



A Note on Three-Step Iterative Method with Seventh Order of Convergence for Solving Nonlinear Equations

Napassanan Srisarakham and Montri Thongmoon¹

Mathematics and Applied Mathematics Research Unit, Department
of Mathematics, Faculty of Science, Mahasarakham University,
Mahasarakham 44150, Thailand.

e-mail : ekjitrtras@yahoo.com (N. Srisarakham)
montri.t@msu.ac.th (M. Thongmoon)

Abstract : In this paper, we present a new higher order iterative method for solving nonlinear equations. This method based on a Halley's iterative method and using predictor-corrector technique. The convergence analysis of this method is discussed. It is established that the new method has convergence order seven. Numerical tests show that the new method is comparable with the well-known existing methods and gives better results.

Keywords : nonlinear equations; iterative method; convergence analysis; Halley's method; Householder method.

2010 Mathematics Subject Classification : 41A25; 65H05; 65K05.

1 Introduction

Iterative methods for solving nonlinear equations are an important area of research in numerical analysis. They can be used interesting applications in several branches of pure and applied science can be studied in the general framework of the nonlinear equations (see [1]). Due to their importance, several numerical methods have been suggested and analysed under certain conditions. These numerical methods have been constructed using different techniques such as Taylor series,

¹Corresponding author.

homotopy perturbation method and its variant forms, quadrature formula (see [2, 3]), variational iteration method, and decomposition method. For more details see [2]-[21]. In this paper, based on a Halley and Householder iterative methods and using predictor-corrector technique, we construct modification of Newton's method with higher-order convergence for solving nonlinear equations. The error equations are given theoretically to show that the proposed technique has seventh order convergence. Commonly in the literature the efficiency of an iterative method is measured by the efficiency index defined as $I \approx p^{\frac{1}{d}}$ (see [13]), where p is the order of convergence and d is the total number of functional evaluations per step. Therefore this method has efficiency index $7^{\frac{1}{5}} \approx 1.475$ which are higher than $2^{\frac{1}{2}} \approx 1.4142$ of the Steffensen's method (SM) (see [14]), $3^{\frac{1}{4}} \approx 1.3161$ of the (DHM) method (see [15]). Several examples are given to illustrate the efficiency and performance of this method.

2 Iterative Methods

Consider the nonlinear equation of the type $f(x) = 0$. For simplicity, assume that r is a simple root zero and γ is an initial guess sufficiently closed to r . Using the Taylor's series expansion of the function $f(x)$, we have

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

from which we have

$$x = \gamma - \frac{f(\gamma)}{f'(\gamma)} - \frac{(x - \gamma)^2}{2}f''(\gamma). \quad (2.2)$$

This formulation allows us to suggest the following iterative methods for solving the nonlinear equations

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ w_n &= y_n - \frac{f(y_n)}{f'(y_n)} - \frac{(f(y_n))^2 f''(y_n)}{2(f'(y_n))^3} \\ x_{n+1} &= w_n - \frac{(x_n - w_n)f(w_n)}{f(x_n) - 2f(w_n)}. \end{aligned} \quad (2.3)$$

To reduce the number of functions and improve the efficiency index, we replace the second derivative $f''(y_n)$ by

$$f''(y_n) = \frac{2}{y_n - x_n} \left(2f'(y_n) + f'(x_n) - 3 \frac{f(y_n) - f(x_n)}{y_n - x_n} \right) \quad (2.4)$$

which is approximated by Noor et al. [13], hence we get the new algorithm as follows:

Algorithm(*):

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ w_n &= y_n - \frac{f(y_n)}{f'(y_n)} - \frac{(f(y_n))^2 P_f(x_n, y_n)}{2(f'(y_n))^3} \\ x_{n+1} &= w_n - \frac{(x_n - w_n)f(w_n)}{f(x_n) - 2f(w_n)} \end{aligned} \quad (2.5)$$

where

$$P_f(x_n, y_n) = \frac{2}{y_n - x_n} \left(2f'(y_n) + f'(x_n) - 3 \frac{f(y_n) - f(x_n)}{y_n - x_n} \right) \approx f''(y_n). \quad (2.6)$$

3 Convergence Analysis

Let us consider the convergence analysis of Algorithm(*).

Theorem 3.1. *Let r be a simple zero of sufficient differentiable function $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ for an open interval I . If x_0 is sufficient closed to r , then the three-step iterative method defined by Algorithm(*) has seven order convergence.*

Proof. Let r be a simple zero of sufficient differentiable function $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$. Since f is sufficient differentiable function, by Taylor's expanding of f about r , we get

$$\begin{aligned} f(x) &= f(r) + (x - r)f'(r) + \frac{(x - r)^2}{2!}f''(r) + \frac{(x - r)^3}{3!}f'''(r) \\ &+ \frac{(x - r)^4}{4!}f^{(4)}(r) + \dots \end{aligned} \quad (3.1)$$

Substituting $x = x_n$, we get

$$\begin{aligned} f(x_n) &= f(r) + (x_n - r)f'(r) + \frac{(x_n - r)^2}{2!}f''(r) + \frac{(x_n - r)^3}{3!}f'''(r) \\ &+ \frac{(x_n - r)^4}{4!}f^{(4)}(r) + \dots \end{aligned} \quad (3.2)$$

Calculating first derivative of f with respect to x , and substituting $x = x_n$ we have

$$f'(x_n) = f'(r) + (x_n - r)f''(r) + \frac{3(x_n - r)^2}{3!}f'''(r) + \frac{4(x_n - r)^3}{4!}f^{(4)}(r) + \dots \quad (3.3)$$

and

$$f''(x_n) = f''(r) + \frac{6(x_n - r)}{3!}f'''(r) + \frac{12(x_n - r)^2}{4!}f^{(4)}(r) + \dots \quad (3.4)$$

So

$$f(x_n) = \sum_{k=0}^{\infty} \frac{(x_n - r)^k}{k!} f^{(k)}(r), \quad (3.5)$$

$$f'(x_n) = \sum_{k=1}^{\infty} \frac{k(x_n - r)^{k-1}}{k!} f^{(k)}(r), \quad (3.6)$$

and

$$f''(x_n) = \sum_{k=2}^{\infty} \frac{k(k-1)(x_n - r)^{k-2}}{k!} f^{(k)}(r). \quad (3.7)$$

Since r is a simple zero of function f , then $f(r) = 0$. We get that

$$f(x_n) = f'(r) (e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 + c_6 e_n^6 + c_7 e_n^7 + c_8 e_n^8 + c_9 e_n^9 + c_{10} e_n^{10} + \dots). \quad (3.8)$$

We get that

$$f'(x_n) = f'(r) (1 + 2c_2 e_n + 3c_3 e_n^2 + 4c_4 e_n^3 + 5c_5 e_n^4 + 6c_6 e_n^5 + 7c_7 e_n^6 + 8c_8 e_n^7 + 9c_9 e_n^8 + 10c_{10} e_n^9 + \dots) \quad (3.9)$$

and

$$f''(x_n) = f'(r) (2c_2 + 6c_3 e_n + 12c_4 e_n^2 + 20c_5 e_n^3 + 30c_6 e_n^4 + 42c_7 e_n^5 + 56c_8 e_n^6 + 72c_9 e_n^7 + 90c_{10} e_n^8 + \dots) \quad (3.10)$$

where $c_k = \frac{1}{k!} \frac{f^{(k)}(r)}{f'(r)}$; $k = 2, 3, 4, \dots$ and $e_n = x_n - r$. Since $y_n = x_n - \frac{f(x_n)}{f'(x_n)}$, and substituting $f(x_n)$ and $f'(x_n)$ we get that

$$y_n = x_n - \frac{e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 + c_6 e_n^6 + c_7 e_n^7 + c_8 e_n^8 + c_9 e_n^9 + c_{10} e_n^{10} + \dots}{1 + 2c_2 e_n + 3c_3 e_n^2 + 4c_4 e_n^3 + 5c_5 e_n^4 + 6c_6 e_n^5 + 7c_7 e_n^6 + 8c_8 e_n^7 + 9c_9 e_n^8 + 10c_{10} e_n^9 + \dots}. \quad (3.11)$$

So

$$y_n = r + c_2 e_n^2 - 2(c_2^2 - c_3) e_n^3 - (7c_2 c_3 - 4c_2^3 - 3c_4) e_n^4 + (6c_3^2 - 20c_3 c_2^2 - 10c_2 c_4 - 4c_5 - 8c_2^4) e_n^5 + \dots. \quad (3.12)$$

From (3.1), substituting $x = y_n$ we get

$$f(y_n) = f(r) + (y_n - r) f'(r) + \frac{(y_n - r)^2}{2!} f''(r) + \frac{(y_n - r)^3}{3!} f'''(r) + \frac{(y_n - r)^4}{4!} f^{(4)}(r) + \dots \quad (3.13)$$

Substituting (3.12) into (3.13) and rearranging, we have that

$$f(y_n) = f'(r) (c_2 e_n^2 - 2(c_2^2 - c_3) e_n^3 - (7c_2 c_3 - 5c_2^3 - 3c_4) e_n^4 - 2(6c_2^4 - 12c_3^2 c_3 + 3c_3^2 + 5c_2 c_4 - 2c_5) e_n^5 - (28c_2^5 - 73c_2^3 c_3 + 37c_2 c_3^2 + 34c_2^4 c_4 - 17c_3 c_4 - 13c_2 c_5 + 5c_6) e_n^5 + \dots). \quad (3.14)$$

By (3.3), we get

$$f'(y_n) = f'(r) (1 + 2c_2(y_n - r) + 3c_3(y_n - r)^2 + 4c_4(y_n - r)^3 + 5c_5(y_n - r)^4 + 6c_6(y_n - r)^5 + \dots). \quad (3.15)$$

Substituting (3.12) into (3.15), we have that

$$f'(y_n) = f'(r) (1 + 2c_2 e_n^2 + 4(c_2 c_3 - c_2^3) e_n^3 + (6c_2 c_4 + 8c_2^4 - 11c_2^2 c_3) e_n^4 + \dots). \quad (3.16)$$

From (3.14) and (3.16), we have that

$$\begin{aligned} \frac{f(y_n)}{f'(y_n)} &= c_2 e_n^2 - 2(c_2^2 - c_3) e_n^3 - (7c_2 c_3 - 5c_2^3 - 3c_4) e_n^4 \\ &\quad + 2(8c_2^3 c_3 - 2c_2^4 - 3c_3^2 - 5c_2 c_4 + 2c_5) e_n^5 - (13c_2 c_5 - 22c_2^2 c_4 \\ &\quad - 5c_5 - 6c_2^5 - 32c_2^2 c_3 + 17c_3 c_4 - 29c_2 c_3^2) e_n^6 + \dots \end{aligned} \tag{3.17}$$

Since $P_f(x_n, y_n) = \frac{2}{y_n - x_n} \left(2f'(y_n) + f'(x_n) - 3\frac{f(y_n) - f(x_n)}{y_n - x_n} \right)$, then

$$\begin{aligned} P_f(x_n, y_n) &= f'(r) \left(2c_2 + (6c_2 c_3 - 2c_4) e_n^2 - 4(3c_3(c_2^2 - c_3) - c_2 c_4 + c_5) e_n^3 \right. \\ &\quad \left. + 2(12c_2^3 c_3 - 21c_2 c_3^2 + c_2^2 c_4 + 13c_3 c_4 + c_2 c_5 - 3c_5) e_n^4 + \dots \right) \end{aligned} \tag{3.18}$$

Let $w_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{(f(y_n))^2 P_f(x_n, y_n)}{2(f'(y_n))^3}$, then

$$w_n = r + c_2^2(2c_2^3 - c_2 c_3 + c_4) e_n^6 + \dots \tag{3.19}$$

and we get

$$x_n - w_n = e_n - c_2^2(2c_2^3 - c_2 c_3 + c_4) e_n^6 + \dots \tag{3.20}$$

Substituting $x = w_n$ into (3.1), we get

$$f(w_n) = f'(r) \left[(w_n - r) + \frac{(w_n - r)^2}{2!} \frac{f''(r)}{f'(r)} + \frac{(w_n - r)^3}{3!} \frac{f'''(r)}{f'(r)} + \frac{(w_n - r)^4}{4!} \frac{f^{(4)}(r)}{f'(r)} + \dots \right] \tag{3.21}$$

Substituting (3.19) into (3.21), we get

$$\begin{aligned} f(w_n) &= f'(r) \left[c_2^2(2c_2^3 - c_2 c_3 + c_4) e_n^6 + 2c_2(-6c_2^5 + 9c_2^3 c_3 - 3c_2^2 c_4 + 2c_3 c_4 \right. \\ &\quad \left. + c_2(-3c_3^2 + c_5)) e_n^7 + \dots \right] \end{aligned} \tag{3.22}$$

From (3.8) and (3.22), we have that

$$\begin{aligned} f(x_n) - 2f(w_n) &= f'(r) \left[e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 \right. \\ &\quad \left. + (c_6 - 4c_2^5 + 2c_2^3 c_3 - 2c_2^2 c_4) e_n^6 + \dots \right] \end{aligned} \tag{3.23}$$

From (3.20) and (3.22), we have that

$$(x_n - w_n) f(w_n) = f'(r) \left[c_2^2(2c_2^3 - c_2 c_3 + c_4) e_n^7 + \dots \right] \tag{3.24}$$

From (3.23) and (3.24), we have that

$$\begin{aligned} \frac{(x_n - w_n) f(w_n)}{f(x_n) - 2f(w_n)} &= c_2^2(2c_2^3 - c_2 c_3 + c_4) e_n^6 + 2c_2(-6c_2^5 + 9c_2^3 c_3 - 3c_2^2 c_4 + 2c_3 c_4 \\ &\quad + c_2(-3c_3^2 + c_5)) e_n^7 + \dots \end{aligned} \tag{3.25}$$

Let $x_{n+1} = w_n - \frac{(x_n - w_n) f(w_n)}{f(x_n) - 2f(w_n)}$ and from (3.19) and (3.25), we have that

$$x_{n+1} = r + 2c_2(6c_2^5 - 9c_2^3 c_3 + 3c_2^2 c_4 - 2c_3 c_4 - c_2(-3c_3^2 + c_5)) e_n^7 + \dots \tag{3.26}$$

Then

$$e_{n+1} = 2c_2(6c_2^5 - 9c_2^3 c_3 + 3c_2^2 c_4 - 2c_3 c_4 - c_2(-3c_3^2 + c_5)) e_n^7 + O(e_n^8) \tag{3.27}$$

This imply that the Algorithm (*) has seven order convergence. \square

4 Numerical Experiments

All computations were done using MAPLE using 100 digit floating-point arithmetic. We accept an approximate solution rather than the exact root, depending on the precision (ε) of the computer. So, the following stopping criteria are used for computer programs:

1. $|x_{n+1} - x_n| < \varepsilon$
2. $|f(x_{n+1})| < \varepsilon$.

We present some numerical test results for various iterative schemes, in Table 1. Compared were the Abbasbandy's method (AM)[16], the Homeier's method (HM)[17], the method of M. Aslam Noor (NR)[18] and the Chun's method (CM)[19]. We used $\varepsilon = 10^{-15}$ and the test functions with initial guess solution x_0 as follows:

1. $f_1(x) = \sin^2 x - x^2 + 1, x_0 = -1$
2. $f_2(x) = x^2 - e^x - 3x + 2, x_0 = 2$
3. $f_3(x) = \cos x - x, x_0 = 1.7$
4. $f_4(x) = (x - 1)^3 - 1, x_0 = 3.5$
5. $f_5(x) = x^3 - 10, x_0 = 1.5$
6. $f_6(x) = xe^{x^2} - \sin^2 x + 3 \cos x + 5, x_0 = -2$
7. $f_7(x) = e^{x^2+7x-30} - 1, x_0 = 3.5$.

As convergence criterion, it was required that the distance of two consecutive approximations δ for the zero was less than $\varepsilon = 10^{-15}$. Also displayed are the number of iterations to approximate the zero (IT), the approximate zero x_n , and the value $|f(x_{n+1})|$.

Table 1. Comparison of numerical methods.

	IT	x_n	$ f(x_{n+1}) $	δ
$f_1, x_0 = -1$				
NM	7	-1.4044916482153412260350868178	$1.04E - 50$	$7.33E - 26$
AM	5	-1.4044916482153412260350868178	$5.81E - 55$	$1.39E - 18$
HM	4	-1.4044916482153412260350868178	$5.4E - 62$	$7.92E - 21$
CM	5	-1.4044916482153412260350868178	$2.0E - 63$	$1.31E - 17$
NR	5	-1.4044916482153412260350868178	$1.3E - 40$	$5.4E - 31$
Algorithm (*)	3	-1.4044916482153412260350868178	$0.1E - 98$	$0.32E - 14$
$f_2, x_0 = 2$				
NM	6	.25753028543986076045536730494	$2.93E - 55$	$9.1E - 28$
AM	5	.25753028543986076045536730494	$1.0E - 63$	$1.45E - 26$
HM	5	.25753028543986076045536730494	0	$9.33E - 43$
CM	4	.25753028543986076045536730494	$1.0E - 63$	$9.46E - 29$
NR	4	.25753028543986076045536730494	$9.3E - 28$	$2.5E - 28$
Algorithm (*)	3	.25753028543986076045536730494	$1.0E - 98$	$0.54E - 27$

Table 1. Comparison of numerical methods (Cont.).

	IT	x_n	$ f(x_{n+1}) $	δ
$f_3, x_0 = 1.7$				
NM	5	.73908513321516064165531208767	$2.03E - 32$	$2.34E - 16$
AM	4	.73908513321516064165531208767	$7.14E - 47$	$8.6E - 16$
HM	4	.73908513321516064165531208767	$5.02E - 59$	$9.64E - 20$
CM	4	.73908513321516064165531208767	0	$1.86E - 53$
NR	4	.73908513321516064165531208767	$3.7E - 54$	$2.4E - 54$
Algorithm (*)	2	.73908513321516064165531208767	$0.20E - 47$	$0.44E - 06$
$f_4, x_0 = 3.5$				
NM	8	2	$2.06E - 42$	$8.28E - 22$
AM	5	2	0	$4.3E - 22$
HM	5	2	0	$1.46E - 24$
CM	5	2	0	$2.74E - 24$
NR	5	2	0	$5.8E - 24$
Algorithm (*)	3	2	$0.56E - 44$	$0.34E - 06$
$f_5, x_0 = 1.5$				
NM	7	2.1544346900318837217592935665	$2.06E - 54$	$5.64E - 28$
AM	5	2.1544346900318837217592935665	$5.0E - 63$	$1.18E - 25$
HM	4	2.1544346900318837217592935665	$5.0E - 63$	$9.8E - 23$
CM	5	2.1544346900318837217592935665	$5.0E - 63$	$1.57E - 22$
NR	5	2.1544346900318837217592935665	$8.1E - 45$	$5.8E - 46$
Algorithm (*)	3	2.1544346900318837217592935665	$0.31E - 98$	$0.92E - 16$
$f_6, x_0 = -2$				
NM	9	-1.2076478271309189270094167584	$2.27E - 40$	$2.73E - 21$
AM	6	-1.2076478271309189270094167584	$4.0E - 63$	$4.35E - 45$
HM	6	-1.2076478271309189270094167584	$4.0E - 63$	$2.57E - 32$
CM	6	-1.2076478271309189270094167584	$4.0E - 63$	$2.1E - 36$
NR	6	-1.2076478271309189270094167584	$1.0E - 37$	$4.9E - 39$
Algorithm (*)	4	-1.2076478271309189270094167584	$0.5E - 98$	$0.34E - 19$
$f_7, x_0 = 3.5$				
NM	13	3	$1.52E - 47$	$4.2E - 25$
AM	7	3	$4.33E - 48$	$2.25E - 17$
HM	8	3	$2.0E - 62$	$2.43E - 33$
CM	8	3	$2.0E - 62$	$2.12E - 23$
NR	8	3	$5.0E - 25$	$3.8E - 26$
Algorithm (*)	5	3	$0.17E - 37$	$0.33E - 08$

Remark 4.1. The order of convergence of the iterative method defined by Algorithm (*) is 7. Per iteration of Algorithm (*) requires three evaluations of the function and two evaluation of first derivative. We take into account the definition of efficiency index (see [13]), if we suppose that all the evaluations have the same cost as function one, we have that the efficiency index of Algorithm(*) is $7^{frac{15}} \approx 1.475$, which is better than $2^{frac{12}} \approx 1.4142$ of the Steffensens method (SM) (see [14]), $3^{frac{14}} \approx 1.3161$ of the (DHM) method (see [15]).

5 Conclusion

In this paper, a new iterative method for solving of nonlinear equations is presented. This method based on a Halley and Householder iterative methods.

The error equations are given theoretically to show the technique has seventh-order convergence. This method attain efficiency index of 1.475, which makes it competitive. In addition, the numerical experiments has been tested on a series of examples published in the literature and show good results when compared it with the previous literature.

Acknowledgements : The authors would like to thank to Faculty of Science, Mahasarakham University for financial support and thankful to reviewers for their valuable comments in improvement of the paper.

References

- [1] I.K. Argyros, S.K. Khattri, An improved semilocal convergence analysis for the Chebyshev method, *Journal of Applied Mathematics and Computing* 42 (1-2) (2013) 509-528.
- [2] S.K. Khattri, T. Steihaug, Algorithm for forming derivative-free optimal methods, *Numer. Algor.* 65 (2014) 809-824.
- [3] M.A. Noor, Fifth-order convergent iterative method for solving nonlinear equations using quadrature formula, *J. Math. Control Sci. Appl.* 1 (2007) 241-249.
- [4] S.K. Khattri, T. Log, Constructing third-order derivative-free iterative methods, *Int. J. Comput. Math.* 88 (7) (2011) 1509-1518.
- [5] S.K. Khattri, Quadrature based optimal iterative methods with applications in high precision computing, *Numer. Math. Theor. Meth. Appl.* 5 (2012) 592-601.
- [6] M.A. Hafiz, A new combined bracketing method for solving nonlinear equations, *Journal of Mathematical and Computational Science* 3 (1) (2013) 87-93.
- [7] M.A. Hafiz, S.M.H. Al-Goria, Solving nonlinear equations using a new tenth- and seventh-order methods free from second derivative, *International Journal of Differential Equations and Applications* 12 (3) (2013) 169-183.
- [8] M.A. Hafiz, S.M.H. Al-Goria, New ninth- and seventh-order methods for solving nonlinear equations, *European Scientific Journal* 8 (27) (2012) 83-95.
- [9] M.S.M. Bahgat, New two-step iterative methods for solving nonlinear equations, *J. Math. Research* 4 (3) (2012) 128-131.
- [10] A. Melman, Geometry and convergence of Halley's method, *SIAM Rev.* 39 (4) (1997) 728-735.
- [11] K.I. Noor, M.A. Noor, Predictor-corrector Halley method for nonlinear equations, *Appl. Math. Comput.* 188 (2007) 1587-1591.

- [12] K.I. Noor, M.A. Noor, S. Momani, Modified Householder iterative method for nonlinear equations, *Appl. Math. Comput.* 190 (2007) 1534-1539.
- [13] M.A. Noor, W.A. Khan, K.I. Noor, E. AI-Said, Higher-order iterative methods free from second derivative for solving nonlinear equations, *International Journal of the Physical Sciences* 8 (2011) 1887-1893.
- [14] M.A. Hafiz, Solving nonlinear equations using Steffensen-type methods with optimal order of convergence, *Palestine Journal of Mathematics* 3 (1) (2014) 113-119.
- [15] M. Dehghan, M. Hajarian, Some derivative free quadratic and cubic convergence iterative formula for solving nonlinear equations, *Comp. & App. Math.* 29 (2010) 19-30.
- [16] S. Abbasbandy, Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method, *Appl. Math. Comput.* 145 (2003) 887-893.
- [17] H.H. Homier, On Newton-type methods with cubic convergence, *J. Comput. Appl. Math.* 176 (2005) 425-432.
- [18] M.A. Noor, Three-step iterative methods for nonlinear equations, *Appl. Math. Comput.* 183 (2006) 322-327.
- [19] C. Chun, Iterative methods improving Newtons method by the decomposition method, *Comput. Math. Appl.* 50 (2005) 1559-1568.
- [20] J.R. Sharma, R. Sharma, Modified Jarratt method for computing multiple roots, *Appl. Math. Comput.* 217 (2010) 878-881.
- [21] S. Li, L. Cheng, B. Neta, Some fourth-order nonlinear solvers with closed formulae for multiple roots, *Comput. Math. Appl.* 59 (2010) 126-135.

(Received 7 May 2016)

(Accepted 7 October 2016)