



Calhoun: The NPS Institutional Archive
DSpace Repository

Faculty and Researchers

Faculty and Researchers' Publications

2008

New Third Order Nonlinear Solvers for Multiple Roots (uncorrected proof)

Neta, Beny

<https://hdl.handle.net/10945/39430>

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>



New third order nonlinear solvers for multiple roots

B. Neta

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

Abstract

Two third order methods for finding multiple zeros of nonlinear functions are developed. One method is based on Chebyshev's third order scheme (for simple roots) and the other is a family based on a variant of Chebyshev's which does not require the second derivative. Two other more efficient methods of lower order are also given. These last two methods are variants of Chebyshev's and Osada's schemes. The informational efficiency of the methods is discussed. All these methods require the knowledge of the multiplicity.

Published by Elsevier Inc.

Keywords: Nonlinear equations; High order; Multiple roots; Fixed point; Chebyshev; Osada

1. Introduction

There is a vast literature on the solution of nonlinear equations and nonlinear systems, see for example Ostrowski [1], Traub [2], Neta [3] and references there. Here we develop several high order fixed point type methods to approximate a multiple root. There are several methods for computing a zero ξ of multiplicity m of a nonlinear equation $f(x) = 0$, see Neta [3]. Newton's method is only of first order unless it is modified to gain the second order of convergence, see Rall [4] or Schröder [5]. This modification requires a knowledge of the multiplicity. Traub [2] has suggested to use any method for $f^{(m)}(x)$ or $g(x) = \frac{f(x)}{f'(x)}$. Any such method will require higher derivatives than the corresponding one for simple zeros. Also the first one of those methods require the knowledge of the multiplicity m . In such a case, there are several other methods developed by Hansen and Patrick [6], Victory and Neta [7], Dong [8,9], Neta and Johnson [10], Neta [11] and Werner [12]. Since in general one does not know the multiplicity, Traub [2] suggested a way to approximate it during the iteration.

For example, the quadratically convergent modified Newton's method is (see [5])

$$x_{n+1} = x_n - m \frac{f_n}{f'_n} \quad (1)$$

and the cubically convergent Halley's method [13] is a special case of the Hansen and Patrick's method [6]

E-mail address: bneta@nps.edu

$$x_{n+1} = x_n - \frac{f_n}{\frac{m+1}{2m} f'_n - \frac{f_n f''_n}{2f'_n}}, \tag{2}$$

where $f_n^{(i)}$ is short for $f^{(i)}(x_n)$. Another third order method was developed by Victory and Neta [7] and based on King's fifth order method (for simple roots) [14]

$$y_n = x_n - u_n, \tag{3}$$

$$x_{n+1} = y_n - \frac{f(y_n)}{f'_n} \frac{f_n + Af(y_n)}{f_n + Bf(y_n)},$$

where

$$A = \mu^{2m} - \mu^{m+1}, \tag{4}$$

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

$$\mu = \frac{m}{m-1}, \tag{5}$$

and

$$u_n = \frac{f_n}{f'_n}. \tag{6}$$

Dong [8] has developed two third order methods requiring two evaluations of f and one evaluation of f'

$$\begin{cases} y_n = x_n - \sqrt{m}u_n, \\ x_{n+1} = y_n - m\left(1 - \frac{1}{\sqrt{m}}\right)^{1-m} \frac{f(y_n)}{f'_n}, \end{cases} \tag{7}$$

$$\begin{cases} y_n = x_n - u_n, \\ x_{n+1} = y_n + \frac{u_n f(y_n)}{f(y_n) - (1 - \frac{1}{m})^{m-1} f_n}, \end{cases} \tag{8}$$

where u_n is given by (6).

Yet two other third order methods developed by Dong [9], both require the same information and both based on a family of fourth order methods (for simple roots) due to Jarratt [15]:

$$\begin{cases} y_n = x_n - u_n, \\ x_{n+1} = y_n - \frac{f_n}{\left(\frac{m}{m-1}\right)^{m+1} f'(y_n) + \frac{m-m^2-1}{(m-1)^2} f'_n}, \end{cases} \tag{9}$$

$$\begin{cases} y_n = x_n - \frac{m}{m+1} u_n, \\ x_{n+1} = y_n - \frac{\frac{m}{m+1} f_n}{\left(1 + \frac{1}{m}\right) f'(y_n) - f'_n}, \end{cases} \tag{10}$$

where u_n is given by (6).

Osada [16] has developed a third order method using the second derivative,

$$x_{n+1} = x_n - \frac{1}{2}m(m+1)u_n + \frac{1}{2}(m-1)^2 \frac{f'_n}{f''_n}, \tag{11}$$

where u_n is given by (6).

Neta and Johnson [10] have developed a fourth order method requiring one function- and three derivative-evaluation per step. The method is based on Jarratt's method [17] given by the iteration

$$x_{n+1} = x_n - \frac{f_n}{a_1 f'_n + a_2 f'(y_n) + a_3 f'(\eta_n)}, \tag{12}$$

where u_n is given by (6) and

$$\begin{aligned}
 y_n &= x_n - au_n, \\
 v_n &= \frac{f_n}{f'(y_n)}, \\
 \eta_n &= x_n - bu_n - cv_n.
 \end{aligned} \tag{13}$$

Neta and Johnson [10] give a table of values for the parameters a, b, c, a_1, a_2, a_3 for several values of m . In the case $m = 2$ they found a method that will require only two derivative-evaluations ($a_3 = 0$). This was not possible for higher m .

Neta [11] has developed a fourth order method requiring one function- and three derivative-evaluation per step. The method is based on Murakami's method [18] given by the iteration

$$x_{n+1} = x_n - a_1u_n - a_2v_n - a_3w_3(x_n) - \psi(x_n), \tag{14}$$

where u_n is given by (6), v_n, y_n , and η_n are given by (13) and

$$\begin{aligned}
 w_3(x_n) &= \frac{f_n}{f'(\eta_n)}, \\
 \psi(x_n) &= \frac{f_n}{b_1f'_n + b_2f'(y_n)}.
 \end{aligned} \tag{15}$$

Neta [11] gives a table of values for the parameters $a, b, c, a_1, a_2, a_3, b_1, b_2$ for several values of m .

A method of order 1.5 requiring two function- and one derivative-evaluation is given by Werner [12]. It is only for double roots

$$\begin{aligned}
 y_n &= x_n - u_n, \\
 x_{n+1} &= x_n - s_nu_n,
 \end{aligned} \tag{16}$$

where

$$s_n = \begin{cases} \frac{2}{1 + \sqrt{1 - 4f(y_n)/f_n}} & \text{if } f(y_n)/f_n \leq \frac{1}{4}, \\ \frac{1}{2}f_n/f(y_n) & \text{otherwise.} \end{cases}$$

Later we give a table comparing the efficiency of these methods and of our new ones we develop here.

2. A new third order scheme

We would like to develop a new method for multiple roots based on Chebyshev's method (see [19–21]). Chebyshev's method is given for simple roots by

$$x_{n+1} = x_n - u_n \left[1 + \frac{1}{2}u_n \frac{f''_n}{f'_n} \right]. \tag{17}$$

Let us take the following two-parameter family

$$x_{n+1} = x_n - \alpha u_n \left[1 + \beta u_n \frac{f''_n}{f'_n} \right]. \tag{18}$$

We now show how to choose the parameters α, β so that the method is of third order for the case of multiple roots.

Expand $f(x_n)$, and $f'(x_n)$ in Taylor series (truncated after the N th power, $N > m$) about the root ξ , we have (using a symbolic manipulator Maple [22])

$$f(x_n) = f(x_n - \xi + \xi) = f(\xi + e_n) = \frac{f^{(m)}(\xi)}{m!} \left(e_n^m + \sum_{i=m+1}^N A_i e_n^i \right) \tag{19}$$

or

$$f(x_n) = \frac{f^{(m)}(\xi)}{m!} e_n^m \left(1 + \sum_{i=m+1}^N B_{i-m} e_n^{i-m} \right), \tag{20}$$

where

$$A_i = \frac{m! f^{(i)}(\xi)}{i! f^{(m)}(\xi)}, \quad i > m, \tag{21}$$

$$B_{i-m} = A_i,$$

and

$$e_n = x_n - \xi, \tag{22}$$

$$f'(x_n) = \frac{f^{(m)}(\xi)}{(m-1)!} e_n^{m-1} \left(1 + \sum_{i=m+1}^N \frac{i}{m} B_{i-m} e_n^{i-m} \right). \tag{23}$$

Then u_n will be (neglecting terms of order higher than 3)

$$u_n = \frac{1}{m} e_n - \frac{B_1}{m^2} e_n^2 + \frac{(m+1)B_1^2 - 2mB_2}{m^3} e_n^3 \tag{24}$$

Now expand the second derivative and get $u_n \frac{f''_n}{f'_n}$

$$f''(x_n) = \frac{f^{(m)}(\xi)}{(m-2)!} e_n^{m-2} \left(1 + \frac{m+1}{m-1} B_1 e_n + \frac{(m+1)(m+2)}{m(m-1)} B_2 e_n^2 + \frac{(m+2)(m+3)}{m(m-1)} B_3 e_n^3 \right), \tag{25}$$

$$u_n \frac{f''_n}{f'_n} = \frac{m-1}{m} + \frac{2}{m^2} B_1 e_n - 3 \frac{(m+1)B_1^2 - 2mB_2}{m^3} e_n^2 + \frac{(m+3)(m+1)^2 B_1^3 - (3m+4)(m+3)mB_1 B_2 + 3m^2(m+3)B_3}{m^4} e_n^3. \tag{26}$$

Now substitute (24) and (26) in (18)

$$e_{n+1} = \frac{-\alpha m + m^2 + \alpha \beta (1-m)}{m^2} e_n + \frac{\alpha B_1 ((m-3)\beta + m)}{m^3} e_n^2 + \left[\frac{2\alpha(m + \beta(m-4))}{m^3} B_2 - \frac{\alpha(m(m+1) + \beta(m^2 - 3m - 6))}{m^4} B_1^2 \right] e_n^3. \tag{27}$$

If we choose $(m \neq 3)$

$$\alpha = -\frac{m(m-3)}{2}, \tag{28}$$

$$\beta = -\frac{m}{m-3},$$

we have the third order method

$$x_{n+1} = x_n + \frac{m(m-3)}{2} u_n \left[1 - \frac{m}{m-3} u_n \frac{f''_n}{f'_n} \right] \tag{29}$$

with the error at x_{n+1} becomes

$$e_{n+1} = \left(-\frac{1}{m} B_2 + \frac{3+m}{2m^2} B_1^2 \right) e_n^3. \tag{30}$$

Clearly the case $m = 3$ need to be considered separately. In this case the method is only second order with the choice

$$\beta = \frac{3}{2} \frac{3-\alpha}{\alpha} \tag{31}$$

43 and
44

$$45 \quad x_{n+1} = x_n - \alpha u_n \left[1 + \frac{3}{2} \frac{3-\alpha}{\alpha} u_n \frac{f_n'''}{f_n'} \right] \quad (32)$$

46 with the error at x_{n+1} becomes

$$47 \quad e_{n+1} = \frac{\alpha}{9} B_1 e_n^2 + \left(\frac{\alpha-1}{3} B_2 + \frac{9-7\alpha}{27} B_1^2 \right) e_n^3. \quad (33)$$

48 We can annihilate one of the terms in the coefficient of e_n^3 by choosing $\alpha = 1$ or $\alpha = \frac{9}{7}$, but the method is only
49 second order.

50 3. New methods not requiring second derivative

51 Here we develop 3 new methods not requiring second derivative. First is a third order method based on a
52 modification of Chebyshev's method. The one-parameter family of modified Chebyshev methods (see Kou
53 and Li [23]) is given by

$$54 \quad x_{n+1} = x_n - u_n \left[\frac{\theta^2 + \theta - 1}{\theta^2} + \frac{f(y_n)}{\theta^2 f_n'} \right], \quad (34)$$

55 where

$$56 \quad y_n = x_n - \theta u_n. \quad (35)$$

57 Two special cases were discussed in [23]. The choice $\theta = 1$ yields the method

$$58 \quad x_{n+1} = x_n - u_n \left[1 + \frac{f(x_n - u_n)}{f_n} \right]. \quad (36)$$

59 Another possibility is to choose θ so that

$$60 \quad \theta^2 + \theta - 1 = 0$$

61 i.e.

$$62 \quad \theta = \frac{-1 \pm \sqrt{5}}{2}. \quad (37)$$

63 In this case, the method is

$$64 \quad x_{n+1} = x_n - \frac{2}{3 - \sqrt{5}} \frac{f(x_n - \theta u_n)}{f_n'}. \quad (38)$$

65 We now construct a family similar to (34) for the case of multiple roots. Let

$$66 \quad x_{n+1} = x_n - u_n \left[\beta + \gamma \frac{f(y_n)}{f_n} \right], \quad (39)$$

67 where

$$68 \quad y_n = x_n - \alpha u_n. \quad (40)$$

69 Using Maple [22] to get all the Taylor series necessary, we have (up to third order terms)

$$70 \quad e_{n+1} = c_1 e_n + c_2 e_n^2 + c_3 e_n^3, \quad (41)$$

71 where

$$72 \quad c_1 = \frac{m - \beta - \gamma \rho}{m},$$

$$73 \quad c_2 = B_1 \frac{(m - \alpha)(\gamma \rho + \beta) - \gamma \rho \alpha^2}{m^2(m - \alpha)}, \quad (42)$$

and

$$\rho = \left(\frac{m-\alpha}{m}\right)^m. \quad (43)$$

Before we list c_3 , we choose β and γ to annihilate the coefficients c_1 and c_2

$$\begin{aligned} \beta &= m - \frac{m(m-\alpha)}{\alpha^2}, \\ \gamma &= \frac{m(m-\alpha)}{\rho\alpha^2}. \end{aligned} \quad (44)$$

In this case, we have

$$c_3 = -\frac{m-\alpha}{m^2}B_2 + \frac{1}{2} \frac{m(m+3) - 2\alpha(m+1)}{m^2(m-\alpha)}B_1^2. \quad (45)$$

We can choose the last parameter, α , to annihilate the coefficient of B_1^2 , i.e.

$$\alpha = \frac{1}{2} \frac{m(m+3)}{m+1}. \quad (46)$$

The choice $\alpha = m$ is not possible. Using the value of α from (46) in (44), we have the following:

$$\begin{aligned} \beta &= \frac{m^3 + 4m^2 + 9m + 2}{(m+3)^2}, \\ \gamma &= \frac{2^{m+1}(m^2-1)}{(m+3)^2 \left(\frac{m-1}{m+1}\right)^m}, \\ e_{n+1} &= -\frac{1}{2} \frac{m-1}{m(m+1)}B_2e_n^3. \end{aligned} \quad (47)$$

The one parameter family (39), (40), and (44) is third order requiring two function- and one derivative-evaluation per step.

The second new method is based on the approximation given by Neta [24] for the second derivative in (29), i.e.

$$f_n'' = \frac{6}{h^2}(f_{n-1} - f_n) + \frac{2}{h}f_{n-1}' + \frac{4}{h}f_n', \quad (48)$$

where $h = x_n - x_{n-1}$. This will give the following method:

$$x_{n+1} = x_n + \frac{m(m-3)}{2}u_n \left[1 - \frac{m}{m-3}u_n w(x_n)\right], \quad (49)$$

where

$$w(x_n) = \frac{6(f_{n-1} - f_n) + 2hf_{n-1}' + 4hf_n'}{h^2 f_n'}. \quad (50)$$

This modified method is of order 2.732 for $m \neq 3$ and it requires one function- and one derivative-evaluation per step. It also requires an additional starting value which we can be obtained using Newton's method (first derivative is required anyway).

The third new method is based on the approximation (48) for the second derivative in (11), i.e.

$$x_{n+1} = x_n - \frac{m(m+1)}{2}u_n + \frac{(m-1)^2}{2w(x_n)}, \quad (51)$$

where $w(x_n)$ is given by (50). This modified method is also of order 2.732 and it requires one function- and one derivative-evaluation per step. It also requires an additional starting value which we can be obtained using Newton's method (first derivative is required anyway).

Table 1
Comparison of methods for multiple roots

Algorithm	p	d	E	I	f'	f''	f'''
Werner [12] (16) $m = 2$	1.5	3	0.5	1.145	1		
Schröder [5] (1)	2	2	1	1.414	1		
Hansen and Patrick [6]	3	3	1	1.442	1	1	
Halley (2)	3	3	1	1.442	1	1	
Laguerre	3	3	1	1.442	1	1	
Hansen and Patrick [6]	3	4	0.75	1.316	1	1	1
Victory and Neta [7] (3)	3	3	1	1.442	1		
Dong [8] (7), (8)	3	3	1	1.442	1		
Dong [9] (9), (10)	3	3	1	1.442	2		
Osada [16]	3	3	1	1.442	1	1	
Neta and Johnson [10] (12) $m \neq 2$	4	4	1	1.414	3		
Neta and Johnson [10] (12) $m = 2$	4	3	1.333	1.587	2		
Neta [11]	4	4	1	1.414	1	3	
Neta (29) $m \neq 3$	3	3	1	1.442	1	1	
Neta (32) $m = 3$	2	3	0.667	1.259	1	1	
Neta (39)	3	3	1	1.442	1		
Neta (49)	2.732	2	1.366	1.653	1		
Neta (51)	2.732	2	1.366	1.653	1		

We now give the definitions of informational efficiency (see Traub [2])

$$E = p/d \quad (52)$$

and efficiency index

$$I = p^{1/d}. \quad (53)$$

where p is the order of the method and d is the number of function (and derivatives) evaluations per step. Clearly it is assumed that the cost of evaluating a function or any of the derivatives required is identical. In Table 1 we list all methods known (to the author) for finding roots with multiplicity m . It can be seen that the informational efficiency is almost always unity. The efficiency index for our methods here is the highest for a general m . Of the two third order methods developed here, we prefer the second which has the same efficiency independent of the multiplicity. These two schemes are still preferable over the fourth order methods given by Neta and Johnson [10] and Neta [11] because we have a closed form formula for the parameters. The last two methods are of order lower than three but of higher efficiency.

4. Numerical experiments

In this section we report on numerical experiments using Halley's method (2), Chebyshev's method (29), the modified Chebyshev method (39) (with $\alpha = m/2$) all for multiple roots, the modified method (49) and the modified Osada scheme (51). In our first example we took a quadratic polynomial having a double roots at $\xi = 1$

$$f(x) = x^2 - 2x + 1. \quad (54)$$

Here we started with $x_0 = 0$ and the convergence achieved in 1 iteration for the first 3 methods and 2 iterations for the last two schemes. In the second example we took a polynomial having two double roots at $\xi = \pm 1$

$$f(x) = x^4 - 2x^2 + 1. \quad (55)$$

Starting at $x_0 = 0.8$ Halley's and Chebyshev's methods converged in 4 iterations but the modified Chebyshev required 3 iterations. The last two modified schemes required 5 iterations. When we start at $x_0 = 0.6$ all the methods required the same number of iterations as before. The results are given in Table 2, where we used $a(-b)$ to denote $a \times 10^{-b}$. Note that the modified Chebyshev was consistently better than the first two schemes. The last two schemes are slower since their order is less than three.

Table 2
Comparison of 5 methods for 4 different initial guesses for Example 2

x_0	Method									
	(2)		(29)		(39)		(49)		(51)	
	n	$ f(x_n) $	n	$ f(x_n) $	n	$ f(x_n) $	n	$ f(x_n) $	n	$ f(x_n) $
0.8	4	0	4	0	3	0	5	0	5	0
0.6	4	6(-15)	4	5.8(-15)	3	2(-20)	5	0	5	0

Note that $a(-b)$ means $a \times 10^{-b}$.

Table 3
Results for Example 3

Method	n	$ f(x_n) $
(2)	3	1(-18)
(32)	3	1(-15)
(39)	3	0
(49)	7	1(-14)
(51)	9	0

Note that $a(-b)$ means $a \times 10^{-b}$.

The next example is a polynomial with triple root at $\xi = 1$. Recall that Chebyshev’s method for $m = 3$ has a free parameter α . In this example we used $\alpha = 3/2$

$$f(x) = x^5 - 8x^4 + 24x^3 - 34x^2 + 23x - 6. \tag{56}$$

The iteration starts with $x_0 = 0$ and the results are summarized in Table 3.

Another example with double root at $\xi = 0$ is

$$f(x) = x^2 e^x. \tag{57}$$

Starting at $x_0 = 0.1$ or at $x_0 = 0.2$ the first three methods converged in 2 iterations and the last two required 4–5 iterations. The results are given in Table 4.

Table 4
Results for Example 4

Method	$x_0 = 0.1$		$x_0 = 0.2$	
	n	$ f(x_n) $	n	$ f(x_n) $
(2)	2	4(-26)	2	8(-21)
(29)	2	2(-22)	2	3(-17)
(39)	2	2(-22)	2	3(-17)
(49)	4	3(-15)	5	5(-26)
(51)	4	1(-14)	5	1(-24)

Note that $a(-b)$ means $a \times 10^{-b}$.

Table 5
Results for Example 5

Method	$x_0 = 0$		$x_0 = 0.5$	
	n	$ f(x_n) $	n	$ f(x_n) $
(2)	3	1(-18)	3	0
(29)	5	0	3	0
(39)	4	1(-18)	3	0
(49)	5	0	5	1(-18)
(51)	5	2(-18)	5	1(-18)

Note that $a(-b)$ means $a \times 10^{-b}$.

The last example having a double root at $\xi = 1$ is

$$f(x) = 3x^4 + 8x^3 - 6x^2 - 24x + 19. \quad (58)$$

Now we started with $x_0 = 0$ and $x_0 = 0.5$ and the results are summarized in Table 5. Again the last two schemes are slower to converge.

5. Conclusions

We have developed four new methods to obtain multiple roots. The first scheme is third order and it requires a special care in the case that the root is triple. The second family does not require the use of the second derivative. Instead we have another function evaluation at an off-step. The other two methods of order 2.732 are more efficient. The numerical experiments show the rapid convergence of the third order methods with a slight edge to (39). The last two methods require slightly more iterations but less function evaluations per step.

References

- [1] A.M. Ostrowski, Solutions of Equations and System of equations, Academic Press, New York, 1960.
- [2] J.F. Traub, Iterative Methods for the Solution of Equations, Prentice Hall, New Jersey, 1964.
- [3] B. Neta, Numerical Methods for the Solution of Equations, Net-A-Sof, California, 1983.
- [4] L.B. Rall, Convergence of the Newton process to multiple solutions, Numer. Math. 9 (1966) 23–37.
- [5] E. Schröder, Über unendlich viele Algorithmen zur Auflösung der Gleichungen, Math. Ann. 2 (1870) 317–365.
- [6] E. Hansen, M. Patrick, A family of root finding methods, Numer. Math. 27 (1977) 257–269.
- [7] H.D. Victory, B. Neta, A higher order method for multiple zeros of nonlinear functions, Int. J. Comput. Math. 12 (1983) 329–335.
- [8] C. Dong, A basic theorem of constructing an iterative formula of the higher order for computing multiple roots of an equation, Math. Numer. Sinica 11 (1982) 445–450.
- [9] C. Dong, A family of multipoint iterative functions for finding multiple roots of equations, Int. J. Comput. Math. 21 (1987) 363–367.
- [10] B. Neta, A.N. Johnson, High order nonlinear solver for multiple roots, Comput. Math. Appl., doi:10.1016/j.camwa.2007.09.001.
- [11] B. Neta, Extension of Murakami's high order nonlinear solver to multiple roots, Int. J. Comput. Math., submitted for publication.
- [12] W. Werner, Iterationsverfahren höherer Ordnung zur Lösung nicht linearer Gleichungen, Z. Angew. Math. Mech. 61 (1981) T322–T324.
- [13] E. Halley, A new, exact and easy method of finding the roots of equations generally and that without any previous reduction, Phil. Trans. Roy. Soc. Lond. 18 (1694) 136–148.
- [14] R.F. King, A family of fourth order methods for nonlinear equations, SIAM J. Numer. Anal. 10 (1973) 876–879.
- [15] P. Jarratt, Some fourth order multipoint methods for solving equations, Math. Comp. 20 (1966) 434–437.
- [16] N. Osada, An optimal multiple root-finding method of order three, J. Comput. Appl. Math. 51 (1994) 131–133.
- [17] P. Jarratt, Multipoint iterative methods for solving certain equations, Comput. J. 8 (1966) 398–400.
- [18] T. Murakami, Some fifth order multipoint iterative formulae for solving equations, J. Inform. Process. 1 (1978) 138–139.
- [19] V.F. Candela, A. Marquina, Recurrence relations for rational cubic methods II: the Chebyshev method, Computing 45 (1990) 355–367.
- [20] D.J. Hofsommer, Note on the computation of the zeros of functions satisfying a second order differential equation, Math. Table Other Aids Comp. 12 (1958) 58–60.
- [21] D.B. Popovski, A family of one point iteration formulae for finding roots, Int. J. Comput. Math. 8 (1980) 85–88.
- [22] D. Redfern, The Maple Handbook, Springer-Verlag, New York, 1994.
- [23] J. Kou, Y. Li, Modified Chebyshev's method free from second derivative for nonlinear equations, Appl. Math. Comp. 187 (2007) 1027–1032.
- [24] B. Neta, On Popovski's method for nonlinear equations, J. Comput. Appl. Math., in press.