Available online at www.tjnsa.com
J. Nonlinear Sci. Appl. 9 (2016), 1035–1042
Research Article

Journal of Nonlinear Science and Applications
Print: ISSN 2008-1898 Online: ISSN 2008-1901

Iterative methods for solving scalar equations

Shin Min Kang[a], Faisal Ali[b], Arif Rafiq[c,*]

[a] *Department of Mathematics and the RINS, Gyeongsang National University, Jinju 52828, Korea.*
[b] *Centre for Advanced Studies in Pure and Applied Mathematics, Bahauddin Zakariya University, Multan 60800, Pakistan.*
[c] *Department of Mathematics and Statistics, Virtual University of Pakistan, Lahore 54000, Pakistan.*

Communicated by Y. J. Cho

Abstract

In this paper, we establish new iterative methods for the solution of scalar equations by using the decomposition technique mainly due to Daftardar-Gejji and Jafari [V. Daftardar-Gejji, H. Jafari, J. Math. Anal. Appl., **316** (2006), 753–763].

Keywords: Iterative methods, nonlinear equations, order of convergence, multiple roots.
2010 MSC: 65H05.

1. Introduction

Solving the nonlinear equations has been one of the scorching topics for scientists during the last few decades. Many iterative methods involving various techniques have been established to find the approximate roots of nonlinear equations, see [1, 3, 4, 5, 7, 10, 11, 13] and references there in. These methods can be classified as one-step, two-step and three-step methods. In [5], Chun has proposed and studied several one-step and two-step iterative methods with higher-order convergence by using the decomposition technique of Adomian [2]. Several other iterative methods have also been developed for finding the simple zero of nonlinear equations. In many physical problems we have to deal with the nonlinear equation having roots with multiplicity $m \geq 2$.

Firstly, Schröder [17] introduced the following modified Newton's method for finding the multiple roots:

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

During the last two decades, much attention has been devoted by various researchers for solving nonlinear equation with multiple roots. Chun et al. [6, 8], Homeier [12], Osada [16] have developed some techniques to find the multiple roots of nonlinear equations. In the recent years, the researchers have made significant

*Corresponding author
Email addresses: `smkang@gnu.ac.kr` (Shin Min Kang), `faisalali@bzu.edu.pk` (Faisal Ali), `aarafiq@gmail.com` (Arif Rafiq)

Received 2015-08-12

and interesting contribution in this field [9, 14, 15].

Having motivation from the recent research, we establish new iterative methods for finding multiple roots of scalar equations by using decomposition techniques due to Daftardar-Gejji and Jafari [10].

Consider the nonlinear equation

$$f(x) = 0. \tag{1.1}$$

It is well known that if α is a root with multiplicity m, then it is also a root of $f'(x) = 0$ with multiplicity $m-1$ of $f''(x) = 0$ with multiplicity $m-2$ and so on. Hence if initial guess x_0 is sufficiently close to α, the expressions

$$\begin{aligned} &x_0 - m\frac{f(x_0)}{f'(x_0)}, \\ &x_0 - (m-1)\frac{f'(x_0)}{f''(x_0)}, \\ &x_0 - (m-2)\frac{f''(x_0)}{f'''(x_0)}, \\ &\vdots \end{aligned} \tag{1.2}$$

will have the same value.

Remark 1.1. The generalized Newton's formula

$$x_{n+1} = x_n - 2\frac{f(x_n)}{f'(x_n)} \tag{1.3}$$

gives a quadratic convergence when the equation $f(x) = 0$ has a pair of double roots in the neighborhood of x_0. It may be noted that for the double root α near to x_0, $f(\alpha) = 0 = f'(\alpha)$.

2. New iterative methods

We can rewrite the nonlinear equation (1.1) as a coupled system:

$$f(\gamma) + (x-\gamma)\frac{f'(\gamma) + f'(x)}{2} = 0, \tag{2.1}$$

where γ is the initial approximation for a zero of (1.1).

Now, we can rewrite (2.1) in the following form:

$$\begin{aligned} x &= \gamma - 2\frac{f(\gamma)}{f'(\gamma)} - (x-\gamma)\frac{f'(x)}{f'(\gamma)} \\ &= c + N(x), \end{aligned} \tag{2.2}$$

where

$$c = \gamma - 2\frac{f(\gamma)}{f'(\gamma)} \tag{2.3}$$

and

$$N(x) = -(x-\gamma)\frac{f'(x)}{f'(\gamma)}. \tag{2.4}$$

Here $N(x)$ is a nonlinear operator.

As in [6], the solution of (2.2) has the series form

$$x = \sum_{i=0}^{\infty} x_i. \tag{2.5}$$

The nonlinear operator $N(x)$ can be decomposed as it has been shown in [7]. Also the series $\sum_{i=0}^{\infty} x_i$ converges absolutely and uniformly to a unique solution of equation (2.2) if the nonlinear operator

$$N(x) = N\left(\sum_{i=0}^{\infty} x_i\right) = N(x_0) + \sum_{i=1}^{\infty}\left\{N\left(\sum_{j=0}^{i} x_j\right) - N\left(\sum_{j=0}^{i-1} x_j\right)\right\} \tag{2.6}$$

is a contraction. Combining (2.2), (2.4) and (2.6), we have

$$\sum_{i=0}^{\infty} x_i = c + N(x_0) + \sum_{i=1}^{\infty}\left\{N\left(\sum_{j=0}^{i} x_j\right) - N\left(\sum_{j=0}^{i-1} x_j\right)\right\}. \tag{2.7}$$

Thus we have the following iterative scheme:

$$\begin{aligned} x_0 &= c, \\ x_1 &= N(x_0), \\ x_2 &= N(x_0 + x_1) - N(x_0), \\ &\vdots \\ x_{n+1} &= N(x_0 + x_1 + \ldots + x_n) - N(x_0 + x_1 + \ldots + x_{n-1}), \quad n = 1, 2, \ldots. \end{aligned}$$

Then

$$x_1 + x_2 + \ldots + x_{n+1} = N(x_0 + x_1 + \ldots + x_n), \quad n = 1, 2, \ldots$$

and

$$x = c + \sum_{i=1}^{\infty} x_i. \tag{2.8}$$

From (2.3), (2.4) and (2.8), we have

$$x_0 = c = \gamma - 2\frac{f(\gamma)}{f'(\gamma)} \tag{2.9}$$

and

$$\begin{aligned} x_1 &= N(x_0) \\ &= -(x_0 - \gamma)\frac{f'(x_0)}{f'(\gamma)} \quad = 2\frac{f(\gamma)f'(x_0)}{f'^2(\gamma)}. \end{aligned} \tag{2.10}$$

It follows from (2.3), (2.8) and (2.8) that

$$\begin{aligned} x &\approx x_0 \\ &= c \\ &= \gamma - 2\frac{f(\gamma)}{f'(\gamma)}. \end{aligned} \tag{2.11}$$

This enables us to suggest the following method for solving the nonlinear equation (1.1).

Algorithm 2.1. For the given x_0 compute the approximate solution x_{n+1} by the iterative schemes:

$$x_{n+1} = x_n - 2\frac{f(x_n)}{f'(x_n)}, \quad f'(x_n) \neq 0, \quad n = 0, 1, 2, \ldots,$$

which is known as the generalized Newton's formula and is quadratically convergent.

Again by using (2.3), (2.4), (2.8) and (2.8), we conclude that

$$\begin{aligned} x &\approx x_0 + x_1 \\ &= c + x_1 \\ &= x_0 + N(x_0) \\ &= \gamma - 2\frac{f(\gamma)}{f'(\gamma)} + 2\frac{f(\gamma)f'(x_0)}{f'^2(\gamma)}. \end{aligned} \tag{2.12}$$

Using (2.12), we can suggest the following two-step iterative method for solving nonlinear equation (1.1) as follows:

Algorithm 2.2. For the given x_0 compute the approximate solution x_{n+1} by the iterative schemes:

$$\begin{aligned} y_n &= x_n - 2\frac{f(x_n)}{f'(x_n)}, \quad f'(x_n) \neq 0, \quad n = 0, 1, 2, \dots, \\ x_{n+1} &= y_n + 2\frac{f(x_n)f'(y_n)}{f'^2(x_n)}. \end{aligned}$$

Again, using (2.4), (2.9) and (2.10), we can calculate

$$\begin{aligned} N(x_0 + x_1) &= -(x_0 + x_1 - \gamma)\frac{f'(x_0 + x_1)}{f'(\gamma)} \\ &= 2\frac{f(\gamma)}{f'(\gamma)}\left(1 - \frac{f'(x_0)}{f'(\gamma)}\right)\frac{f'(x_0 + x_1)}{f'(\gamma)}. \end{aligned} \tag{2.13}$$

From (2.8), (2.9), (2.11) and (2.13), we get

$$\begin{aligned} x &\approx x_0 + x_1 + x_2 \\ &= c + N(x_0) + N(x_0 + x_1) - N(x_0) \\ &= c + N(x_0 + x_1) \\ &= \gamma - 2\frac{f(\gamma)}{f'(\gamma)} + 2\frac{f(\gamma)}{f'(\gamma)}\left(1 - \frac{f'(x_0)}{f'(\gamma)}\right)\frac{f'(x_0 + x_1)}{f'(\gamma)}. \end{aligned} \tag{2.14}$$

Using (2.14), we can suggest and analyze the following three-step iterative method for solving nonlinear equation (1.1).

Algorithm 2.3. For a given $x_{0,}$ compute the approximate solution by the iterative schemes:

$$\begin{aligned} y_n &= x_n - 2\frac{f(x_n)}{f'(x_n)}, \quad f'(x_n) \neq 0, \quad n = 0, 1, 2, \dots, \\ z_n &= 2\frac{f(x_n)f'(y_n)}{f'^2(x_n)}, \\ x_{n+1} &= y_n + 2\frac{f(x_n)}{f'(x_n)}\left(1 - \frac{f'(y_n)}{f'(x_n)}\right)\frac{f'(y_n + z_n)}{f'(x_n)}. \end{aligned}$$

3. Convergence analysis

In this section, the convergence analysis of Algorithms 2.2 and 2.3 is given.

Theorem 3.1. *Assume that the function $f : D \subset \mathbb{R} \to \mathbb{R}$ for an open interval D has a multiple root $\alpha \in D$ of multiplicity 2 . Let $f(x)$ be sufficiently smooth in the neighborhood of the root α. Then the order of convergence of the methods defined by* Algorithms 2.2 *and* 2.3 *is 2.*

Proof. Let α be a root of $f(x)$ of multiplicity 2. Then by expanding $f(x_n)$, $(f'(x_n))^2$ in Taylor's series about α, we obtain

$$f(x_n) = c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + O(e_n^5), \tag{3.1}$$

$$f'(x_n) = 2c_2 e_n + 3c_3 e_n^2 + 4c_4 e_4^3 + 5c_5 e_n^4 + O(e_n^5), \tag{3.2}$$

$$\left(f'(x_n)\right)^2 = 4c_2^2 e_n^2 + 12c_2 c_3 e_n^3 + \left(16c_2c_4 + 9c_3^2\right)e_4^4 + O(e_n^5), \tag{3.3}$$

where $e_n = x_n - \alpha$ and $c_k = \frac{f^{(k)}(\alpha)}{k!}$, $k = 2, 3, \ldots$.

Using (3.1) and (3.2), we have

$$\frac{f(x_n)}{f'(x_n)} = \frac{1}{2}e_n - \frac{1}{4}\frac{c_3}{c_2}e_n^2 + \left(-\frac{1}{2}\frac{c_4}{c_2} + \frac{3}{8}\frac{c_3^2}{c_2^2}\right)e_n^3 + \left(-\frac{3}{4}\frac{c_5}{c_2} + \frac{5}{4}\frac{c_3c_4}{c_2^2} - \frac{9}{16}\frac{c_3^3}{c_2^3}\right)e_n^4 + O(e_n^5). \tag{3.4}$$

Thus

$$y_n = x_n - 2\frac{f(x_n)}{f'(x_n)} = \alpha + \frac{1}{2}\frac{c_3}{c_2}e_n^2 + \left(\frac{c_4}{c_2} - \frac{3}{4}\frac{c_3^2}{c_2^2}\right)e_n^3 + \left(\frac{3}{2}\frac{c_5}{c_2} - \frac{5}{2}\frac{c_3c_4}{c_2^2} + \frac{9}{8}\frac{c_3^3}{c_2^3}\right)e_n^4 + O(e_n^5). \tag{3.5}$$

Expanding $f'(y_n)$ by Taylor's series about α, we get

$$f'(y_n) = \frac{c_3^2}{c_2}e_n^3 + \left(\frac{5c_3c_4}{c_2} - \frac{15}{4}\frac{c_3^3}{c_2^2}\right)e_n^4 + \left(\frac{9c_3c_5}{c_2} - \frac{24c_4c_3^2}{c_2^2} + \frac{87}{8}\frac{c_3^4}{c_2^3} + \frac{6c_4^2}{c_2}\right)e_n^5 + O(e_n^6). \tag{3.6}$$

Using (3.3), (3.5) and (3.6), the error term becomes

$$e_{n+1} = \frac{1}{2}\frac{c_3}{c_2}e_n^2 + \left(\frac{c_4}{c_2} - \frac{1}{4}\frac{c_3^2}{c_2^2}\right)e_n^3 + \left(\frac{3}{2}\frac{c_5}{c_2} - \frac{7}{2}\frac{c_3^3}{c_2^3}\right)e_n^4 + O(e_n^5). \tag{3.7}$$

Next for Algorithm 2.3, we use (3.1), (3.2) and (3.6) to compute z_n as follows

$$z_n = 2\frac{f(x_n)f'(y_n)}{f'^2(x_n)} = \frac{1}{2}\frac{c_3^2}{c_2^2}e_n^3 + \left(\frac{5}{2}\frac{c_3c_4}{c_2^2} - \frac{23}{8}\frac{c_3^3}{c_2^3}\right)e_n^4 + O(e_n^5). \tag{3.8}$$

Using (3.5) and (3.8), we have

$$y_n + z_n = \alpha + \frac{1}{2}\frac{c_3}{c_2}e_n^2 + \left(\frac{c_4}{c_2} - \frac{1}{4}\frac{c_3^2}{c_2^2}\right)e_n^3 + \left(\frac{3}{2}\frac{c_5}{c_2} - \frac{7}{4}\frac{c_3^3}{c_2^3}\right)e_n^4 + O(e_n^5). \tag{3.9}$$

Expanding $f'(y_n + z_n)$ by Taylor's series about α, we obtain

$$f'(y_n + z_n) = \frac{c_3^2}{c_2}e_n^3 + \left(\frac{5c_3c_4}{c_2} - \frac{5}{4}\frac{c_3^3}{c_2^2}\right)e_n^4 + \left(\frac{9c_3c_5}{c_2} - \frac{75c_3^4}{8c_2^3} + \frac{6c_4^2}{c_2} - \frac{3c_4c_3^2}{c_2c_2^2}\right)e_n^5 + O(e_n^6). \tag{3.10}$$

Thus

$$1 - \frac{f'(y_n)}{f'(x_n)} = 1 - \frac{1}{2}\frac{c_3^2}{c_2^2}e_n^2 + \left(-\frac{5}{2}\frac{c_3c_4}{c_2^2} + \frac{21}{8}\frac{c_3^3}{c_2^3}\right)e_n^3 + \left(-\frac{9}{2}\frac{c_3c_5}{c_2^2} + \frac{67c_4c_3^2}{4c_2^3} - \frac{75c_3^4}{8c_2^4} - \frac{3c_4^2}{c_2^2}\right)e_n^4 + O(e_n^5) \tag{3.11}$$

and

$$\frac{f'(y_n + z_n)}{f'(x_n)} = \frac{1}{2}\frac{c_3^2}{c_2^2}e_n^2 + \left(\frac{5}{2}\frac{c_3c_4}{c_2^2} - \frac{11}{8}\frac{c_3^3}{c_2^3}\right)e_n^3 + \left(\frac{9}{2}\frac{c_3c_5}{c_2^2} - \frac{21c_3^4}{8c_2^4} + \frac{3c_4^2}{c_2^2} - \frac{25c_4c_3^2}{4c_2^3}\right)e_n^4 + O(e_n^5). \tag{3.12}$$

Now using (3.4), (3.5), (3.11) and (3.12), we have the error term as

$$e_{n+1} = \frac{1}{2}\frac{c_3}{c_2}e_n^2 + \left(\frac{c_4}{c_2} - \frac{1}{4}\frac{c_3^2}{c_2^2}\right)e_n^3 + \left(-\frac{1}{2}\frac{c_3}{c_2^3} + \frac{3}{2}\frac{c_5}{c_2}\right)e_n^4 + O(e_n^5). \tag{3.13}$$

The last equation shows that the convergence order of Algorithms 2.3 is 2. This completes the proof. □

4. Numerical examples

In this section we consider some numerical examples to demonstrate the performance of the newly developed iterative method. In the following table, we compare our proposed methods (Algorithms 2.2 and 2.3) (NS1 and NS2) with classical Newton's method (NM), generalized Newton's method (Algorithm 2.1) (GNM), Chun et al. [6, Equations (35) and (36)] (BNM1 and BNM2). All the computations for above mentioned methods, are performed using software Maple 13 and $\varepsilon = 10^{-10}$ as tolerance and also the following criteria is used for estimating the zero:

(i) $\delta = |x_{n+1} - x_n| < \varepsilon$,
(ii) $|f(x_n)| < \varepsilon$,
(iii) Maximum numbers of iterations = 500.

$f(x)$	$x_\circ$	Methods	No. of iterations	$x[k]$	$f(x_n)$	δ
$(x-2)^2$	1.97	NM	10	1.9999707031250000	$8.58e-10$	$2.93e-05$
		GNM	1	2.0000000000000000	$0.00e+00$	$3.00e-02$
		BNM1	1	2.0000000000000000	$0.00e+00$	$3.00e-02$
		BNM2	1	2.0000000000000000	$0.00e+00$	$3.00e-02$
		NS1	1	2.0000000000000000	$0.00e+00$	$3.00e-02$
		NS2	1	2.0000000000000000	$0.00e+00$	$3.00e-02$
$\arctan^2 x$	0.02	NM	10	0.0000195243074874	$3.81e-10$	$1.95e-05$
		GNM	1	−0.0000053329067398	$2.84e-11$	$2.00e-02$
		BNM1	1	0.0000026713622895	$7.14e-12$	$2.00e-02$
		BNM2	1	0.0000026617546045	$7.08e-12$	$2.00e-02$
		NS1	1	−0.0000106700806580	$1.14e-10$	$2.00e-02$
		NS2	1	−0.0000160143740905	$2.56e-10$	$2.00e-02$
$(e^x-4x^2)^2$	−0.39	NM	12	−0.4077725133649210	$2.72e-10$	$4.20e-06$
		GNM	2	−0.4077767961647673	$1.16e-13$	$3.05e-04$
		BNM1	2	−0.4077767094044696	$1.78e-27$	$1.69e-05$
		BNM2	1	−0.4077777243328640	$1.59e-11$	$1.78e-02$
		NS1	2	−0.4077774521231989	$8.51e-12$	$6.31e-04$
		NS2	2	−0.4077794589496283	$1.17e-10$	$9.91e-04$
x^3-x^2-x+1	1.03	NM	11	1.0000148669755619	$4.42e-10$	$1.49e-05$
		GNM	2	1.0000000121033821	$2.93e-16$	$2.20e-04$
		BNM1	1	1.0000055621052384	$6.19e-11$	$3.00e-02$
		BNM2	1	1.0000009278935850	$1.72e-12$	$3.00e-02$
		NS1	2	1.0000000940022467	$1.77e-14$	$4.34e-04$
		NS2	2	1.0000003051214826	$1.86e-13$	$6.38e-04$
$(\sin x-\cos x)^2$	0.7	NM	12	0.7853773818407633	$8.64e-10$	$2.08e-05$
		GNM	2	0.7853981633944398	$1.81e-23$	$2.08e-04$
		BNM1	2	0.7853981633972462	$8.17e-26$	$1.07e-04$
		BNM2	2	0.7853981633972653	$6.70e-26$	$1.03e-04$
		NS1	2	0.7853981633487803	$4.74e-21$	$4.12e-04$
		NS2	2	0.7853981631471178	$1.25e-19$	$6.30e-04$
$(\sin^2 x-x^2+1)^2$	1.45	NM	12	1.4045035561877959	$8.74e-10$	$1.19e-05$
		GNM	2	1.4044934737850935	$2.05e-11$	$1.53e-03$
		BNM1	2	1.4044916482200438	$1.36e-22$	$1.39e-04$
		BNM2	1	1.4044991170005235	$3.44e-10$	$4.55e-02$
		NS1	2	1.4045043854006513	$10.0e-10$	$2.85e-03$
		NS2	3	1.4044916512879718	$5.82e-17$	$3.62e-05$
x^3-3x+2	1.06	NM	12	1.0000149394272540	$6.70e-10$	$1.49e-05$
		GNM	2	1.0000000565392868	$9.59e-15$	$5.82e-04$
		BNM1	2	1.0000000000000007	$1.51e-30$	$1.94e-05$
		BNM2	1	1.0000033833477819	$3.43e-11$	$6.00e-02$
		NS1	2	1.0000004348814882	$5.67e-13$	$1.14e-03$
		NS2	2	1.0000013939540372	$5.83e-12$	$1.67e-03$

$f(x)$	$x_\circ$	Methods	No. of iterations	$x[k]$	$f(x_n)$	δ
$x^4-11x^3+36x^2$ $-16x-64$	4.02	NM	9	4.0005212575599936	$7.08e-10$	$2.61e-04$
		GNM	4	4.0002478869802387	$7.62e-11$	$4.96e-04$
		BNM1	3	4.0002640566429309	$9.21e-11$	$8.54e-04$
		BNM2	3	4.0001472421448571	$1.60e-11$	$6.10e-04$
		NS1	4	4.0005530334574302	$8.46e-10$	$8.04e-04$
		NS2	5	4.0003008933355271	$1.36e-10$	$3.96e-04$
$(x-\tan x)^2$	0.05	NM	1	0.0416722242070108	$5.83e-10$	$8.33e-03$
		GNM	1	0.0333444484140216	$1.53e-10$	$1.67e-02$
		BNM1	1	0.0288991908845572	$6.47e-11$	$2.11e-02$
		BNM2	1	0.0255627464167388	$3.10e-11$	$2.44e-02$
		NS1	1	0.0355381781052461	$2.24e-10$	$1.44e-02$
		NS2	1	0.0359642913156559	$2.41e-10$	$1.40e-02$
$(x-\sin x)^2$	0.06	NM	1	0.0499987998456958	$4.34e-10$	$1.00e-02$
		GNM	1	0.0399975996913915	$1.14e-10$	$2.00e-02$
		BNM1	1	0.0346644425641644	$4.82e-11$	$2.53e-02$
		BNM2	1	0.0306651148409437	$2.31e-11$	$2.93e-02$
		NS1	1	0.0426315727995842	$1.67e-10$	$1.74e-02$
		NS2	1	0.0431436396944703	$1.79e-10$	$1.69e-02$
$(x^2+\sin\frac{x}{5}-\frac{1}{4})^2$	0.37	NM	11	0.4099740214194201	$3.36e-10$	$1.80e-05$
		GNM	2	0.4099948386255551	$8.27e-12$	$1.70e-03$
		BNM1	2	0.4099920179505115	$1.55e-21$	$2.37e-04$
		BNM2	2	0.4099920179891372	$2.43e-33$	$3.20e-05$
		NS1	2	0.4100186071366364	$7.35e-10$	$3.68e-03$
		NS2	3	0.4099920555014645	$1.46e-15$	$1.13e-04$
$(x^2+7x-30)^2$	3.03	NM	14	3.0000018395052141	$5.72e-10$	$1.84e-06$
		GNM	2	3.0000000003653009	$2.26e-17$	$6.89e-05$
		BNM1	1	3.0000004727268600	$3.78e-11$	$3.00e-02$
		BNM2	1	3.0000000032477361	$1.78e-15$	$3.00e-02$
		NS1	2	3.0000000028956322	$1.42e-15$	$1.37e-04$
		NS2	2	3.0000000096539636	$1.58e-14$	$2.04e-04$

5. Conclusions

In the present work, we have proposed two new iterative methods (NS1 and NS2) with convergence order 2 for finding the multiple roots of nonlinear equations. The numerical results presented in the Table given in the previous section reveal that our iterative methods are even comparable with the methods developed by Chun et al. [6] (BNM1 and BNM2) with convergence order 3. The idea and technique employed in this paper can be developed to higher-order multi-step iterative methods for solving nonlinear equations having roots with multiplicity greater than one.

References

[1] S. Abbasbandy, *Improving Newton-Raphson method for nonlinear equations modified Adomian decomposition method*, Appl. Math. Comput., **145** (2003), 887–893. 1

[2] G. Adomian, *Nonlinear Stochastic Systems and Applications to Physics*, Kluwer Academic Publishers, Dordrecht (1989). 1

[3] E. Babolian, J. Biazar, *On the order of convergence of Adomian method*, Appl. Math. Comput., **130** (2002), 383–387. 1

[4] E. Babolian, J. Biazar, *Solution of nonlinear equations by modified Adomian decomposition method*, Appl. Math. Comput., **132** (2002), 167–172. 1

[5] C. Chun, *Iterative methods improving Newton's method by the decomposition method*, Comput. Math. Appl., **50** (2005), 1559–1568. 1

[6] C. Chun, H. J. Bae, B. Neta, *New families of nonlinear third-order solvers for finding multiple roots*, Comput. Math. Appl., **57** (2009), 1574–1582. 1, 2, 4, 5

[7] C. Chun, Y. Ham, *A one-parameter fourth-order family of iterative methods for nonlinear equations*, Appl. Math. Comput., **189** (2007), 610–614. 1, 2

[8] C. Chun, B. Neta, *A third-order modification of Newton's method for multiple roots*, Appl. Math. Comput., **211** (2009), 474–479. 1

[9] C. Chun, B. Neta, *Basin of attraction for Zhou-Chen-Song fourth order family of methods for multiple roots*, Math. Comput. Simulation, **109** (2015), 74–91. 1

[10] V. Daftardar-Gejji, H. Jafari, *An iterative method for solving nonlinear functional equations*, J. Math. Anal. Appl., **316** (2006), 753–763. 1

[11] V. I. Hasanov, I. G. Ivanov, G. Nedzhibov, *A new modification of Newton's method*, Appl. Math. Engin. Econom. (Sozopol, 2001), pp. 278–286, Heron Press, Sofia, (2002). 1

[12] H. H. H. Homeier, *On Newton-type methods for multiple roots with cubic convergence*, J. Comput. Appl. Math., **231** (2009), 249–254. 1

[13] E. Isaacson, H. B. Keller, *Analysis of Numerical Methods*, John Wiley & Sons, Inc., New York, (1966). 1

[14] B. Neta, C. Chun, *On a family of Laguerre methods to find multiple roots of nonlinear equations*, Appl. Math. Comput., **219** (2013), 10987–11004. 1

[15] B. Neta, C. Chun, M. Scott, *On a development of iterative methods for multiple roots*, Appl. Math. Comput., **224** (2013), 358–361. 1

[16] N. Osada, *An optimal multiple root-finding method of order three*, J. Comput. Appl. Math., **51** (1994), 131–133. 1

[17] E. Schröder, *Über unendlich viele algorithmen zur auflösung der gleichungen*, Math. Ann., **2** (1870), 317–365. 1