ARTICLE IN PRESS

Computers and Mathematics with Applications ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

New families of nonlinear third-order solvers for finding multiple roots

Changbum Chun [a], Hwa ju Bae [a], Beny Neta [b,*]

[a] Department of Mathematics, Sungkyunkwan University, Suwon 440-746, Republic of Korea

[b] Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA 93943, United States

ARTICLE INFO

Article history:
Received 16 September 2008
Accepted 16 October 2008

Keywords:
Newton's method
Multiple roots
Iterative methods
Nonlinear equations
Order of convergence
Root-finding

ABSTRACT

In this paper, we present two new families of iterative methods for multiple roots of nonlinear equations. One of the families require one-function and two-derivative evaluation per step, and the other family requires two-function and one-derivative evaluation. It is shown that both are third-order convergent for multiple roots. Numerical examples suggest that each family member can be competitive to other third-order methods and Newton's method for multiple roots. In fact the second family is even better than the first.

Published by Elsevier Ltd

1. Introduction

Solving nonlinear equations is one of the most important problems in numerical analysis. In this paper, we consider iterative methods to find a multiple root α of multiplicity m, i.e., $f^{(j)}(\alpha) = 0,\ j = 0, 1, \ldots, m-1$ and $f^{(m)}(\alpha) \neq 0$, of a nonlinear equation $f(x) = 0$.

The well known Newton's method for finding a multiple root α is given by

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

which converges quadratically [1].

There exists a cubically convergent Halley method [2] which Hansen and Patrick [3] extended to multiple roots, which is given by

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

In recent years, some modifications of the Newton method for multiple roots have been proposed and analyzed, see for example [4–14] and references therein. These methods have been proven to be competitive to Newton's method in their performance and efficiency. There are, however, not yet so many methods known in open literature that can handle the case of multiple roots, see [13]. To deal with the multiple roots case, one may use the observation that the functions

$$u = \frac{f}{f'}, \qquad f^{(m-1)}, \qquad f^{1/m}$$

* Corresponding author.
E-mail addresses: cbchun@skku.edu (C. Chun), bneta@nps.edu, byneta@gmail.com (B. Neta).

0898-1221/$ – see front matter. Published by Elsevier Ltd
doi:10.1016/j.camwa.2008.10.070

have only a simple zero at α, and any of the iterative methods for a simple zero may be used [15]. However, this approach might become problematic due to higher computational costs. This being the case, development of iterative methods to approximate a multiple root is required, and this is our motivation for this work.

In this paper, we present two new third-order families of methods for multiple roots. The first one is based on the composition of Osada's third-order multiple root-finding method [4]

$$x_{n+1} = x_n - \frac{1}{2}m(m+1)\frac{f(x_n)}{f'(x_n)} + \frac{1}{2}(m-1)^2\frac{f'(x_n)}{f''(x_n)}, \tag{3}$$

and the Euler–Chebyshev third-order multiple root-finding method [15]

$$x_{n+1} = x_n - \frac{m(3-m)}{2}\frac{f(x_n)}{f'(x_n)} - \frac{m^2}{2}\frac{f(x_n)^2 f''(x_n)}{f'(x_n)^3}. \tag{4}$$

This family proposed here is shown to be locally cubically convergent. Its performance is often a little better than the two third-order methods from which this family is derived, and its practical utility is demonstrated by numerical examples.

The other method is based on one of the third order methods due to Dong [5], i.e.

$$y_n = x_n - u_n, \tag{5}$$

$$x_{n+1} = y_n + \frac{u_n f(y_n)}{f(y_n) - (1-\frac{1}{m})^{m-1} f(x_n)}, \tag{6}$$

where

$$u_n = \frac{f(x_n)}{f'(x_n)} \tag{7}$$

and a third order method due to Victory and Neta [6], i.e.

$$y_n = x_n - u_n, \tag{8}$$

$$x_{n+1} = y_n - \frac{f(y_n)}{f'(x_n)}\frac{f(x_n) + Af(y_n)}{f(x_n) + Bf(y_n)}, \tag{9}$$

where

$$A = \mu^{2m} - \mu^{m+1}, \tag{10}$$

$$B = -\frac{\mu^m(m-2)(m-1)+1}{(m-1)^2}, \tag{11}$$

$$\mu = \frac{m}{m-1}. \tag{12}$$

This family is also of third order but requires two-function and one-derivative evaluations.

2. Development of methods and convergence analysis

To derive the first method, let us consider the composition of the methods (3) and (4) in the form

$$\begin{aligned} x_{n+1} = {} & x_n - \frac{\theta}{2}\left[m(m+1)\frac{f(x_n)}{f'(x_n)} - (m-1)^2\frac{f'(x_n)}{f''(x_n)}\right] \\ & - \frac{(1-\theta)}{2}\left[m(3-m)\frac{f(x_n)}{f'(x_n)} + m^2\frac{f(x_n)^2 f''(x_n)}{f'(x_n)^3}\right], \end{aligned} \tag{13}$$

where $\theta \in \mathbf{R}$, from which we suggest the following one-parameter family of methods for multiple roots

$$x_{n+1} = x_n - \frac{m[(2\theta-1)m + 3 - 2\theta]}{2}\frac{f(x_n)}{f'(x_n)} + \frac{\theta(m-1)^2}{2}\frac{f'(x_n)}{f''(x_n)} - \frac{(1-\theta)m^2}{2}\frac{f(x_n)^2 f''(x_n)}{f'(x_n)^3}, \tag{14}$$

For the family of methods defined by (14), we have the following analysis of convergence.

Theorem 2.1. *Let $\alpha \in I$ be a multiple root of multiplicity m of a sufficiently differentiable function $f : I \to \mathbf{R}$ for an open interval I. If x_0 is sufficiently close to α, then the methods defined by (14) are cubically convergent for any real value of θ, and satisfies the error equation*

$$e_{n+1} = K_3 e_n^3 + O(e_n^4), \tag{15}$$

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

where $e_n = x_n - \alpha$ and the error constant K_3 is given by

$$K_3 = (\theta - 1)\frac{f^{(m+1)}(\alpha)}{f^{(m)}(\alpha)} + \gamma\frac{f^{(m+1)}(\alpha)^2}{f^{(m)}(\alpha)^2} - \frac{1}{(m+2)(m+1)m}\frac{f^{(m+2)}(\alpha)}{f^{(m)}(\alpha)}, \tag{16}$$

$$\gamma = \frac{2(1-\theta)m^5 + 2(\theta-1)m^4 + (2\theta-1)m^3 + (10\theta-9)m^2 + (19-20\theta)m + 8\theta - 9}{2(m+1)^2m^2(m-1)^2}. \tag{17}$$

Proof. Using Taylor expansion of $f(x_n)$ about α, we have

$$f(x_n) = \frac{f^{(m)}(\alpha)}{m!}e_n^m\left[1 + \bar{C}_1e_n + \bar{C}_2e_n^2 + O(e_n^3)\right], \tag{18}$$

$$f(x_n)^2 = \frac{f^{(m)}(\alpha)^2}{(m!)^2}e_n^{2m}\left[1 + 2\bar{C}_1e_n + [\bar{C}_1^2 + 2\bar{C}_2]e_n^2 + O(e_n^3)\right], \tag{19}$$

$$f'(x_n) = \frac{f^{(m)}(\alpha)}{(m-1)!}e_n^{m-1}\left[1 + \bar{D}_1e_n + \bar{D}_2e_n^2 + O(e_n^3)\right], \tag{20}$$

$$f''(x_n) = \frac{f^{(m)}(\alpha)}{(m-2)!}e_n^{m-2}\left[1 + \bar{S}_1e_n + \bar{S}_2e_n^2 + O(e_n^3)\right], \tag{21}$$

$$f(x_n)^2f''(x_n) = \frac{f^{(m)}(\alpha)^3}{(m-2)!(m!)^2}e_n^{3m-2}[1 + (2\bar{C}_1 + \bar{S}_1)e_n + (\bar{C}_1^2 + 2\bar{C}_2 + 2\bar{S}_1\bar{C}_1 + \bar{S}_2)e_n^2 + O(e_n^3)], \tag{22}$$

$$f'(x_n)^3 = \frac{f^{(m)}(\alpha)^3}{[(m-1)!]^3}e_n^{3m-3}\left[1 + 3\bar{D}_1e_n + (3\bar{D}_1^2 + 3\bar{D}_2)e_n^2 + O(e_n^3)\right], \tag{23}$$

where $\bar{C}_j = \frac{m!}{(m+j)!}\frac{f^{(m+j)}(\alpha)}{f^{(m)}(\alpha)}$, $\bar{D}_j = \frac{(m-1)!}{(m+j-1)!}\frac{f^{(m+j)}(\alpha)}{f^{(m)}(\alpha)}$, and $\bar{S}_j = \frac{(m-2)!}{(m+j-2)!}\frac{f^{(m+j)}(\alpha)}{f^{(m)}(\alpha)}$.

Dividing (18), (20) and (22) by (20), (21) and (23), respectively give us

$$\frac{f(x_n)}{f'(x_n)} = \frac{e_n}{m}\left[1 + (\bar{C}_1 - \bar{D}_1)e_n + (\bar{C}_2 - \bar{D}_2 - \bar{C}_1\bar{D}_1 + \bar{D}_1^2)e_n^2 + O(e_n^3)\right], \tag{24}$$

$$\frac{f'(x_n)}{f''(x_n)} = \frac{e_n}{m-1}\left[1 + (\bar{D}_1 - \bar{S}_1)e_n + (\bar{D}_2 - \bar{S}_2 - \bar{D}_1\bar{S}_1 + \bar{S}_1^2)e_n^2 + O(e_n^3)\right], \tag{25}$$

$$\begin{aligned}\frac{f(x_n)^2f''(x_n)}{f'(x_n)^3} &= \frac{m-1}{m^2}e_n[1 + (2\bar{C}_1 + \bar{S}_1 - 3\bar{D}_1)e_n \\ &\quad + (\bar{C}_1^2 + 2\bar{C}_2 + 2\bar{C}_1\bar{S}_1 + \bar{S}_2 + 6\bar{D}_1^2 - 3\bar{D}_2 - 6\bar{C}_1\bar{D}_1 - 3\bar{D}_1\bar{S}_1)e_n^2 + O(e_n^3)].\end{aligned} \tag{26}$$

Now, substituting (24)–(26) into the error equation

$$e_{n+1} = e_n - \frac{m[(2\theta-1)m + 3 - 2\theta]}{2}\frac{f(x_n)}{f'(x_n)} + \frac{\theta(m-1)^2}{2}\frac{f'(x_n)}{f''(x_n)} - \frac{(1-\theta)m^2}{2}\frac{f(x_n)^2f''(x_n)}{f'(x_n)^3}, \tag{27}$$

we have

$$e_{n+1} = K_1e_n + K_2e_n^2 + K_3e_n^3 + O(e_n^4) \tag{28}$$

where

$$K_1 = 1 - \frac{[(2\theta-1)m + 3 - 2\theta]}{2} + \frac{\theta(m-1)}{2} - \frac{(1-\theta)(m-1)}{2} = 0, \tag{29}$$

and

$$\begin{aligned}K_2 &= -\frac{[(2\theta-1)m + 3 - 2\theta]}{2}(\bar{C}_1 - \bar{D}_1) + \frac{\theta(m-1)}{2}(\bar{D}_1 - \bar{S}_1) \\ &\quad - \frac{(1-\theta)(m-1)}{2}(2\bar{C}_1 + \bar{S}_1 - 3\bar{D}_1),\end{aligned} \tag{30}$$

which, after some simplification, can easily be shown to be zero.

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

The coefficient K_3 is given by,

$$K_3 = -\frac{[(2\theta-1)m+3-2\theta]}{2}(\bar{C}_2 - \bar{D}_2 - \bar{C}_1\bar{D}_1 + \bar{D}_1^2) + \frac{\theta(m-1)}{2}(\bar{D}_2 - \bar{S}_2 - \bar{D}_1\bar{S}_1 + \bar{S}_1^2)$$
$$-\frac{(1-\theta)(m-1)}{2}(\bar{C}_1^2 + 2\bar{C}_2 + 2\bar{S}_1\bar{C}_1 + \bar{S}_2 + 6\bar{D}_1^2 - 3\bar{D}_2 - 6\bar{C}_1\bar{D}_1 - 3\bar{S}_1\bar{D}_1). \tag{31}$$

By a simple manipulation, it may be shown that (31) reduces to

$$K_3 = (\theta-1)\frac{f^{(m+1)}(\alpha)}{f^{(m)}(\alpha)} + \gamma\frac{f^{(m+1)}(\alpha)^2}{f^{(m)}(\alpha)^2} - \frac{1}{(m+2)(m+1)m}\frac{f^{(m+2)}(\alpha)}{f^{(m)}(\alpha)}, \tag{32}$$

$$\gamma = \frac{2(1-\theta)m^5 + 2(\theta-1)m^4 + (2\theta-1)m^3 + (10\theta-9)m^2 + (19-20\theta)m + 8\theta - 9}{2(m+1)^2m^2(m-1)^2}. \tag{33}$$

Therefore, we obtain

$$e_{n+1} = K_3e_n^3 + O(e_n^4), \tag{34}$$

which indicates that the order of convergence of the methods defined by (14) is at least three. This completes the proof. □

The family (14) includes, as particular cases, the following.
For $\theta = 1$ and $\theta = 0$, we obtain the Osada method (3) and the Euler–Chebyshev method (4), respectively.
For $\theta = 1/2$, we obtain a new third-order method for multiple roots

$$x_{n+1} = x_n - m\frac{f(x_n)}{f'(x_n)} + \frac{(m-1)^2}{4}\frac{f'(x_n)}{f''(x_n)} - \frac{m^2}{4}\frac{f(x_n)^2f''(x_n)}{f'(x_n)^3}, \tag{35}$$

For $\theta = -1$, we obtain another new third-order method for multiple roots

$$x_{n+1} = x_n - \frac{1}{2}m(5-3m)\frac{f(x_n)}{f'(x_n)} - \frac{1}{2}(m-1)^2\frac{f'(x_n)}{f''(x_n)} - m^2\frac{f(x_n)^2f''(x_n)}{f'(x_n)^3}. \tag{36}$$

In a similar fashion, the proposed approach can be continuously applied to other multiple roots iterative methods, to derive methods for multiple roots. Clearly if we combine two different methods using the exact same information, then the efficiency (see [15]) will not be affected.

The second family is based on a composition of Dong's third-order method (6) and the Victory–Neta third-order scheme (9)

$$y_n = x_n - u_n, \tag{37}$$

$$x_{n+1} = y_n + \theta\frac{u_nf(y_n)}{f(y_n) - (1-\frac{1}{m})^{m-1}f(x_n)} - (1-\theta)\frac{f(y_n)}{f'(x_n)}\frac{f(x_n)+Af(y_n)}{f(x_n)+Bf(y_n)}. \tag{38}$$

The scheme (38) reduces to Dong's method if $\theta = 1$ and the Victory–Neta third-order scheme if $\theta = 0$. Neta [13] has shown that Dong's method (6) has the following error term

$$e_{n+1} = \left(\frac{1}{2m}\bar{C}_1^2 - \frac{m-1}{m^2}\bar{C}_2\right)e_n^3. \tag{39}$$

For the Victory–Neta scheme, the error term is given by

$$e_{n+1} = \left(K_1\bar{C}_1^2 - \frac{m-1}{m^2}\bar{C}_2\right)e_n^3 \tag{40}$$

where

$$K_1 = \frac{(m^2+2)(m-1) - m(m+2)\mu^{-m}}{2m^3(m-1-\mu^{-m})}. \tag{41}$$

Notice that the composite scheme (38) will have the following error term

$$e_{n+1} = \left((1-\theta)K_1 + \theta\frac{1}{2m}\right)\bar{C}_1^2e_n^3 \tag{42}$$

i.e. the terms with $\bar{C}_2$ were annihilated.

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

Table 1
Comparison of various third-order multiple-roots iterative schemes and the Newton method.

	IT	NFE	$f(x_*)$
$f_1, x_0 = 3$			
NM	7	14	6.10e−56
HM	5	15	1.07e−81
OM	5	15	1.30e−46
ECM	5	15	1.79e−57
CM1	5	15	1.35e−51
CM2	5	15	4.11e−74
NM1	1	3	4.11e−74
NM2	1	3	4.11e−74
$f_1, x_0 = -1$			
NM	25	50	3.63e−63
HM	10	30	−1.84e−39
OM	15	45	4.75e−56
ECM	17	51	−3.29e−35
CM1	15	45	7.27e−67
CM2	7	21	9.94e−41
NM1	1	3	9.94e−41
NM2	1	3	9.94e−41
$f_2, x_0 = 2.3$			
NM	7	14	7.31e−52
HM	5	15	4.84e−57
OM	5	15	2.07e−38
ECM	5	15	1.73e−47
CM1	5	15	2.25e−42
CM2	5	15	7.14e−70
NM1	5	15	3.11e−62
NM2	5	15	1.10e−54
$f_2, x_0 = 2$			
NM	7	14	5.11e−64
HM	5	15	7.43e−77
OM	5	15	3.53e−51
ECM	5	15	1.53e−63
CM1	5	15	1.44e−56
CM2	5	15	2.44e−98
NM1	5	15	1.06e−80
NM2	5	15	9.31e−71
$f_3, x_0 = 0$			
NM	4	8	1.03e−55
HM	3	9	1.68e−53
OM	3	9	5.83e−62
ECM	3	9	4.31e−58
CM1	3	9	8.95e−60
CM2	3	9	1.49e−55
NM1	1	3	1.49e−55
NM2	1	3	1.49e−55
$f_3, x_0 = 1$			
NM	4	8	3.46e−52
HM	4	12	1.39e−85
OM	4	12	2.01e−91
ECM	4	12	2.24e−89
CM1	4	12	2.24e−90
CM2	4	12	1.68e−87
NM1	1	3	1.68e−87
NM2	1	3	1.68e−87
$f_4, x_0 = 1.7$			
NM	5	10	6.04e−47
HM	4	12	9.12e−43
OM	4	12	1.17e−39
ECM	4	12	5.25e−41
CM1	4	12	−2.69e−40
CM2	4	12	9.74e−43
NM1	1	3	−9.74e−43
NM2	1	3	−9.74e−43

(*continued on next page*)

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

Table 1 (*continued*)

	IT	NFE	$f(x_*)$
$f_4, x_0 = 1$			
NM	5	10	1.22e−60
HM	4	12	1.78e−85
OM	4	12	1.42e−78
ECM	4	12	1.43e−81
CM1	4	12	−5.35e−80
CM2	4	12	2.59e−85
NM1	1	3	−2.59e−85
NM2	1	3	−9.74e−43
$f_5, x_0 = 3$			
NM	6	12	2.70e−45
HM	4	12	7.44e−45
OM	5	15	3.12e−85
ECM	5	15	1.89e−94
CM1	5	15	1.16e−89
CM2	4	12	6.54e−34
NM1	1	3	6.54e−34
NM2	1	3	−9.74e−43
$f_5, x_0 = -1$			
NM	10	20	5.23e−49
HM	11	33	2.22e−65
OM	24	72	7.70e−44
ECM	23	69	1.87e−52
CM1	26	78	4.35e−67
CM2	32	96	8.76e−49
NM1	1	3	8.76e−49
NM2	1	3	8.76e−49
$f_6, x_0 = -2$			
NM	8	16	5.60e−37
HM	5	15	1.60e−61
OM	6	18	5.09e−45
ECM	6	18	3.21e−64
CM1	6	18	8.66e−54
CM2	6	18	7.73e−94
NM1	1	3	7.73e−94
NM2	1	3	7.73e−94
$f_6, x_0 = -1$			
NM	6	12	5.61e−60
HM	3	9	4.75e−35
OM	5	15	1.56e−103
ECM	4	12	1.47e−47
CM1	4	12	1.05e−38
CM2	4	12	1.98e−89
NM1	1	3	1.98e−89
NM2	1	3	1.981e−89
$f_7, x_0 = 1.7$			
NM	6	12	3.80e−57
HM	4	12	7.40e−47
OM	5	15	1.81e−76
ECM	4	12	1.01e−37
CM1	5	15	8.04e−90
CM2	4	12	2.26e−44
NM1	1	3	2.26e−44
NM2	1	3	2.26e−44
$f_7, x_0 = 2$			
NM	5	10	2.09e−40
HM	4	12	1.55e−65
OM	4	12	3.45e−53
ECM	4	12	1.67e−59
CM1	4	12	6.64e−56
CM2	4	12	6.75e−77
NM1	1	3	6.75e−77
NM2	1	3	6.75e−7

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

Table 1 (*continued*)

	IT	NFE	$f(x_*)$
$f_8, x_0 = 4$			
NM	6	12	1.18e−61
HM	4	12	2.33e−60
OM	4	12	2.70e−36
ECM	4	12	1.82e−39
CM1	4	12	7.80e−38
CM2	4	12	4.85e−43
NM1	1	3	4.85e−43
NM2	1	3	4.85e−43
$f_8, x_0 = 3$			
NM	5	10	1.10e−54
HM	4	12	2.10e−113
OM	4	12	6.75e−80
ECM	4	12	1.57e−84
CM1	4	12	3.83e−82
CM2	4	12	8.47e−90
NM1	1	3	8.47e−90
NM2	1	3	8.47e−90
$f_9, x_0 = 3.5$			
NM	12	24	1.17e−46
HM	7	21	2.07e−57
OM	9	27	1.63e−61
ECM	8	24	8.16e−36
CM1	9	27	7.39e−82
CM2	8	24	2.76e−61
NM1	1	3	2.76e−61
NM2	1	3	2.76e−61
$f_9, x_0 = 4.5$			
NM	27	54	1.62e−44
HM	15	45	2.19e−100
OM	20	60	1.58e−64
ECM	18	54	2.33e−34
CM1	19	57	1.77e−48
CM2	17	51	8.95e−46
NM1	1	3	8.95e−46
NM2	1	3	8.95e−46
$f_{10}, x_0 = 11$			
NM	5	10	−2.38e−65
HM	3	9	2.34e−36
OM	3	9	1.7e−36
ECM	3	9	4.53e−53
CM1	3	9	2.50e−40
CM2	3	9	4.46e−36
NM1	1	3	4.46e−36
NM2	1	3	4.46e−36
$f_{10}, x_0 = 7$			
NM	5	10	−1.00e−48
HM	4	12	3.04e−79
OM	4	12	−1.06e−82
ECM	3	9	−3.18e−45
CM1	4	12	−5.98e−95
CM2	4	12	9.29e−82
NM1	1	3	9.29e−82
NM2	1	3	9.29e−82
$f_{11}, x_0 = 3.5$			
NM	6	12	6.67e−33
HM	4	12	2.48e−38
OM	5	15	1.18e−58
ECM	5	15	2.44e−80
CM1	5	15	1.33e−67
CM2	4	12	4.20e−36
NM1	1	3	4.20e−36
NM2	1	3	4.20e−36

(*continued on next page*)

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

Table 1 (*continued*)

	IT	NFE	$f(x_*)$
$f_{11}, x_0 = 5$			
NM	8	16	6.45e−38
HM	5	15	2.13e−42
OM	6	12	9.04e−36
ECM	6	12	6.53e−64
CM1	1	3	1.64e−47
CM2	5	15	5.30e−35
NM1	1	3	5.30e−35
NM2	1	3	5.30e−35
$f_{12}, x_0 = 6$			
NM	5	10	9.91e−60
HM	3	9	1.29e−33
OM	4	12	4.82e−102
ECM	3	9	9.08e−40
CM1	3	9	4.95e−35
CM2	3	9	3.80e−40
NM1	1	3	3.80e−40
NM2	1	3	3.80e−40
$f_{12}, x_0 = 11$			
NM	5	10	1.42e−59
HM	3	9	4.48e−36
OM	3	9	4.72e−33
ECM	3	9	1.63e−41
CM1	3	9	4.57e−36
CM2	3	9	2.76e−39
NM1	1	3	2.76e−39
NM2	1	3	2.76e−39

3. Numerical examples

We present some numerical test results for various third-order multiple root-finding methods and the Newton method in Table 1. Methods compared included the Newton method (1) (NM), Halley-like method (2) (HM), Osada's method (3) (OM), the Euler–Chebyshev method (4) (ECM), the methods (35) (CM1), (36) (CM2), (38) with $\theta = 1/2$ (NM1) and (38) with $\theta = -1$ (NM2) introduced in this contribution.

All computations were done using the MAPLE using 128 digit floating point arithmetics (Digits := 128).

The following functions are used for the comparison and we display the approximate zeros x_* found, up to the 28th decimal place

Function	m	x_*
$f_1(x) = (x^3 + 4x^2 - 10)^3$	$m = 3$	1.36523001341409684576080682 90
$f_2(x) = (\sin^2 x - x^2 + 1)^2$	$m = 2$	1.40449164821534122603508681 78
$f_3(x) = (x^2 - e^x - 3x + 2)^5$	$m = 5$	0.25753028543986076045536730494
$f_4(x) = (\cos x - x)^3$	$m = 3$	0.73908513321516064165531208767
$f_5(x) = ((x-1)^3 - 1)^6$	$m = 6$	2
$f_6(x) = (xe^{x^2} - \sin^2 x + 3\cos x + 5)^4$	$m = 4$	−1.20764782713091892700941675 84
$f_7(x) = (\sin x - x/2)^2$	$m = 2$	1.89549426703398094714403573 81
$f_8(x) = (x^3 - 10)^8$	$m = 8$	2.15443469003188372175929356 65
$f_9(x) = (e^{x^2+7x-30} - 1)^4$	$m = 4$	3.0
$f_{10}(x) = (\sqrt{x} - \frac{1}{x} - 3)^3$	$m = 3$	9.63359556283269519240631270 92
$f_{11}(x) = (e^x + x - 20)^2$	$m = 2$	2.84243895378444706781658594 02
$f_{12}(x) = (\ln(x) + \sqrt{x} - 5)^4$	$m = 4$	8.30943269423157179534695568 27

Displayed in Table 1 are the number of iterations (IT) required, such that $|f(x_n)| < 10^{-32}$, the number of functional evaluations (NFE) counted as the sum of the number of evaluations of the function itself plus the number of evaluations of the derivative, and the value $f(x_*)$ after required iterations.

The results presented in Table 1 show that, for the functions we tested, the first family introduced here has at least an equal performance when compared to the other multiple root-finding methods of the same order. It is often a little better than the two third-order methods from which it was derived, and converges more rapidly than Newton's method for multiple roots. The second family is much better because, in all cases apart from NM1 and NM2, it requires only one iteration to achieve the required accuracy.

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070

References

[1] E. Schröder, Über unendlich viele Algorithmen zur Auflösung der Gleichungen, Math. Ann. 2 (1870) 317–365.
[2] E. Halley, A new, exact and easy method of finding the roots of equations generally and without any previous reduction, Phil. trans. R. Soc. London 18 (1694) 136–148.
[3] E. Hansen, M. Patrick, A family of root finding methods, Numer. Math. 27 (1977) 257–269.
[4] N. Osada, An optimal multiple root-finding method of order three, J. Comput. Appl. Math. 51 (1994) 131–133.
[5] C. Dong, A family of multipoint iterative functions for finding multiple roots, Int. J. Comput. Math. 21 (1987) 363–367.
[6] H.D. Victory, B. Neta, A higher order method for multiple zeros of nonlinear functions, Int. J. Comput. Math. 12 (1983) 329–335.
[7] M. Frontini, E. Sormani, Modified Newton's method with third-order convergence and multiple roots, J. Comput. Appl. Math. 156 (2003) 345–354.
[8] J. Kou, Y. Li, X. Wang, A composite fourth-order iterative method for solving non-linear equations, Appl. Math. Comput. 184 (2007) 471–475.
[9] B. Neta, Extension of Murakami's High order nonlinear solver to multiple roots, Int. J. Comput. Math. (in press).
[10] B. Neta, New Third Order Nonlinear Solvers for Multiple Roots, Appl. Math. Comput. 202 (2008) 162–170. doi:10.1016/j.amc.2008.01.031.
[11] C. Chun, B. Neta, A third-order modification of Newton's method for multiple roots, Appl. Math. Comput. AMC-S-08-01123 (submitted for publication).
[12] B. Neta, A.N. Jhonson, High-order nonlinear solver for multiple roots, Comput. Math. Appl. 55 (2008) 2012–2017.
[13] B. Neta, Numerical Methods for the Solution of Equations, Net-A-Sof, California, 1983.
[14] E. Hansen, M. Patrick, A family of root finding methods, Numer. Math. 27 (1977) 257–269.
[15] J.F. Traub, Iterative Methods for the Solution of Equations, Chelsea Publishing Company, New York, 1977.

Please cite this article in press as: C. Chun, et al., New families of nonlinear third-order solvers for finding multiple roots, Computers and Mathematics with Applications (2008), doi:10.1016/j.camwa.2008.10.070