Palestine Journal of Mathematics

Vol. 6(1)(2017) , 338–344

An efficient newly developed third order iterative method for solving non linear equations

Muhammad Ahsan Tufail, Armaghan Arshad and Muhammad Zeeshan

Communicated by Ayman Badawi

MSC 2010 Classifications: Primary 65Hxx; Secondary 65H05,41A25.

Keywords and phrases: Non linear equations, iterative methods, order of convergence, numerical examples.

We like to point out that all authors contributed equally in this article.

Abstract. Several two or three step iterative methods, by combining previously known methods, have been developed for solving non linear equations. In this paper, we have derived an iterative method by using Taylor series and some previously known methods. The aim of the paper is to develop a method that is cost effective and better in speed of convergence than other existing iterative methods of same order of convergence. We have proved that the proposed method has third order of convergence. To check the efficiency of proposed method, we have presented a comparison of the proposed method with some previously known methods on the base of cost, speed of convergence and on the CPU time a method take to solve an equation. The obtained numerical result shows that the proposed method can compete better with other methods.

1 Introduction

To solve the most basic problem i.e. f(x)=0 of Numerical analysis, a large no. of methods have been found. See [1]-[14]. Some methods have derivatives in their formula's while some methods are derivative free methods. Some authors have found the formula by using Taylor series, quadrature formula's while some authors have combined two different methods to obtain a high order convergence method. Recently a number of methods have been developed by combining two different formula's see [10]-[14]. Inspired by the way author's are presenting their methods for solving a non linear equation we have suggested a two step iterative method for solving a non linear equation by combining the Taylor series with Newton method and the method proposed by Ujevic [10].
Consider the Taylor series,

$$f(x) = f(x_n) + f'(x_n)(x - x_n) + \frac{f''(x_n)(x - x_n)^2}{2} + ... \tag{1.1}$$

Let x is the root, then $f(x) = 0$. Putting $f(x) = 0$ in Eq 1.1

$$0 = f(x_n) + f'(x_n)(x - x_n) + \frac{f''(x_n)(x - x_n)^2}{2} + ... \tag{1.2}$$

Neglecting double derivative $(f''(xn))$ and high order derivative terms, Eq 1.2 becomes

$$0 = f(x_n) + f'(x_n)(x - x_n) \tag{1.3}$$

$$\frac{-f(x_n)}{f'(x_n)} = (x - x_n) \tag{1.4}$$

$$x_n - \frac{f(x_n)}{f'(x_n)} = x \tag{1.5}$$

$$x = x_n - \frac{f(x_n)}{f'(x_n)} \tag{1.6}$$

Let $x = x_{n+1}$,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{1.7}$$

Eq 1.7 is known as Newton Raphson method.
Exapanding Taylor series to second order derivative, a new method known as Halley method is developed, which is defined by Eq 1.8 as

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{(2(f'(x_n))2 - f(x_n)f''(x_n))} \tag{1.8}$$

Ujevic [10] derived an iterative method, by using quadrature rules,

$$z_n = x_n - \frac{af(x_n)}{f'(x_n)} \tag{1.9}$$

Where $a = (0, 1)$.

$$x_{n+1} = x_n + \frac{4(z_n - x_n)f(x_n)}{(3f(x_n) - 2f(z_n))} \tag{1.10}$$

Eq 1.10 is proposed by Nend Ujevic.

2 Derivation and convergence analysis of proposed method

Consider the Taylor series,

$$f(x) = f(x_n) + f'(x_n)(x - x_n) + \frac{f''(x_n)(x - x_n)^2}{2!} + ... \tag{2.1}$$

Putting $f(x) = 0$ in Eq 2.1

$$0 = f(x_n) + f'(x_n)(x - x_n) + \frac{f''(x_n)(x - x_n)^2}{2} \tag{2.2}$$

$$-f(x_n) = f'(x_n)(x - x_n) + \frac{f''(x_n)(x - x_n)^2}{2} \tag{2.3}$$

$$\frac{-f(x_n)}{[f'(x_n) + 1/2f''(x_n)(x - x_n)]} + x_n = x \tag{2.4}$$

$$x = \frac{-f(x_n)}{[f'(x_n) + 1/2f''(x_n)(x - x_n)]} + x_n \tag{2.5}$$

Let $x = x_{n+1}$,

$$x_{n+1} = \frac{-f(x_n)}{[f'(x_n) + 1/2f''(x_n)(x_{n+1} - x_n)]} + x_n \tag{2.6}$$

From Eq 1.10

$$x_{n+1} = x_n + \frac{4(z_n - x_n)f(x_n)}{(3f(x_n) - 2f(z_n))} \tag{2.7}$$

Where $z_n = x_n - \frac{af(x_n)}{f'(x_n)}$
We choose $a = 0.5$.

$$z_n = x_n - \frac{0.5f(x_n)}{f'(x_n)} \tag{2.8}$$

From Eq 2.7,

$$x_{n+1} - x_n = \frac{4(z_n - x_n)f(x_n)}{(3f(x_n) - 2f(z_n))} \tag{2.9}$$

Putting $x_{n+1} - x_n$ value in Eq 2.6,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \frac{f''(x_n)}{2}\left(\frac{4(z_n - x_n)f(x_n)}{(3f(x_n) - 2f(z_n))}\right)} \tag{2.10}$$

Eq 2.8 and Eq 2.10 constitute an iterative method. The algorithm of proposed two step iterative method is defined by Eq 2.8 and Eq 2.10. The arrangement is given below

$z_n = x_n - \frac{0.5f(x_n)}{f'(x_n)}$

$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)+\frac{f''(x_n)}{2}(\frac{4(z_n-x_n)f(x_n)}{(3f(x_n)-2f(z_n))})}$

Lemma 2.1. *Let f is a differentiable function and has root r in the open interval H i.e. $f : H \subseteq R \to R$ and $r \in H$. Consider an iterative method $G(x)$, with the sequence x_n and rate of convergence P, then*

$$x_{n+1} - r = c(x_n - r)^P + O[x_n - r]^{P+1} \tag{2.11}$$

Let $x_{n+1} - r = e_{n+1}$, and $x_n - r = en$, $c \neq 0$.

$$e_{n+1} = c(e_n)^P + O[e_n]^{p+1} \tag{2.12}$$

Proof.
Applying Taylor series on iterative method $G(x)$,

$$x_{n+1} - r = G(x_n) \tag{2.13}$$

$$e_{n+1} = G(r) + G^{'}(r)e_n + \frac{(G^{''}(r)e_n^2)}{2!} + \cdots + \frac{(G^p(r)e_n^p)}{p!} + O[e_n]^{p+1} \tag{2.14}$$

As r is the root so, Putting $G(r) = 0$.

$$e_{n+1} = \frac{(G^p(r)e_n^p)}{p!} + O[e_n]^{p+1} \tag{2.15}$$

lemma is proved.

Theorem 2.2. *Let f is a differentiable function $f : H \subseteq R \to R$ and has root r in the open interval H i.e. $r \in H$ if x_o is sufficiently close to r then algorithm defined by Eq 2.10 has cubic convergence and satisfies the following error equation:*

$$e_{n+1} = (\frac{(0.75c_2^2)}{c_1^2} - \frac{c_3}{c_1})e_n^3 + O[e_n]^4 \tag{2.16}$$

Proof.
Applying Taylor series, exapanding function about r, we get

$$f(x_n) = c_1e_n + c_2e_n^2 + c_3e_n^3 + \cdots \tag{2.17}$$

$$f'(xn) = c_1 + 2c_2e_n + 3c_3e_n^2 + \cdots \tag{2.18}$$

$$f''(x_n) = 2c_2 + 6c_3e_n + \cdots \tag{2.19}$$

From Eq 2.17,Eq 2.18, previously define Eq 2.8 becomes

$$z_n = 0.5e_n + \frac{(0.5c_2e_n^2)}{c_1} + (\frac{(-c_2^2 + c_1c_3)e_n^3}{c_1^2}) + O[e_n]^4 \tag{2.20}$$

$$f(z_n) = 0.5c_1e_n + 0.75c_2e_n^2 + (\frac{(0.25c_2^2)}{c_1} + 0.5(\frac{(0.5c_2^2)}{c_1} + 0.25c_3) + \frac{(-c_2^2 + c_1c_3)}{c_1})e_n^3 + O[e_n]^4 \tag{2.21}$$

Eq 2.9 from Eq 2.20 and Eq 2.21 becomes

$$\frac{4(z_n - x_n)f(x_n)}{(3f(x_n) - 2f(z_n))} = -e_n + \frac{(0.75c_2e_n^2)}{c_1} + \frac{((-1.0625c_2^2 + 1.375c_1c_3)e_n^3)}{c_1^2} + O[e_n]^4 \tag{2.22}$$

Now putting all above equation results in Eq 2.10, we get

$$e_{n+1} = (\frac{(0.75c_2^2)}{c_1^2} - \frac{c_3}{c_1})e_n^3 + O[e_n]^4 \tag{2.23}$$

Hence proved that, the proposed method has cubic convergence.
Some other approaches that are commonly used to find the order of convergence are defined in definition 2.3 and 2.4.

Definition 2.3. Computational local order of convergence (COC): Let there is a function $f(x)$ and has root a and suppose that x_{n+1}, x_n, and x_{n-1} are three consecutive iterations closer to the root a. Then, the computational order of convergence ρ can be approximated using the formula

$$\rho \approx \frac{\ln|(x_{k+1} - a)/(x_k - a)|}{\ln|(x_k - a)/(x_{k-1} - a)|} \tag{2.24}$$

Where a is the root of equation and ρ is the order of convergence.

Now we define another approach to find order of convergence. This approach does not depend on the root of equation.

Definition 2.4. Approximated computational local order of convergence (ACOC): Let there is a function $f(x)$ and has root a and suppose that x_{n+1}, x_n, and x_{n-1} are three consecutive iterations closer to the root a. Then, the approximated computational order of convergence ρ can be approximated using the formula

$$\rho \approx \frac{\ln|(x_{k+1} - x_k)/(x_k - x_{k-1})|}{\ln|(x_k - x_{k-1})/(x_{k-1} - x_{k-2})|} \tag{2.25}$$

3 Numerical Results

Some numerical examples are presented in this section, to analyze the performance of proposed method. In most cases the performance of method is analyzed on the base of no.of iterations a method takes to converge to a desired solution. In this paper, we have analyze the proposed method on the base of cost, speed of convergence and the CPU time a method takes to converge to a desired solution. The cost of method depends on the no.of function evaluations a method take to converge to the required solution. Seven examples are presented in the paper and it is observed that the proposed method is better in cost, CPU time and in the speed of convergence compared to other third order iterative methods. The method does not diverges on points where other methods diverges. The tolerance considered is $\varepsilon = 10^{-15}$. The stopping criteria used is

(i) $|f(x_{n+1})| < \varepsilon$

(ii) $|(x_{n+1}) - x_n| < \varepsilon$

Iteration limit is 200. All examples are taken from [14].

$f_1(x) = sin(x)^2 - x^2 + 1$
$f_2(x) = (x)^2 - e^x - 3x + 2$
$f_3(x) = x - cos(x)$
$f_4(x) = (x - 1)^3 - 1$
$f_5(x) = (x)^3 - 10$
$f_6(x) = xe^{x^2} - sin(x)^2 + 3cos(x) + 5$
$f_7(x) = e^{(x^2+7x-30)} - 1$

Now we define some third order iterative methods with which we compare our results.
Chun (CN) method [12] is defined as

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} \tag{3.1}$$

$$x_{n+1} = x_n - \frac{(f(x_n) + f(y_n))}{f'(x_n)} \tag{3.2}$$

Noor method [14] is defined as

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} \tag{3.3}$$

$$x_{n+1} = x_n - [\frac{(3f'(x_n) - f'(y_n))}{(2f'(x_n))}]\frac{f(x_n)}{f'(x_n)} \tag{3.4}$$

NNM method [13] is defined as

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} \tag{3.5}$$

$$z_n = \frac{-f(y_n)}{f'(x_n)} \tag{3.6}$$

$$x_{n+1} = y_n - \frac{f(y_n + z_n)}{f'(x_n)} \tag{3.7}$$

Proposed method	NNM	CN	[14]
4	4	3	3

Table 1. No.of function evaluations per iteration of iterative methods

Functions	NI	NFE	COC	ACOC	CPU
$f_1(x), x_o = 1.1$					
Proposed Method	3	12	2.8	2.8	0.011891
CN	4	12	2.56	2.56	0.01263
NNM	6	24	2.9	2.9	0.028085
[14]	4	12	2.67	2.67	0.014188
$f_2(x), x_o = -1.75$					
Proposed Method	3	12	2.97	2.97	0.011656
CN	4	12	2.89	2.89	0.012404
NNM	5	20	2.9	2.9	0.014273
[14]	4	12	2.65	2.65	0.027215
$f_3(x), x_o = -3.5$					
Proposed Method	8	32	3	3	0.021076
CN	37	111	3	3	0.056639
NNM	DIV	-	-	-	-
[14]	11	33	3	3	0.028042
$f_4(x), x_o = -3.5$					
Proposed Method	17	68	3	3	0.043047
CN	73	219	2.97	2.97	0.114071
NNM	19	76	3	3	0.044074
[14]	161	483	2.88	2.88	0.326647
$f_5(x), x_o = -0.5$					
Proposed Method	15	60	3	3	0.031167
CN	20	60	3	3	0.033619
NNM	54	216	3	3	0.101928
[14]	67	201	3	3	0.137171
$f_6(x), x_o = 0$					
Proposed Method	33	132	2.98	2.98	0.101265
CN	DIV	-	-	-	-
NNM	DIV	-	-	-	-
[14]	DIV	-	-	-	-
$f_7(x), x_o = 3.5$					
Proposed Method	6	24	3	3	0.020078
CN	8	24	3	3	0.020972
NNM	10	40	3	3	0.027333
[14]	9	27	3	3	0.030364

Table 2. comparison of iterative methods

DIV= Diverges.
NI= No.of iterations.
NFE= No.of function evaluations.
CPU= CPU time a method take to solve a non linear equation.

4 Conclusion

We have suggested and analyzed, a new two step iterative method with cubic convergence. The efficiency of proposed method is analyzed by different numerical examples. The numerical results are the evident of the performance of the method. From the numerical results, we conclude that the proposed method is better in speed of convergence, cost and in CPU time a method take to solve a non linear equation.

References

[1] E. Halley, A new, exact and easy method for finding the roots of equations generally, and that without any previous reduction, *Philos. R. Soc. London.* **18**, 136–147 (1964)

[2] A.M. Ostrowski, *Solution of Equations in Euclidean and Banach Space*, 3rd ed.; Academic Press, New York (1973).

[3] J.F. Traub, *Iterative Methods for the Solution of Equations*, Prentice-Hall. Englewood Cliffs, N J (1964).

[4] P. A. Jarratt, *Review of Methods for Solving Nonlinear Algebraic Equations*, Gordon and Breach. Science Publishers, London (1970).

[5] X.Y. Wu and H.W. Wu, On a class of quadratic convergence iteration formulae without derivatives,*Applied Mathematics and Computation.* **10 (7)**, 77–80 (2000)

[6] G. Alefeld and F.A. Potra, Some efficient methods for enclosing zeroes of nonlinear equations,*BIT.* **32**, 334–344 (1992)

[7] S. Abbasbandy, Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method,*Applied Mathematics and Computation.* **145**, 887–893 (2003)

[8] Chun, C. Iterative methods improving Newton's method by the decomposition method.,*Computers and Mathematics with Applications.* **50**, 1559–1568 (2005)

[9] M.Grau and J.L. Diaz-Barrero, An improvement to Ostrowski root-finding method, *Applied Mathematics and Computation.* **173** 450–456 (2006).

[10] N. Ujevic, A method for solving nonlinear equations, *Applied Mathematics and Computation.* **174**, 1416–1426 (2006)

[11] M.A. Noor, F. Ahmad and S. Javeed, Two-Step Iterative Methods for Nonlinear Equations. *Applied Mathematics and Computation.* , **181**, 1068–1075 (2006).

[12] C. Chun, Iterative methods improving Newton's method by the decomposition method, *Computers and Mathematics with Applications.* **50**, 1559–1568 (2005).

[13] M.A. Noor and K.I. Noor, Three-step iterative methods for nonlinear equations, *Applied Mathematics and Computation.* **183**, 322–327 (2006).

[14] M.A. Noor , K.I. Noor and M. Hassan, Third-order iterative methods free from second derivatives for nonlinear equation, *Applied Mathematics and Computation.* **190**,1551–1556 (2007).

Author information

Muhammad Ahsan Tufail, COMSATS Institute of Information Technology, Attock, Pakistan.
E-mail: m.ahsantufail@gmail.com

Armaghan Arshad, COMSATS Institute of Information Technology, Attock, Pakistan.
E-mail: arshadarmaghan@yahoo.com

Muhammad Zeeshan, WAH Engineering College, WAH Cantt, Pakistan.
E-mail: zeeshanmalik44@yahoo.com

Received: September 5, 2015.

Accepted: February 3, 2016.