Available online at http://scik.org
J. Math. Comput. Sci. 2022, 12:104
https://doi.org/10.28919/jmcs/6949
ISSN: 1927-5307

# AN IMPROVED NEWTON-RAPHSON METHOD WITH QUADRATIC CONVERGENCE FOR SOLVING NONLINEAR TRANSCENDENTAL EQUATIONS

K. VENKATESHWARLU[1], G. MAHESH[2,*], G. SWAPNA[3]

[1]Department of Mathematics, Anurag University, Hyderabad, Telangana, India

[2]Department of Humanities and Sciences,

Keshav Memorial Institute of Technology Narayanaguda, Hyderabad, Telangana, India

[3]Department of Humanities and Sciences,

Geethanjali College of Pharmacy, Cheeryal(V), Keesara(M), Medchal Dist. Telangana, India



**Abstract.** This paper presents a new scheme to find a non-zero positive root of the non-linear single variable equations. The proposed method is based on the grouping of the arc sine series and Newton-Raphson method. The proposed method is implemented in MATLAB and is applied to various models of problems to ensure the methods applicability. The proposed method is studied on number of numerical examples and results signify that our method is better and more effective as comparable to renowned methods. These results are depicted through error analysis. The convergence of proposed method is proven to be quadratically convergence.

**Keywords:** iteration method; nonlinear equations; quadratic convergence; root finding method.

**2010 AMS Subject Classification:** 65H04, 65H05.

## 1. INTRODUCTION

A Root finding method is an emerging field which has huge applications in science and Engineering. Some of these applications are in circuit analysis, analysis of state equations for a

*Corresponding author

E-mail address: gattumahesh790@gmail.com

Received October 30, 2021

real gas, mechanical motions / oscillations, weather forecasting, in optimization and many other fields of engineering. Also have applications in different models such as study of splitting of water vapor into oxygen and hydrogen at higher temperature and estimation of Oxygen level c (mg/l) in a river downstream from a sewage discharge. In [1], Chen and Li have used modified Regula Falsi method to solve the nonlinear equation. This method is shown to be quadratically convergent for the sequence of diameters and the sequence of iterative points.

In [2], these authors have developed a class of exponential iterative formulae for enclosing simple zeros of nonlinear equation which is also a quadratically convergent. Mamta et al. [3], present two different iterative formulae with quadratic convergence which can be used an alternative to Newton's method or in cases where Newton's method is not successful. In [4, 5] Srinivasarao and his team developed a scheme based on exponential series where the convergence is greater than 2 to compute a non-zero real root of the transcendental equations. In [6], Srinivasarao Thota and Vivek Kumar Srivastav presented an algorithm to find a non-zero real root of the transcendental equations using hyperbolic tangent function with convergence greater than 2. In [7], Baccouch derive a new family of high order derivative-free iteration methods for finding simple and multiple roots of nonlinear algebraic equations. This study is an extension of the second-order Steffensen's method. The idea is to modify the family of derivative-based methods, which are analyzed by the author, to obtain derivative-free methods.

Chun [8] proposed a simple, yet powerful, and simply implementable approach for creating Newton-like iteration formulae for computing nonlinear equation solutions. The novel methodology is based on the homotopy analysis method, which is used to solve equations that are equivalent to nonlinear equations in general form. Yun and Petkovic [9] developed a simple iterative method to remedy the aforementioned drawbacks of the Newton method and the secant method. This approach was quadratically convergent without the need for a derivative of the function f(x) or the time spent selecting a good starting guess. Pinkham and Sansiribhan [10] used Hybrid Algorithm based on Bisection and Modified Newton's Method for evaluation of nth root. This procedure is the combination of inverse sine, Bisection method and Newton's method. The results obtained by this method converges if $z > 1$ with the initial interval [1, z] and if $0 < z < 1$, it converges with initial interval [0, 4]. In [11], the authors extended the work done by Chen and Li [12] in their paper. In [12], the authors presented a class of regula falsi iterative formulae for solving nonlinear equations. In [11], Gupta and Parhi accelerate further the convergence of the method of [11] from quadratic to cubic. In [13], a Review of Bracketing Methods for Finding Zeros of Nonlinear Function is done by comparing the results of Bracketing Method with the standard root finding methods.

Mahesh et al. [14] introduced an Iterative method for solving non-linear transcendental equations through a series approximation. This algorithm is better in reducing error rapidly, hence converges faster as compared to the existing methods. Venkateshwarlu et al. [15] used a procedure which is the combination of the inverse $\tan(x)$ function and the Newton-Raphson method. This method is tested on number of numerical examples and results indicate that this method is better and more effective as compared to well-known methods. This approach will also help to produce a non-zero real root of a given non-linear equations (transcendental, algebraic, and exponential) in the commercial package. In [16] Yun developed a derivative free simple iterative formula which is similar to Muller's method. This iterative formula converges quadratically.

## 2. Proposed Method

The proposed formula using inverse sine series is proposed as

$$x_{n+1} = x_n \left[1 + \frac{1}{2}\arcsin\left(\frac{-2f(x_n)\sqrt{(x_n f'(x_n))^2 - f^2(x_n)}}{(x_n f'(x_n))^2}\right)\right],\ n = 0,1,2,\ldots \tag{1}$$

### 2.1. Convergence Analysis.

**Theorem 1.** *Suppose $\alpha \neq 0$ is a real exact root of $f(x)$ and $\theta$ is a sufficiently small neighborhood of $\alpha$. Let $f''(x)$ exists and $f'(x) \neq 0$ in $\theta$. Then the proposed iterative formula in equation* (1) *generates a quadratically convergent series of iterations.*

***Proof:***

*The given proposed formula for new approximation is*

$$x_{n+1} = x_n \left[1 + \frac{1}{2}\arcsin\left(\frac{-2f(x_n)\sqrt{(x_n f'(x_n))^2 - f^2(x_n)}}{(x_n f'(x_n))^2}\right)\right],\ n = 0,1,2,\ldots \tag{2}$$

*And using the fact,* $\lim_{x_n\to\alpha}\left[1+\frac{1}{2}\arcsin\left(\frac{-2f(x_n)\sqrt{(x_nf'(x_n))^2-f^2(x_n)}}{(x_nf'(x_n))^2}\right)\right]=1,$

*Hence* $x_{n+1}=\alpha$

*We know that* $\arcsin(x)=x+\frac{1}{2}\frac{x^3}{3}+\frac{1}{2}\frac{3}{4}\frac{x^5}{5}+\dots$

*by using the above expansion,*

$$\arcsin\left(\frac{-2f(x_n)\sqrt{(x_nf'(x_n))^2-f^2(x_n)}}{(x_nf'(x_n))^2}\right)=\frac{-2f(x_n)\sqrt{(x_nf'(x_n))^2-f^2(x_n)}}{(x_nf'(x_n))^2}$$
$$+\frac{1}{2}.\frac{1}{3}\left(\frac{-2f(x_n)\sqrt{(x_nf'(x_n))^2-f^2(x_n)}}{(x_nf'(x_n))^2}\right)^3+\dots$$

*Substituting in the equation* (2),

$$x_{n+1}=x_n+\frac{1}{2}x_n\left(\frac{-2f(x_n)\sqrt{(x_nf'(x_n))^2-f^2(x_n)}}{(x_nf'(x_n))^2}\right)$$
$$+x_n\frac{1}{2^2}.\frac{1}{3}\left(\frac{-2f(x_n)\sqrt{(x_nf'(x_n))^2-f^2(x_n)}}{(x_nf'(x_n))^2}\right)^3+\dots \tag{3}$$

*Now, equation* (3) *reduces to*

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}\sqrt{1-\frac{f^2(x_n)}{(x_nf'(x_n))^2}}+\dots \tag{4}$$

*Since* $\lim_{x_n\to\infty}\frac{f^2(x_n)}{(x_nf'(x_n))^2}=0$ . *the above equation* (4) *reduces to the Newton-Raphson method, which has quadratic convergence. Therefore, the order of convergence of proposed method is at least* $\geq 2$.

## 2.2. Step by step process of proposed method.

Step 1: Recognize the initial approximations of the root $x_0$ and $x_1$ such that $f(x_0)\times f(x_1)<0$

Step 2: Apply the proposed method (i.e., represented in equation (1)) to evaluate the next approximation root.

Step 3: Repeat the above step 2 until we get the desired precision.

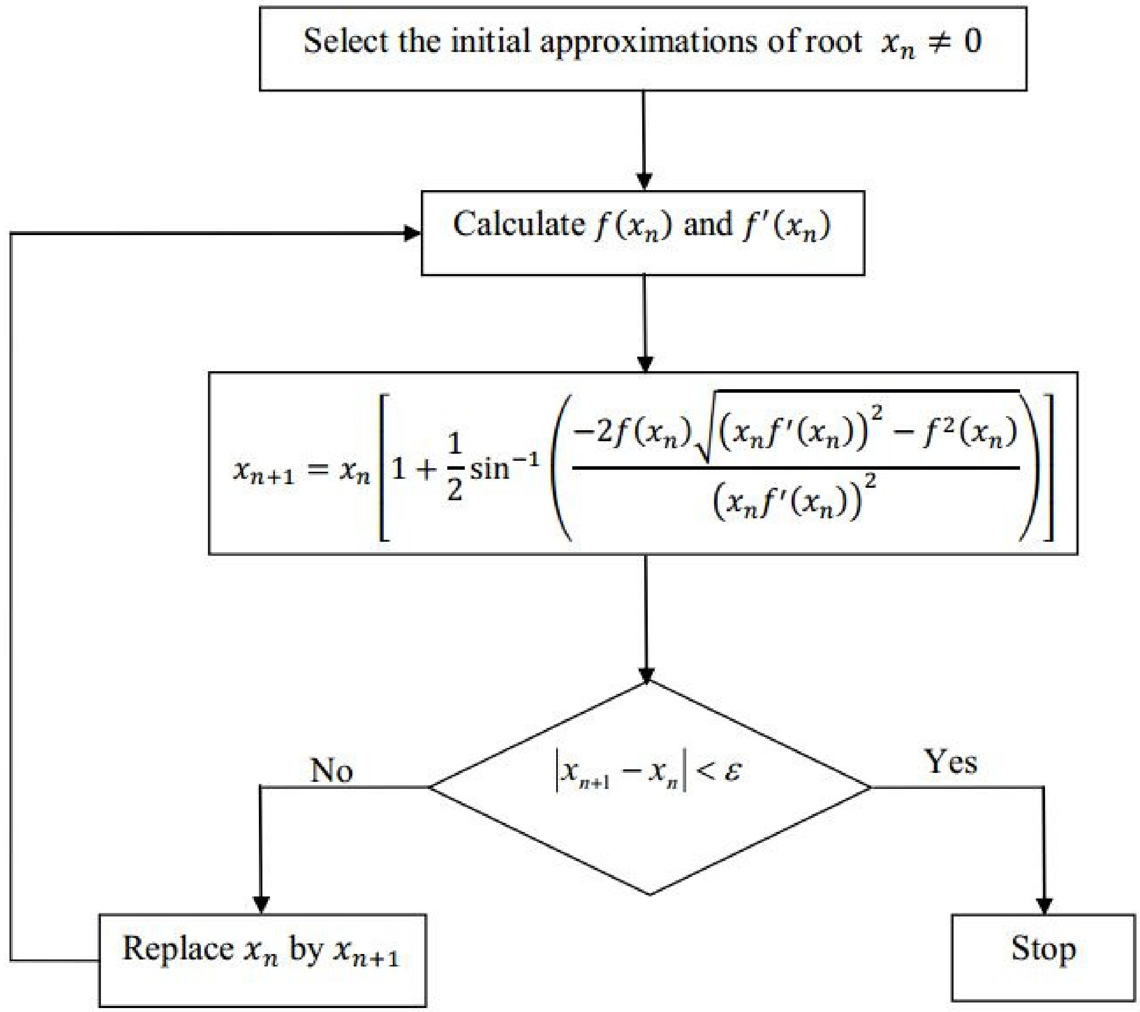


FIGURE 1. Flow chart of proposed method.

**2.3. MATLAB code for the Proposed Method.** In this section MATLAB code for the proposed method is presented

```
clear all
clc
x=zeros(50,1);
x(1)=input('Enter first approximate root i.e., x0 value :');
x(2)=input('Enter second approximate root i.e., x1 value :');
//* choose x0 and x1 such that f(x0) × f(x1) < 0
//* choose appropriate initial approximation let it be x0
```

```
n = 10^(-15);
itr=0;
for i=3:100
t1 = -2*fxn(x(i-1))*sqrt(x(i-1)^2*(fpxn(x(i-1)))^2 - fxn(x(i-1))^2);
t2 = x(i-1)^2*(fpxn(x(i-1)))^2;
x(i) = x(i-1)*(1+0.5*asin(t1/t2));
disp(x(i));
if abs((x(i)-x(i-1))/x(i))*100 < n
root=x(i); disp(itr);
disp(x(i));
error=abs((x(i)-x(i-1))/x(i))*100;
break; end
```

**Function definations**

```
function fx=fxn(x)
fx=Given nonlinear equation ;
end
function fx=fpxn(x)
fx=first order derivative of given nonlinear equation;
end
```

## 3. Comparing With Existing Results

The performance of the proposed method by equation (1), is compared with available results in the literature [12, 17, 18]. From Table 1, shows the proposed method is effective than the method proposed by Xinyuan & Hongwei [17], Zhu & Wu [18] and Chen & Li [12] in terms of number of iterations.

TABLE 1. Comparing number of iterations of the proposed method with Xinyuan & Hongwei [17], Zhu & Wu [18] and Chen & Li [12]

| $f(t)$ | $t_0$ | Xinyuan & Hongwei [17] | Zhu & Wu [18] | Chen & Li [12] | Proposed Method |
|---|---|---|---|---|---|
| $f_1(t) = \log(t)$ | 0.5 | 8 | 4 | 7 | 5 |
| $f_2(t) = t - e^{\sin t} + 1$ | 4 | 9 | 23 | 11 | 4 |
| $f_3(t) = 11t^{11} - 1$ | 1 | Divergent | 6 | 8 | 6 |
| $f_4(t) = te^{-t} - 0.1$ | 0.1 | 5 | 3 | 6 | 3 |

## 4. NUMERICAL EXAMPLES

In this section, the proposed method is applied to some numerical examples to demonstrate its efficacy when compared to traditional methods such as; Bisection, Regula-Falsi and Newton-Raphson methods. All the numerical computations are carried out in MATLAB with accuracy of $10^{-15}$, the maximum iterative numbers are not more than 100. The numerical results obtained are presented in the following tables.

**Example 1:** $f_1(t) = \log(t)$, with $t_0 = 0.5$ and $t_1 = 5$

**Example 2:** $f_2(t) = t - e^{\sin t} + 1$, with $t_0 = 1$ and $t_1 = 4$

**Example 3:** $f_3(t) = 11t^{11} - 1$, with $t_0 = 0.1$ and $t_1 = 1$

**Example 4:** $f_4(t) = te^{-t} - 0.1$, with $t_0 = 0$ and $t_1 = 0.1$

TABLE 2. Comparing number of iterations of by different methods

| Example | Exact root | Bisection method (n) | Regula-Falsi method (n) | Secant method (n) | Proposed method (n) |
|---|---|---|---|---|---|
| 1 | 1.00000000000000 | 50 | 27 | 27 | 5 |
| 2 | 1.69681238680975 | 47 | 32 | 32 | 6 |
| 3 | 0.80413309750367 | 50 | 101 | 110 | 7 |
| 4 | 0.11183255915896 | 48 | 15 | 9 | 3 |

From the above table 2, the results show that the proposed method is adequate in terms of number of iterations in the proposed method is closer to zero than the classical Bisection, Regula falsi and Secant methods.

TABLE 3. Comparing the quantity of iterations produced by different techniques for $f(t) = \log(t)$ with $t_0 = 0.5$ and $t_1 = 5$

| n | BM | $\lvert f(t_n)\rvert$ | n | SM | $\lvert f(t_n)\rvert$ | n | PM | $\lvert f(t_n)\rvert$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 2.75 | 4.3933E-01 | 1 | 1.854634980 | 2.6825E-01 | 1 | 0.882923097 | 5.4077E-02 |
| 2 | 1.625 | 2.1085E-01 | 2 | 1.216307818 | 8.5043E-02 | 2 | 0.993148279 | 2.9859E-03 |
| 3 | 1.0625 | 2.6328E-02 | 3 | 1.058520962 | 2.4699E-02 | 3 | 0.999976526 | 1.0194E-05 |
| 4 | 0.78125 | 1.0720E-01 | 4 | 1.016169349 | 6.9660E-03 | 4 | 0.999999999 | 1.1964E-10 |
| 5 | 0.921875 | 3.5327E-02 | 5 | 1.004494906 | 1.9477E-03 | 5 | 1.000000000 | 0.0000E+00 |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | | | |
| 50 | 1.000000000 | 0.0000E+00 | 27 | 1.000000000 | 0.0000E+00 | | | |

TABLE 4. Comparing the quantity of iterations produced by different techniques for $f(t) = t - e^{\sin t} + 1$ with $t_0 = 1$ and $t_1 = 4$

| n | BM | $\lvert f(t_n)\rvert$ | n | SM | $\lvert f(t_n)\rvert$ | n | PM | $\lvert f(t_n)\rvert$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 2.5 | 1.6806E+00 | 1 | 1.197775115 | 3.3985E-01 | 1 | 1.912366944 | 3.4667E-01 |
| 2 | 1.75 | 7.4902E-02 | 2 | 1.3934301762 | 2.82602E-01 | 2 | 1.725630322 | 3.9673E-02 |
| 3 | 1.375 | 2.9183E-01 | 3 | 1.546346450 | 1.7112E-01 | 3 | 1.697576001 | 1.0232E-03 |
| 4 | 1.5625 | 1.5568E-01 | 4 | 1.635644696 | 7.6929E-02 | 4 | 1.696812959 | 7.6622E-07 |
| 5 | 1.65625 | 5.2131E-02 | 5 | 1.675119087 | 2.8424E-02 | 5 | 1.696812386 | 4.2610E-13 |
| 6 | 1.703125 | 8.5046E-03 | 6 | 1.689613399 | 9.5707E-03 | 6 | 1.696812386 | 2.2204E-15 |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | | | |
| 47 | 1.696812386 | 2.2204E-15 | 32 | 1.696812386 | 2.2204E-15 | | | |

TABLE 5. Comparing the quantity of iterations produced by different techniques for $f(t) = 11t^{11} - 1$ with $t_0 = 0.1$ and $t_1 = 1$

| n | BM | $\lvert f(t_n)\rvert$ | n | SM | $\lvert f(t_n)\rvert$ | n | PM | $\lvert f(t_n)\rvert$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.55 | 9.8468E-01 | 1 | 0.181818181 | 1.0000E+0 | 1 | 0.917261002 | 3.2542E+0 |
| 2 | 0.775 | 3.3364E-01 | 2 | 0.256198341 | 1.0000E+0 | 2 | 0.853423437 | 9.2399E-01 |
| 3 | 0.8875 | 1.9597E+0 | 3 | 0.323816463 | 9.9995E-01 | 3 | 0.816152140 | 1.7727E-01 |
| 4 | 0.83125 | 4.4026E-01 | 4 | 0.385285170 | 9.9969E-01 | 4 | 0.804979768 | 1.1643E-02 |
| 5 | 0.803125 | 1.3704E-02 | 5 | 0.441152816 | 9.9865E-01 | 5 | 0.804137535 | 6.0717E-05 |
| 6 | 0.8171875 | 1.9380E-01 | 6 | 0.491894526 | 9.9551E-01 | 6 | 0.804133097 | 1.6755E-09 |
| 7 | 0.81015625 | 8.5549E-02 | 7 | 0.537897424 | 9.8800E-01 | 7 | 0.8041330975 | 4.1078E-15 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | | | |
| 50 | 0.8041330975 | 4.1078E-15 | 110 | 0.8041330975 | 9.9365E-14 | | | |

TABLE 6. Comparing the quantity of iterations produced by different techniques for $f(t) = te^{-t} - 0.1$ with $t_0 = 0$ and $t_1 = 0.1$

| n | BM | $\lvert f(t_n)\rvert$ | n | SM | $\lvert f(t_n)\rvert$ | n | PM | $\lvert f(t_n)\rvert$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.55 | 2.1732E-01 | 1 | 0.110517091 | 1.0000E+0 | 1 | 0.111712417 | 9.5428E-05 |
| 2 | 0.325 | 1.3482E-01 | 2 | 0.111816133 | 1.3045E-05 | 2 | 0.111832543 | 1.2165E-08 |
| 3 | 0.2125 | 7.1819E-02 | 3 | 0.111832353 | 1.6301E-07 | 3 | 0.111832559 | 1.3878E-17 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | | | |
| 48 | 0.111832559 | 2.3592E-15 | 9 | 0.111832559 | 1.3878E-17 | | | |

## 5. CONCLUSION

In the current study, an attempt made to improve Newton-Raphson method for solving nonlinear transcendental equations. This method is based on algebraic function of inverse sine series. The rate of convergence of the attempted method is discussed and proved to be quadratic. This attempted method is programmed using Matlab software. The numerical computation result predicts that the attempted method is successful in terms of dropping number of iterations with minimized erroneousness. To show that effectiveness of the attempted method, the results are compared with available typical methods like: Bisection, Regula-Falsi, Secant methods. The attempted method has obvious realistic helpfulness from a real-world standpoint.

## CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

## REFERENCES

[1] J. Chen, W. Li, An exponential regula falsi method for solving nonlinear equations, Numer. Algorithms. 41 (2006), 327–338.

[2] J. Chen, W. Li, On new exponential quadratically convergent iterative fromulae, Appl. Math. Comput. 180 (2006), 242–246.

[3] V. K. Mamta, V. K. Kukreja, S. Sing, On a class of quadratically convergent iteration formulae, Appl. Math. Comput. 166 (2005), 633–637.

[4] T. Srinivasarao, G. Tekle, P. Shanmugasundaram, new algorithms for computing a root of non-linear equations using exponential series, Palest. J. Math. 10 (2021), 128-134.

[5] T. Srinivasarao, A New Root–Finding Algorithm Using Exponential Series, Ural Math. J. 5 (2019), 83–90.

[6] T. Srinivasarao, S. Vivek Kumar, An Algorithm to Compute Real Root of Transcendental Equations Using Hyperbolic Tangent Function, Int. J. Open Probl. Comput. Sci. Math. 14 (2021), 1–14.

[7] M. Baccouch, A family of high order derivative-free iterative methods for solving root-finding problems, Int. J. Appl. Comput. Math. 5 ( 2019), 1–31.

[8] C. Chun, Construction of Newton-like iteration methods for solving nonlinear equations, Numer. Math. 104 ( 2006), 297–315.

[9] B.I. Yen, M.S. Petkovic , A quadratically convergent iterative method for non-linear equations, J. Korean Math. Soc. 48( 2011), 487–497.

[10] S. Pinkham, S. Sansiribhan, A New Hybrid Algorithm of Bisection and Modified Newton's Method for the nth root-finding of a Real Number, New J. Phys. 1593 (2020), 12–20.

[11] D.K. Gupta, S.K. Parhi, An improved class of regula falsi methods of third order for solving non-linear equations in R, J. Appl. Math. Comput. 33 (2010), 35–45.

[12] J. Chen, W. Li, An improved exponential regula falsi methods with quadratic convergence of both diameter and point for solving nonlinear equations, Appl. Numer. Math. 57 (2007), 80–88.

[13] S. Intep, A review of bracketing methods for finding zeros of nonlinear functions, Appl. Math. Sci. 12 (2018), 137–146.

[14] G. Mahesh, G. Swapna, K. Venkateshwarlu, An iterative method for solving non-linear transcendental equations, J. Math. Comput. Sci. 10 (2020), 1633–1642.

[15] K. Venkateshwarlu, V.S. Triveni, G. Mahesh, G. Swapna G, A new trigonometrical method for solving non-linear transcendental equations, J. Math. Comput. Sci. 25 (2021), 176–181.

[16] B.I. Yun, Solving nonlinear equations by a new derivative free iterative method, Appl. Math. Comput. 217 (2011), 5768–5773.

[17] X. Wu, H. Wu, On a class of quadratic convergence iteration formulae without derivatives, Appl. Math. Comput. 107 (2000), 77–80.

[18] Y. Zhu, X. Wu, A free-derivative iteration method of order three having convergence of both point and interval for nonlinear equations, Appl. Math. Comput. 137 (2003), 49–55.