

A new trigonometrical method for solving non-linear transcendental equations



K. Venkateshwarlu^a, V. S. Triveni^a, G. Mahesh^{a,*}, G. Swapna^b

^aDepartment of Freshman Engineering, Geethanjali College of Engineering and Technology, Cheeryal(V), Keesara(M), Medchal Dist. Telangana, India.

^bDepartment of Humanities and Sciences, Geethanjali College of Pharmacy, Cheeryal(V), Keesara(M), Medchal Dist. Telangana India.

Abstract

This paper presents a new algorithm to find a non-zero positive real root of the transcendental equations. The proposed method is based on the combination of the inverse $\tan(x)$ function and the Newton-Raphson method. Implementation of the proposed method in MATLAB is applied to different problems to ensure the method's applicability. The proposed method is tested on number of numerical examples and results indicate that our methods are better and more effective as compared to well-known methods. Error calculation has been done for available existing methods and the new proposed method. The errors have been reduced rapidly and obtained the real root in less number of iterations as compared to renowned methods. Certain numerical examples are presented in this paper to show the effectiveness of the proposed method. The Convergence of the proposed method is discussed and shown that the method reduces to Newton-Raphson method that is quadratic convergent. 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.

Keywords: Nonlinear equation, iteration method, transcendental equations.

2020 MSC: 65H04, 65H05.

©2022 All rights reserved.

1. Introduction

Root finding methods have enormous applications in many fields such as Finding Methods Applied to Digital Maximum Power Point tracking of sustainable photovoltaic energy generation, computation of gradient retention times in liquid chromatography, for solving non-linear differential equations, in circuit analysis, analysis of state equations for a real gas, mechanical motions/oscillations, weather forecasting, in optimization and many other fields of engineering designing processes. Root finding methods can also be applied in the discrete stochastic arithmetic (DSA) to validate the class of multi-step iterative methods and find the optimal numerical solution of non-linear equations.

In [5], Gemechu used derivative estimations up to the third-order (in root finding, some new initiatives) are applied in Taylor's approximation of a non-linear function/equation to achieve efficient iterative

*Corresponding author

Email address: gattumahesh.fe@gcet.edu.in (G. Mahesh)

doi: [10.22436/jmcs.025.02.06](https://doi.org/10.22436/jmcs.025.02.06)

Received: 2021-02-16 Revised: 2021-04-10 Accepted: 2021-05-01

methods. Competent methods of higher order for solving simple roots of nonlinear equations, which improve the convergence of some basic existing methods, are investigated. The Control of Accuracy and Debugging for Numerical Applications (CADNA) library are applied in [1]. By using this approach, the optimal number of iteration and the optimal solution with its accuracy are found. In this case, the usual stopping termination in the iterative procedure is replaced by a new criterion that is independent of the given tolerance such that the optimal results are evaluated computationally. In [2], Kwasinski and Chun discussed the application of classical mathematical root-finding optimization methods for maximum power point tracking (MPPT) of photovoltaic (PV) systems. Since in this study these methods are implemented digitally, practical issues encountered when digitally implementing a method originally based on a continuous domain are also explored. In particular, this study discusses potential errors inherently caused by digital processes not substantially explored in previous MPPT papers, such as algorithm numerical stability, quantization error, and discretization error analysis.

López-Ureña et. al [6], enhanced the computation of gradient retention times in liquid chromatography using root-finding methods. In this approach, the authors solved an integral equation (i.e., the fundamental equation of gradient elution), which has an analytical solution only for certain combinations of the retention model and gradient program. This limitation can be overcome by using numerical integration, which is a universal approach although at the cost of longer computation times. A simple algorithm is proposed by the author in [10], to construct Newton iteration formulae of any order commencing from the traditional linearly convergent fixed point iteration method and quadratically convergent Newton-Raphson method of frequently at the disposal of the scientific community. It is also shown that the well-known variants like Halley's method or Householder's methods of a high order can be reproduced from the general case outlined.

Most of the engineering and scientific problems are expressed as non-linear transcendental equations for which the evaluation of roots are more complicated. Such non-linear equations involve in various physical problems like van der waal equation, decay equation, charlesrichter magnitude of earthquake and surface-wave formula. In [7], Mahesh et. al proposed a quadratic convergent iterative method which reduced the error rapidly and very efficient for finding roots of non-linear equations. Noor [9] introduced a two-step iterative method for finding roots of non-linear equations, these methods perform better than one-step iterative methods including Newton method. He also suggested and analyzed a new family of iterative methods for solving non-linear equations using the system of coupled equations coupled with the decomposition technique [8].

2. Proposed method

The alternative iterative trigonometric equation is proposed as

$$x_{n+1} = x_n \left[1 + 2 \tan^{-1} \left(\frac{\left(\frac{-f(x_n)}{x_n f'(x_n)} \right)}{1 + \sqrt{1 - \left(\frac{-f(x_n)}{x_n f'(x_n)} \right)^2}} \right) \right], \quad n = 0, 1, 2, \dots \quad (2.1)$$

By extending the above iterative formula, as in the first two terms, one can obtain the standard Newton-Raphson method, and several methods are obtained based on series truncation. In reality:

Theorem 2.1. Suppose $\alpha \neq 0$ is a real exact root of the algebraic/transcendental equation $f(x)$ and h is a very small neighborhood of α . Let $f''(x)$ exists and $f'(x) \neq 0$ in h . Then the proposed method given in (2.1) produces a sequence of terms $\{x_n : n = 0, 1, 2, \dots\}$ with quadratically convergent.

Proof. The proposed method in (2.1) can be expressed in the form as

$$x_{n+1} = x_n \left[1 + 2 \left[\left(\frac{\left(\frac{-f(x_n)}{x_n f'(x_n)} \right)}{1 + \sqrt{1 - \left(\frac{-f(x_n)}{x_n f'(x_n)} \right)^2}} \right) - \frac{1}{3} \left(\frac{\left(\frac{-f(x_n)}{x_n f'(x_n)} \right)}{1 + \sqrt{1 - \left(\frac{-f(x_n)}{x_n f'(x_n)} \right)^2}} \right)^3 + \frac{1}{5} \left(\frac{\left(\frac{-f(x_n)}{x_n f'(x_n)} \right)}{1 + \sqrt{1 - \left(\frac{-f(x_n)}{x_n f'(x_n)} \right)^2}} \right)^5 - \dots \right] \right]. \quad (2.2)$$

Using standard expansion $\tan^{-1}(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \dots$, Neglecting the higher order terms, we get

$$\begin{aligned}
 x_{n+1} &= x_n \left[1 + 2 \left(\frac{\left(\frac{-f(x_n)}{x_n f'(x_n)} \right)}{1 + \sqrt{1 - \left(\frac{-f(x_n)}{x_n f'(x_n)} \right)^2}} \right) \right] \\
 &= x_n \left[1 + 2 \left(\frac{\left(\frac{-f(x_n)}{x_n f'(x_n)} \right)}{\frac{x_n f'(x_n) + \sqrt{(x_n f'(x_n))^2 - (f(x_n))^2}}{x_n f'(x_n)}} \right) \right] \\
 &= x_n \left[1 + 2 \left(\frac{-f(x_n)}{x_n f'(x_n) + \sqrt{(x_n f'(x_n))^2 - (f(x_n))^2}} \right) \right],
 \end{aligned} \tag{2.3}$$

as $\left(\frac{f(x_n)}{f'(x_n)} \right)^2 = h^2 \rightarrow 0$ since h is very small.

The above equation (2.3) reduces to

$$\begin{aligned}
 x_{n+1} &= x_n \left[1 - 2 \left(\frac{f(x_n)}{2x_n f'(x_n)} \right) \right] \\
 &= x_n - \frac{f(x_n)}{f'(x_n)}.
 \end{aligned}$$

This shows that the proposed method reduces to Newton-Raphson method and having quadratic convergence. □

2.1. Proposed algorithm

- Identify the initial approximations x_0 & x_1 such that $f(x_0) \times f(x_1) < 0$.
- Apply the proposed method to find the next approximation root of the given equation.
- Repeat the above step until we get the desired approximate root.

Flow chart of the proposed method is presented in Figure 1.

3. Numerical experiments

This section presents some numerical examples to explain the efficiency of the proposed method provided in the section above, and comparisons are taken into consideration to ensure that the proposed method is more efficient than other methods.

The following problems are considered to show the effectiveness of our proposed method. Here x_0 is considered as the initial approximation to the root.

Example 3.1. $f(x) = xe^{-x} - 0.1$ with $x_0 = 0.1$.

Example 3.2. $f(x) = 11x^{11} - 1$ with $x_0 = 1$.

Example 3.3. $f(x) = x - e^{\sin x} + 1$ with $x_0 = 2$.

Example 3.4. $f(x) = \ln x$ with $x_0 = 0.5$.

In the numerical experiments, the errors are taken as 10^{-15} , and the maximum number of iterations is limited to 100, the results of examples 1 to 4 are shown in Table 1.

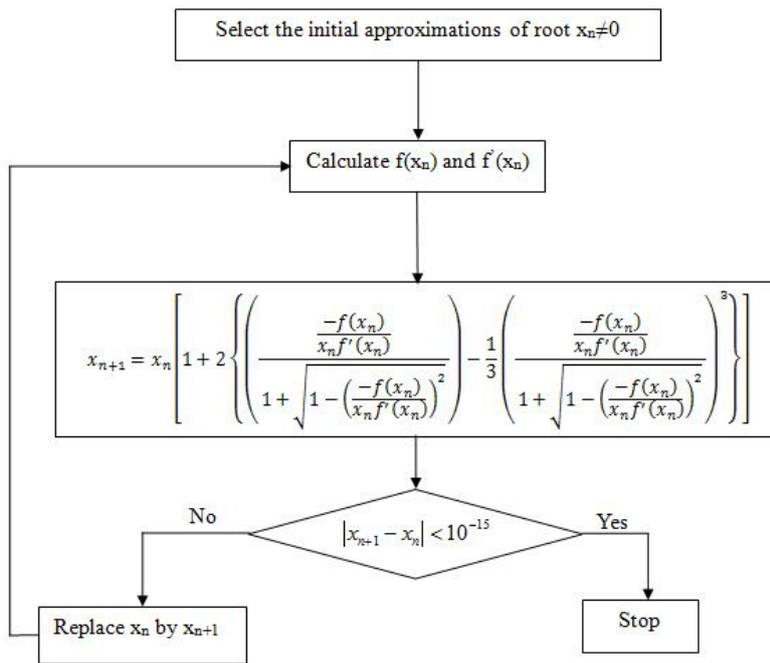


Figure 1: Flow chart of the proposed method.

Table 1: The numerical comparison of examples using various existing methods.

Ex.	Chen & Li [3]			Chen & Li [4]			Proposed method		
	n	x_n	$ f(x_n) $	n	x_n	$ f(x_n) $	n	x_n	$ f(x_n) $
1	6	1.11833e-01	0.00000e+00	6	1.11833e-01	0.00000e+00	3	1.11833e-01	0.0000e+00
2	7	8.04133e-01	1.22124e-15	8	8.04133e-01	4.44089e-16	7	8.04133e-01	4.44089e-16
3	5	1.69681e+00	4.44089e-16	11	1.69681e+00	4.44089e-16	5	1.69681e+00	4.44089e-16
4	6	1.00000e+00	0.00000e+00	7	1.00000e+00	0.00000e+00	5	1.00000e+00	0.0000e+00

Example 3.1. Consider a transcendental equation

$$f(x) = xe^{-x} - 0.1. \tag{3.1}$$

The following Table 2 shows the comparison between different existing methods and proposed method with initial approximations $x_0 = 0$ and $x_1 = 1$. Here n represents iteration numbers and x_n represents the corresponding approximation root.

Table 2: The numerical comparison of Example 3.1 using different existing methods.

Bisection		Regula-Falsi method		Proposed method	
n	x_n	n	x_n	n	x_n
1	0.5	1	0.271828182845905	1	0.11171239019496900
2	0.25000000000000000	2	0.13123614957214600	2	0.11183254383445500
3	0.12500000000000000	3	0.11402370160043800	3	0.11183255915896300
4	0.06250000000000000	4	0.11207786888180100	4	0.11183255915896300
⋮	⋮	⋮	⋮		
42	0.11183255915898400	17	0.11183255915896300		

Example 3.2. Consider a transcendental equation

$$f(x) = 11x^{11} - 1. \tag{3.2}$$

The following Table 3 shows the comparison between different existing methods and proposed method with initial approximations $x_0 = 0$ and $x_1 = 1$. Here n represents iteration numbers and x_n represents corresponding approximation root.

Table 3: The numerical comparison of Example 3.2 using different existing methods.

Bisection		Regula-Falsi method		Proposed method	
n	x_n	n	x_n	n	x_n
1	0.5	1	0.090909090909091	1	0.917261051121725
2	0.75	2	0.173553719005368	2	0.853423490462349
3	0.875	3	0.248685195862868	3	0.816152164916169
4	0.8125	4	0.316986388027222	4	0.804979772313121
⋮	⋮	⋮	⋮		
49	0.804133097503664	108	0.804133097503664	7	0.804133097503664

Example 3.3. Consider a transcendental equation

$$f(x) = x - e^{\sin x} + 1. \tag{3.3}$$

The following Table 4 shows the comparison between different existing methods and proposed method with initial approximations $x_0 = 1.5$ and $x_1 = 2$. Here n represents iteration numbers and x_n represents corresponding approximation root.

Table 4: The numerical comparison of Example 3.3 using different existing methods.

Bisection		Regula-Falsi method		Proposed method	
n	x_n	n	x_n	n	x_n
1	1.75	1	1.645067953924812	1	1.744811921632327
2	1.625	2	1.685074247441264	2	1.698840973066092
3	1.6875	3	1.694253896381327	3	1.696816413391276
4	1.71875	4	1.696259793878769	4	1.696812386825692
⋮	⋮	⋮	⋮	⋮	⋮
49	1.696812386809751	24	1.696812386809751	6	1.696812386809751

Example 3.4. Consider a transcendental equation

$$f(x) = \ln x. \tag{3.4}$$

The following Table 5 shows the comparison between different existing methods and proposed method with initial approximations $x_0 = 0.5$ and $x_1 = 1.2$. Here n represents iteration numbers and x_n represents corresponding approximation root.

Table 5: The numerical comparison of Example 3.4 using different existing methods.

Bisection		Regula-Falsi method		Proposed method	
n	x_n	n	x_n	n	x_n
1	0.85	1	1.000001949490732	1	0.874016607739766
2	1.025	2	1.000000543230269	2	0.992066210397659
3	0.9375	3	1.000000151372449	3	0.999968527492996
4	0.98125	4	1.000000042180307	4	0.99999999504741
⋮	⋮	⋮	⋮		
53	1.000000000000000	23	1.000000000000000	5	1.000000000000000

4. Conclusions

The proposed work proves the primacy for finding the approximate root of a given transcendental function is much better than previously existing methods such as Bisection, Regula-Falsi and secant methods. This act has been illustrated through standard numerical examples. The proposed method is based on the combination of inverse tan series and Newton-Raphson method. The rate of convergence of the proposed method is discussed and found to be quadratic. Performance of the proposed method is done through Matlab programming. On the whole, the proposed method executes much faster and more accurate convergence to the exact solution than the previously existing standard methods. This proposed algorithm can be applied in the discrete stochastic arithmetic (DSA) to validate the class of multi-step iterative methods and find the optimal numerical solution of nonlinear equations, maximum power point tracking (MPPT) of photovoltaic (PV) systems. This method is also applicable for finding potential errors inherently caused by digital processes such as algorithm numerical stability, quantization error and concretization error analysis.

References

- [1] M. A. F. Araghi, *A reliable algorithm to check the accuracy of iterative schemes for solving nonlinear equations: an application of the CESTAC method*, *SeMA J.*, **77** (2020), 275–289. 1
- [2] S. Chen, A. Kwasinski, *Analysis of classical root finding methods applied to digital maximum power point tracking for sustainable photovoltaic energy generation*, *IEEE Trans. Power Electron.*, **26** (2011), 3730–3743. 1
- [3] J. H. Chen, W. G. Li, *An exponential regula falsi method for solving nonlinear equations*, *Numer. Algorithms*, **41** (2006), 327–338. 1
- [4] J. H. Chen, W. G. 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. 1
- [5] T. Gemechu, *Root Finding With Some Engineering Applications*, *Int. J. Innov. Res. Sci. Eng. Tech.*, **3** (2016), 80–85. 1
- [6] S. López-Ureña, J. R. Torres-Lapasió, M. C. García-Alvarez-Coque, *Enhancement in the computation of gradient retention times in liquid chromatography using root-finding methods*, *J. Chromatogr. A*, **1600** (2019), 137–147. 1
- [7] G. Mahesh, G. Swapna, K. Venkateshwarlu, *An iterative method for solving non-linear transcendental equations*, *J. Math. Comput. Sci.*, **10** (2020), 1633–1642. 1
- [8] M. A. Noor, *New family of iterative methods for nonlinear equations*, *Appl. Math. Comput.*, **190** (2007), 553–558. 1
- [9] M. A. Noor, F. Ahmad, S. Javeed, *Two step iterative methods for nonlinear equations*, *Appl. Math. Comput.*, **181** (2006), 1068–1075. 1
- [10] M. Turkyilmazoglu, *A simple algorithm for high order Newton iteration formulae and some new variants*, *Hacet. J. Math. Stat.*, **49** (2020), 425–438. 1