfrac{2C^2}{(B+Cu)^3} = \dfrac{1}{4} m^4 \left(\dfrac{m}{2+m} \right)^{-2m}. \end{cases}$$

Thus we get

$$\begin{cases} A = -\dfrac{1}{2} \dfrac{m(m-2)(m+2)^3 \left(\frac{m}{m+2}\right)^m}{(m^3 - 4m + 8)}, \\ B = -\dfrac{(m^3 - 4m + 8)^2}{m(m^2 + 2m - 4)^3}, \\ C = \dfrac{m^2 (m^3 - 4m + 8) \left(\frac{m+2}{m}\right)^m}{(m^2 + 2m - 4)^3}, \end{cases}$$

which gives the fourth-order convergent method (6), proposed in [9].

Case 5. Let

$$Q(x) = \frac{B + Cx}{1 + Ax}.$$

Then

$$Q'(x) = \frac{C - AB}{(1+Ax)^2} \quad \text{and} \quad Q''(x) = \frac{2A(AB - C)}{(1+Ax)^3}.$$

According to Theorem 2.1, we should solve the following equations:

$$\begin{cases} \dfrac{B + Cu}{1 + Au} = m, \\ \dfrac{C - AB}{(1+Au)^2} = -\dfrac{1}{4} m^{3-m} (2+m)^m, \\ \dfrac{2A(AB - C)}{(1+Au)^3} = \dfrac{1}{4} m^4 \left(\dfrac{m}{2+m} \right)^{-2m}. \end{cases}$$

Thus we have

$$A = -\left(\frac{m+2}{m} \right)^m, \qquad B = -\frac{m^2}{2}, \qquad C = \frac{1}{2} m(m-2) \left(\frac{m+2}{m} \right)^m,$$

Table 1
The number of iterations and function evaluations.

$f(x)$	x_0	(2)	(7)	(11)	(12)
f_1	2.5	6(24)	6(18)	6(18)	6(18)
	3.5	6(24)	6(18)	7(21)	7(21)
f_2	1.8	4(16)	4(12)	4(12)	4(12)
	−2.0	4(16)	4(12)	4(12)	4(12)
f_3	1.5	4(16)	4(12)	4(12)	4(12)
	3.0	41(164)	×	7(21)	8(24)
f_4	−3.5	11(44)	10(30)	10(30)	10(30)
	1.2	×	×	28(84)	77(231)
f_5	3.25	6(24)	5(15)	6(18)	6(18)
	4.25	14(56)	13(39)	13(39)	13(39)
f_6	0.5	5(20)	4(12)	4(12)	4(12)
	15.5	5(20)	4(12)	3(12)	3(12)

Table 2
Value of $|f(x)|$ when the stopping criterion is satisfied.

$f(x)$	x_0	(2)	(7)	(11)	(12)
f_1	2.5	2.6021 e−132	6.0210e−132	5.2705e−127	4.4551e−129
	3.5	3.2011 e−119	3.2011e−119	5.4282e−139	4.2814e−149
f_2	1.8	2.4866 e−130	2.6720e−171	5.5482e−158	2.4977e−173
	−2.0	4.3153 e−129	5.7302e−178	8.8010e−164	2.0963e−181
f_3	1.5	8.6463 e−129	8.6106e−125	3.3436e−126	2.3140e−125
	3.0	7.7305 e−139	×	7.0380e−162	9.6051e−163
f_4	−3.5	2.1541 e−130	5.8668e−159	2.9531e−141	1.5017e−146
	1.2	×	×	7.0971e−167	5.0241e−140
f_5	3.25	6.7913 e−122	1.1760e−120	7.5369e−174	1.5180e−176
	4.25	1.0147 e−124	1.6017e−162	1.3409e−138	6.4336e−144
f_6	0.5	7.5584 e−144	1.9930e−143	1.4892e−138	2.3654e−146
	15.5	2.8011 e−150	5.7295e−121	9.8206e−176	4.0287e−119

and the corresponding method has been proposed in [8], which is also equivalent to (7).

4. Numerical results

In this section, we employ the presented fourth-order methods (8), including (7), (11), and (12), to solve some nonlinear equations and compare them with another fourth-order method (2). All numerical computations have been carried out in a Matlab 7.0 environment using 128- digit floating-point arithmetic. The following test problems have been used with the stopping criterion $|f(x_{n+1})| \le 10^{-120}$, where $x^\star$ is a root of $f(x)$ with multiplicity m.

$f(x)$	$x^\star$	m
$f_1(x) = (\sin^2 x - x^2 + 1)^2$	1.4044916482153412260350868178	2
$f_2(x) = (x^2 - e^x - 3x + 2)^5$	0.2575302854398607604553673049	5
$f_3(x) = (\cos x - x)^3$	0.7390851332151606416553120876	3
$f_4(x) = (xe^{x^2} - \sin^2 x + 3\cos x + 5)^4$	−1.2076478271309189270094167584	4
$f_5(x) = (e^{x^2+7x-30} - 1)^4$	3.0	4
$f_6(x) = \left(\ln x + \sqrt{x} - 5\right)^4$	8.3094326942315717953469556827	4

In Tables 1 and 2, "×" means that the method does not converge to the solution to the corresponding test function. It can be seen that the new presented methods (11) and (12) are superior to method (2). For the test function f_4, starting from the initial point 1.2, methods (2) and (7) both failed. However, two new iterative methods, (11) and (12), perform well. So Table 1 shows that our present methods can compete with method (2) and require fewer iterative steps, especially the number of function evaluations. From Table 2, we can see that, even with fewer iterative steps, the present methods can also obtain high-precision solutions, and thus they are more suitable for high-precision computation.

Acknowledgement

The authors would like to thank the referees for their helpful comments and suggestions, which greatly improved the original manuscript.

References

[1] E. Schröder, Über unendlich viele Algorithmen zur Auflösung der Gleichungen, Math. Ann. 2 (1870) 317–365.
[2] C. Dong, A basic theorem of constructing an iterative formula of the higher order for computing multiple roots of an equation, Math. Numer. Sin. 11 (1982) 445–450.
[3] C. Dong, A family of multipoint iterative functions for finding multiple roots of equations, Int. J. Comput. Math. 21 (1987) 363–367.
[4] H.D. Victory, B. Neta, A higher order method for multiple zeros of nonlinear functions, Int. J. Comput. Math. 12 (1983) 329–335.
[5] B. Neta, A.N. Johnson, High-order nonlinear solver for multiple roots, Comput. Math. Appl. 55 (2008) 2012–2017.
[6] B. Neta, New third order nonlinear solvers for multiple roots, Appl. Math. Comput. 202 (2008) 162–170.
[7] B. Neta, Extension of Murakami's high order nonlinear solver to multiple roots, Int. J. Comput. Math. 87 (2010) 1023–1031.
[8] S. Li, X. Liao, L. Cheng, A new fourth-order iterative method for finding multiple roots of nonlinear equations, Appl. Math. Comput. 215 (2009) 1288–1292.
[9] S. Li, L. Cheng, B. Neta, Some fourth-order nonlinear solvers with closed formulae for multiple roots, Comput. Math. Appl. 59 (2010) 126–135.
[10] P. Jarratt, Multipoint iterative methods for solving certain equations, Comput. J. 8 (1966) 398–400.
[11] P. Jarratt, Some efficient fourth order multipoint methods for solving equations, BIT 9 (1969) 119–124.
[12] J.R. Sharma, R. Sharma, Modified Jarratt method for computing multiple roots, Appl. Math. Comput. 217 (2010) 878–881.
[13] H.T. Kung, J.F. Traub, Optimal order of one-point and multipoint iteration, J. Assoc. Comput. Mach. 21 (1974) 643–651.