

RESEARCH NOTE

Open Access



A derivative-free root-finding algorithm using exponential method and its implementation

Srinivasarao Thota¹, Mohamed M. Awad², P. Shanmugasundaram^{3*} and Laxmi Rathour⁴

Abstract

Objective In this paper, we develop a new root-finding algorithm to solve the given non-linear equations. The proposed root-finding algorithm is based on the exponential method. This algorithm is derivative-free and converges fast.

Results Several numerical examples are presented to illustrate and validation of the proposed methods. Microsoft Excel and Maple implementation of the proposed algorithm is presented with sample computations.

Keywords Iterative algorithms, Nonlinear equations, Exponential method, Algorithm implementation, Root-finding algorithms

Mathematics Subject Classification 65H04, 65Hxx

Introduction

Finding an approximate root of non-linear equations using iterative algorithms plays a significant role in the computational and applied mathematics. The applications of non-linear equations of the type $f(x) = 0$ arise in various branches of scientific computing fields. Solving such non-linear equations is one of the most important problems and frequently appearing in different scientific fields that can be modeled through nonlinear equations. In recent time, several researchers, engineers and scientists focused on solving non-linear equations numerically

as well as analytically. Iterative algorithms play a vital role in finding the solution of such non-linear problems. In general, the roots of non-linear or transcendental equations cannot be expressed in closed form or cannot be computed analytically. The root-finding algorithms provide us to compute approximations to the roots; these approximations are expressed either as small isolating intervals or as floating point numbers. There are various numerical algorithm/methods available in the literature, see for example [1–20], for more details.

Many new modified/hybrid/multi-step iterative algorithms are developed in the last few years, by employing various mathematical algorithms/techniques. Noor et al. discussed the fifth-order second derivative-free algorithm in 2007, see [21], by using the finite difference scheme. Grau-Sanchez et al. presented a fifth-order Chebyshev-Halley type method in 2008, see [22]. Zhanlav et al. proposed a three-step fifth-order iterative algorithm in 2010 [23]. Nazeer et al. introduced a novel second derivative-free Householder's method having fifth-order convergence by using finite-difference scheme [24] in 2016. Recently, in 2021, Amir et al. developed an

*Correspondence:

P. Shanmugasundaram
psserode@mtu.edu.et

¹ Department of Mathematics, Amrita School of Physical Sciences, Amrita Vishwa Vidyapeetham, Amaravati, Andhra Pradesh 522503, India

² Mechanical Power Engineering Department, Faculty of Engineering, Mansoura University, Mansoura 35516, Egypt

³ Department of Mathematics, College of Natural & Computational Sciences, Mizan-Tepi University, Mizan Teferi, Ethiopia

⁴ Department of Mathematics, National Institute of Technology, Chaltlang, 796012 Aizawl, Mizoram, India



© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

efficient and derivative-free algorithm for determining an approximate solution of the given non-linear scalar equations by applying forward- and finite-difference schemes similar to Traub’s method, see [25]. In this paper, we propose a new root- finding algorithm, which is derivative-free, using exponential method. To propose the algorithm with derivative-free, we employ the forward difference scheme and finite difference scheme. This gives computationally low cost. Microsoft Excel and Maple implementations of the proposed algorithm are presented. Maple and Excel implementations with sample computations for differential and transcendental equations are available in the literature, see for example [18, 19, 26, 27] and there are various techniques for different type of applications, see [20, 28–35], the references cited therein.

Preliminaries

In this paper, we consider the non-linear equation of the type

$$f(x) = 0. \tag{1}$$

Iterations techniques are a common approach widely used in various numerical algorithms/methods. It is a hope that an iteration in the general form of $x_{n+1}=g(x_n)$ will eventually converge to the true solution α of the problem (1) at the limit when $n \rightarrow \infty$. The concern is whether this iteration will converge, and, if so, the rate of convergence. Specifically we use the following expression to represent how quickly the error $e_n = \alpha - x_n$ converges to zero. Let $e_n = \alpha - x_n$ and $e_{n+1} = \alpha - x_{n+1}$ for $n \geq 0$ be the errors at n -th and $(n + 1)$ -th iterations respectively. If two positive constants μ and p exist, and

$$\lim_{n \rightarrow \infty} \frac{|e_{n+1}|}{|e_n|^p} = \frac{|\alpha - x_{n+1}|}{|\alpha - x_n|^p} = \mu, \tag{2}$$

then the sequence is said to converge to α . Here $p \geq 1$ is called the *order of convergence*; the constant μ is the *rate of convergence* or *asymptotic error constant*. This expression may be better understood when it is interpreted as $|e_{n+1}| = \mu |e_n|^p$ when $n \rightarrow \infty$. Obviously, the larger p and the smaller μ , the more quickly the sequence converges.

Theorem 1 [16, 36] *Suppose that $g \in C^p[a, b]$. If $g^{(k)}(x) = 0$, for $k = 0, 1, 2, \dots, p - 1$ and $g^{(p)}(x) \neq 0$, then the sequence $\{x_n\}$ is of order p .*

This paper focuses on developing iterative algorithm having fourth-order of convergence. The following section presents the proposed algorithm using Newton–Raphson method and exponential method without computing the derivative.

Main text (a new iterative algorithm)

We assume that α is an exact root of the Eq. (1) and let a be an initial approximation (sufficiently close) to α . In the exponential method, we can find first approximation root using the following formula. See [5] for more details.

$$x = a \exp \left(\frac{-f(a)}{af'(a)} \right).$$

If x_{n+1} is the required root, then the exponential formula can be expressed as, for $n = 0, 1, 2, 3, \dots$,

$$x_{n+1} = x_n \exp \left(\frac{-f(x_n)}{x_n f'(x_n)} \right). \tag{3}$$

which has more than second-order convergence.

Suppose $y_n = x_{n+1}$, where x_{n+1} is the Newton–Raphson formula, is predictor and corrector, then Traub [37] created a new two-step iterative algorithm as follows, $n = 0, 1, 2, 3, \dots$,

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

It is shown in [37] that the Traub’s method has fourth-order convergence. Since Newton–Raphson formula repeated twice, the Traub’s method includes four computations to execute the algorithm. Amir et al. extended the Traub’s method to derivative-free algorithm by applying forward- and finite-difference schemes on Traub’s method.

In this paper, we propose a new two-step iterative algorithm similar to that of Amir et al., and the proposed algorithm has more than fourth-order convergence. The proposed method is created using the exponential method designed by Thota et al. [5]. Using exponential method, one can obtain an approximate root of a given non-linear equation using the formula (3). The order of convergence of the exponentiation method is more than two, see [5] for more details. Using exponential method (3), the proposed algorithm consists of the following steps:

$$\begin{aligned} y_n &= x_n \exp \left(\frac{-f(x_n)}{x_n f'(x_n)} \right), \\ x_{n+1} &= y_n \exp \left(\frac{-f(y_n)}{y_n f'(y_n)} \right). \end{aligned} \tag{4}$$

One can observe that, this is a two-step iteration method to calculate roots of a given non-linear equations. Since there are two steps in the algorithm and it required four evaluations for its execution. The biggest disadvantage of the algorithm (4) is computational cost of each iteration which is more. In order to reduce the high computational cost, we replace the first derivative

by approximation and this suggests a novel derivative-free algorithm. Hence, it can be applied easily to the given non-linear equations where the first derivative is not defined in the domain. We use the forward difference approximation in the predictor to approximate the first derivative as follows, here $f(x_n) \geq 0$,

$$f'(x_n) = \frac{f(x_n + f(x_n)) - f(x_n)}{f(x_n)} = g(x_n). \tag{5}$$

Now, we use finite difference approximation in the corrector step (i.e., in step 2) as follows

$$f'(y_n) = \frac{f(y_n) - f(x_n)}{y_n - x_n} = h(x_n, y_n). \tag{6}$$

Substituting the Eqs. (5)–(6) in algorithm (4), we obtain a new efficient and derivative-free iterative algorithm to calculate the approximate solution of a given non-linear equation as follows

$$\begin{aligned} y_n &= x_n \exp\left(\frac{-f(x_n)}{x_n g(x_n)}\right), \\ x_{n+1} &= y_n \exp\left(\frac{-f(y_n)}{y_n h(x_n, y_n)}\right), \end{aligned} \tag{7}$$

where $g(x_n)$ and $h(x_n, y_n)$ are as given (5)–(6). This is a new iterative algorithm to find a root of transcendental equations in two-step without involvement of any derivative. One of the advantages of the proposed algorithm is existence of root where the first derivative does not exist at some particular points in the domain, and another big advance is the computational complexity. This method has more than fourth order convergence and its convergence analysis is presented in the following section.

Analysis of convergence

In this section, we show in the following theorem that the order of converges of the proposed algorithm is five. Let $I \subset \mathbb{R}$ be an open interval. To prove this, we follow the proofs of ([2], Theorem 5, Theorem 6) or ([16], Theorem 2, Theorem 3, Theorem 4).

Theorem 2 *Let $f: I \rightarrow \mathbb{R}$. Suppose $\alpha \in I$ is a simple root of (1) and θ is a sufficiently small neighborhood of α . Then the iterative formula (7) produces a sequence of iterations $\{x_n; n = 1, 2, \dots\}$ with order of convergence four.*

Proof Let.

$$y = x \exp\left(\frac{-f(x)}{xg}\right), \text{ and } R(x) = y \exp\left(\frac{-f(y)}{yh}\right),$$

where

$$g = \frac{f(x + f(x)) - f(x)}{f(x)}, h = \frac{f(y) - f(x)}{y - x}.$$

Since α is a root of $f(x)$, hence $f(\alpha) = 0$. One can compute that

$$\begin{aligned} R(\alpha) &= \alpha, \\ R'(\alpha) &= 0, \\ R''(\alpha) &= 0, \\ R'''(\alpha) &= 0, \\ R^{iv}(\alpha) &\neq 0. \end{aligned}$$

Hence the Algorithm (7) has fourth-order convergence, by Theorem 1. \square

One can also verify that the order of convergence of the proposed algorithm as in the following example.

Example 1 Consider the following equation.

$$f(x) = x^2 - 1. \tag{8}$$

It has a root $\alpha = -1$. We show, as discussed in proof of Theorem 2, that the proposed algorithm has fourth-order convergence. Following Theorem 2, we have

$$\begin{aligned} g &= \frac{f(x + f(x)) - f(x)}{f(x)} = x^2 - 2x - 1, \\ y &= x \exp\left(\frac{-f(x)}{xg}\right) = xe^{\frac{(x-1)(x+1)}{x(x^2-2x-1)}}, \\ h &= \frac{f(y) - f(x)}{y - x} = x\left(e^{-\frac{(x-1)(x+1)}{x(x^2-2x-1)}} + 1\right), \\ R(x) &= y \exp\left(\frac{-f(y)}{yh}\right) = xe^{\frac{-x^4 e^t - 3x^3 e^t + x^2 e^{-t} + x^2 e^t - x^3 + 2xe^{-t} + xe^t - e^{-t} + x}{x^2(x^2-2x-1)(e^t+1)}}, \end{aligned}$$

where

$$t = -\frac{(x - 1)(x + 1)}{x(x^2 - 2x - 1)}.$$

Now

$$\begin{aligned} R(\alpha) &= -1 = \alpha, \\ R'(\alpha) &= 0, \\ R''(\alpha) &= 0, \\ R'''(\alpha) &= 0, \\ R^{(iv)}(\alpha) &= 8 \neq 0. \end{aligned}$$

Hence, by Theorem 2, the algorithm in (7) has fourth-order convergence.

Numerical examples

Example 2 Consider a transcendental equation $e^x + \cos(x) - 1 = 0$ with $x_0 = -2$. Now we can compute a real of the given equation using the proposed algorithm (7) as follows.

Suppose $f(x) = e^x + \cos(x) - 1$, then we have

$$g(x_0) = \frac{f(x_0+f(x_0))-f(x_0)}{f(x_0)} = 0.5246013002,$$

$$y_0 = x_0 \exp\left(\frac{-f(x_0)}{x_0g(x_0)}\right) = -0.5900190724,$$

$$h(x_0, y_0) = \frac{f(y_0)-f(x_0)}{y_0-x_0} = 1.181617637 \quad \text{and,}$$

$$x_1 = y_0 \exp\left(\frac{-f(y_0)}{y_0h(x_0,y_0)}\right) = -1.025295284.$$

Similarly, we have the values in iteration 2:

$$g(x_1) = \frac{f(x_1+f(x_1))-f(x_1)}{f(x_1)} = 1.222059474,$$

$$y_1 = x_1 \exp\left(\frac{-f(x_1)}{x_1g(x_1)}\right) = -0.9298264088,$$

$$h(x_1, y_1) = \frac{f(y_1)-f(x_1)}{y_1-x_1} = 1.205191949,$$

$$x_2 = y_1 \exp\left(\frac{-f(y_1)}{y_1h(x_1,y_1)}\right) = -0.9237026911.$$

Iteration 3:

$$g(x_2) = \frac{f(x_2+f(x_2))-f(x_2)}{f(x_2)} = 1.194895070,$$

$$y_2 = x_2 \exp\left(\frac{-f(x_2)}{x_2g(x_2)}\right) = -0.9236326626,$$

$$h(x_2, y_2) = \frac{f(y_2)-f(x_2)}{y_2-x_2} = 1.194879228,$$

$$x_3 = y_2 \exp\left(\frac{-f(y_2)}{y_2h(x_2,y_2)}\right) = -0.9236326590.$$

One can obtain the function value at $x_3 = -0.9236326590$ as $f(-0.9236326590) = -5.3608 \times 10^{-11}$. Hence the required root $x = -0.9236326590$ is obtained in 3 iterations using the proposed algorithm.

Example 3 Consider a polynomial equation to find a real root.

$$0.986x^3 - 5.18x^2 + 9.064x - 5.287 = 0 \quad (9)$$

with $x_0=0.6$. Following Example 2 using the proposed algorithm (7), we have Iteration 1:

$$g(x_0) = 11.24874333,$$

$$y_0 = 0.749437179,$$

$$h(x_0, y_0) = 3.427685909,$$

$$x_1 = 1.101280164383, \quad f(x_1) = -0.270349537.$$

Other iterations values are

$$x_2 = 1.387799514358, \quad f(x_2) = -0.048898877,$$

$$x_3 = 1.568877491071, \quad f(x_3) = -0.00884392,$$

$$x_4 = 1.753077607303, \quad f(x_4) = -0.004242231,$$

$$x_5 = 1.883259728433, \quad f(x_5) = -0.00298145,$$

$$x_6 = 1.922476516171, \quad f(x_6) = -0.000608725,$$

$$x_7 = 1.929827783304, \quad f(x_7) = -1.59531E - 06,$$

$$x_8 = 1.929846242848, \quad f(x_8) = -2.30926E - 14.$$

Hence, the required approximate root of the given equation (9) is $x = 1.929846242848$.

Implementation of the proposed algorithm

Implementation in MS Excel

The proposed method can be computed in Excel easily as follows. The number of iterations n, initial guess $x_n, f(x_n), g(x_n), y_n, f(y_n), h(x_n, y_n), x_{n+1}$ and $f(x_{n+1})$ are placed in Excel cells, for example, in A5, B5, C5, D5, E5, F5, G5, H5 and I5 respectively. Enter the respective values in 6th row, i.e., $n = 0, x_0, "=f(B6)", "(f(B6+C6)-C6)/C6", "=B6*EXP((-C6)/(B6*D6)", "=f(E6)", "(F6-C6)/(E6-B6)", "=E6*EXP((-F6)/(E6*G6))"$ and $"=f(H6)"$ respectively in A6–I6. For second iteration, we need to replace x_n by x_{n+1} in B7 using the command $"=H6"$. The last columns, C6–I6, are drag down to get next iteration value. Finally, drag down the entire 8th row, A7–I7, until the required number of iterations, see Fig. 1. Sample computations using MS Excel are presented in the following section.

Example 4 Consider the Eq. (9) presented in Example 3 for sample computations using MS Excel.

$$f(x) = 0.986x^3 - 5.181x^2 + 9.067x - 5.289$$

with $x_0=0.6$. Following the procedure in Section, we have the results as in Fig. 1.

	A	B	C	D	E	F	G	H	I
1	$f(x) = 0.986x^3 - 5.18x^2 + 9.064x - 5.287, \quad x_0 = 0.6$								
2									
3									
4									
5	<i>n</i>	<i>x_n</i>	<i>f(x_n)</i>	<i>g(x_n)</i>	<i>y_n</i>	<i>f(y_n)</i>	<i>h(x_n,y_n)</i>	<i>x_{n+1}</i>	<i>f(x_{n+1})</i>
6	0	0.6	-1.500984	11.24874333	0.749437179	-0.988760286	3.427685909	1.101280164383	-0.270349537
7	1	1.10128016	-0.27034954	1.835110097	1.25890895	-0.118337969	0.964364262	1.387799514358	-0.048898877
8	2	1.38779951	-0.04889888	0.438659824	1.503872154	-0.017316046	0.272095391	1.568877491071	-0.00884392
9	3	1.56887749	-0.00884392	0.095898506	1.663863536	-0.004225489	0.048622206	1.753077607303	-0.004242231
10	4	1.75307761	-0.00424223	-0.007626466	1.276437611	-0.106331922	0.214186163	1.883259728433	-0.00298145
11	5	1.88325973	-0.00298145	0.042551232	1.954646796	0.002483039	0.076547328	1.922476516171	-0.000608725
12	6	1.92247652	-0.00060873	0.078510277	1.930245617	3.46028E-05	0.082805971	1.929827783304	-1.59531E-06
13	7	1.92982778	-1.5953E-06	0.086411605	1.929846245	2.02968E-10	0.086422185	1.929846242848	-2.30926E-14
14	8	1.92984624	-2.3093E-14	0	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!

Fig. 1 Proposed algorithm in Excel

Implementation in Maple

```

EXPEXP := proc (a, Eq, eps, n)
    local a1, f, xn, g, y, h, i;

    i := 0;
    a1 := evalf(a);
    f := unapply(lhs(Eq), x);
    do
        g := (f(a1+f(a1))-f(a1))/f(a1);
        y := a1*exp(-f(a1)/(a1*g));
        h := (f(y)-f(a1))/(y-a1);
        xn := y*exp(-f(y)/(y*h));

        i := i+1;
        if f(xn) = 0 or f(y) = 0 or xn-a1 < eps or i = n then
            return xn
        else a1 := xn
        end if;
        printf("Iteration %g : x = %g \n", i, xn)
    end do
end proc

```

Example 5 Consider the equation $e^x + \cos(x) - 1 = 0$ given in Example 2 for sample computations in Maple.

$$> \text{EXPEXP}(-2, \exp(x) + \cos(x) - 1 = 0, 10.0^{\text{th}}(-10), 5)$$

Iteration 1 : x = -1.0253

Iteration 2 : x = -0.923703

-0.9236326590

Conclusion

In this paper, we proposed a new root-finding algorithm to solve the nonlinear equations. The main idea of this algorithm is based on the exponential method. The proposed algorithm doesn't have any derivative even though exponential method is involved, and moreover it converges fast. Numerical examples are presented to illustrate and validation of the proposed methods. Implementation of the proposed algorithm in Excel and Maple is discussed with sample computations.

Limitations

In this paper, we focused on MS Excel and Maple implementation. However, the proposed algorithms can be implemented in many mathematical software tools such as Mathematica, SCILab, Matlab, etc.

Abbreviation

MS Excel Microsoft Excel

Acknowledgements

The authors are thankful to the reviewers and editor for providing valuable inputs to improve the quality and present format of manuscript.

Author contributions

Dr. ST involved in creation of the proposed algorithms and Maple implementation. Dr. MMA is involved in suggestion and verification of the numerical examples in the present paper. Dr. PS is involved in modification and improvement of the numerical examples. Ms. LR is involved in verification of the numerical examples. All authors read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

The datasets generated and analyzed during the current study are presented in this manuscript.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Received: 20 October 2022 Accepted: 4 October 2023

Published online: 17 October 2023

References

- Kincaid DE, Cheney EW. Numerical analysis. Pacific Grove: Brooks/Cole; 1990.
- Saqib M, Iqbal M, Ali S, Ismaeel T. New fourth and fifth-order iterative methods for solving nonlinear equations. Appl Math. 2015;6:1220–7.
- Waals VD, Diderik J. Over de continuïteit van den gas-en vloeistoestand (on the continuity of the gas and liquid state, Ph.D. dissertation, Leiden Univ., Leiden, The Netherlands; 1873.
- Hussain S, Srivastav VK, Thota S. Assessment of interpolation methods for solving the real life problem. Int J Math Sci Appl. 2015;5(1):91–5.
- Thota S, Gemechu T, Shanmugasundaram P. New algorithms for computing non-linear equations using exponential series. Palestine J Math. 2021;10(1):128–34.
- Thota S, Srivastav VK. Quadratically convergent algorithm for computing real root of non-linear transcendental equations. BMC Res Notes. 2018;11:909.
- Thota S, Srivastav VK. Interpolation based hybrid algorithm for computing real root of non-linear transcendental functions. Int J Math Comput Res. 2014;2(11):729–35.
- Thota S. A new root-finding algorithm using exponential series. Ural Math J. 2019;5(1):83–90.
- Gemechu T, Thota S. On new root finding algorithms for solving nonlinear transcendental equations. Int J Chem Math Phys. 2020;4(2):18–24.
- Parveen T, Singh S, Thota S, Srivastav VK. A new hybrid root-finding algorithm for transcendental equations using bisection, regula-Falsi and Newton-Raphson methods. In: National conference on sustainable & recent innovation in science and engineering (SUNRISE-19); 2019. ISBN No. 978-93-5391-715-9.
- Srivastav VK, Thota S, Kumar M. A new trigonometrical algorithm for computing real root of non-linear transcendental equations. Int J Appl Comput Math. 2019;5:44.
- Thota S. A new hybrid halley-false position type root finding algorithm to solve transcendental equations. In: Istanbul international modern scientific research congress—III, 06–08 May 2022, Istanbul Gedik University, Istanbul, Turkey.
- Thota S. A new three-step root-finding algorithm and its implementation in excel. In: International conference on evolution in pure and applied mathematics (ICEPAM-2022), 16–18 November; 2022, Department of Mathematics, Akal University, Talwandi Sabo, Punjab, India.
- Thota S, Kalyani P, Neeraj G, Balarama Krishna C. A new hybrid root-finding algorithm for solving transcendental equations using exponential and regula-falsi method. In: Virtual international conference on computational intelligence, simulation, financial engineering and mathematical modelling for industry and commerce, 30–31 August 2022, Great Zimbabwe University, Masvingo Zimbabwe.
- Thota S. Solution of generalized Abel's integral equations by homotopy perturbation method with adaptation in laplace transformation. Sohag J Math. 2022;9(2):29–35.
- Thota S, Shanmugasundaram P. On new sixth and seventh order iterative methods for solving non-linear equations using homotopy perturbation technique. BMC Res Notes. 2022;15:267. <https://doi.org/10.1186/s13104-022-06154-5>.
- Thota S, Maghrabi L, Shanmugasundaram P, Kanan M, Al-Sherideh AS. A new hybrid root-finding algorithm to solve transcendental equations using arcsine function. Inf Sci Lett. 2023;12(6):2533–7. <https://doi.org/10.18576/isl/120650>.

18. Thota S, Ayoade AA. On Solving Transcendental Equations Using Various Root Finding Algorithms with Microsoft Excel. 1st ed. Notion Press; 2022. (ISBN-13: 979-8886844238).
19. Thota S. Microsoft excel implementation of numerical algorithms for nonlinear algebraic or transcendental equations. In: 5th international conference on statistics, mathematical modelling and analysis (SMMA 2022), May 27–29; 2022 in Xi'an, China.
20. Thota S, Gemechu T, Ayoade AA. On new hybrid root-finding algorithms for solving transcendental equations using exponential and Halley's methods. *Ural Math J.* 2023;9(1):176–86. <https://doi.org/10.15826/umj.2023.1.016>.
21. Noor MA, Khan WA, Hussain A. A new modified Halley method without second derivatives for nonlinear equation. *Appl Math Comput.* 2007;189(2):1268–73.
22. Grau-Sanchez M, Gutierrez JM. Some variants of the Chebyshev-Halley family of methods with fifth order of convergence. *Int J Comput Math.* 2010;87(4):818–33.
23. Zhanlav T, Chuluunbaatar O, Ankhbayar G. On newton-type methods with fourth and fifth-order convergence. *Discrete Contin Models Appl Comput Sci.* 2010;2(2):30–5.
24. Nazeer W, Tanveer M, Kang SM, Naseem A. A new Householder's method free from second derivatives for solving nonlinear equations and polynomiography. *J Nonlinear Sci Appl.* 2016;9:998–1007.
25. Naseem A, Rehman MA, Abdeljawad T. Real-world applications of a newly designed root-finding algorithm and its polynomiography. *IEEE Access.* 2021;9:160868–77.
26. Thota S. A symbolic algorithm for polynomial interpolation with Stieltjes conditions in maple. *Proc Inst Appl Math.* 2019;8(2):112–20.
27. Thota S. On a third order iterative algorithm for solving non-linear equations with maple implementation. In: National E-conference on interdisciplinary research in science and technology, May 30–31; 2020, Amiruddaula Islmia Degree College, Locknow, India.
28. Rajawat RS, Singh KK, Mishra VN. Approximation by modified Bernstein polynomials based on real parameters. *Math Found Comput.* 2023. <https://doi.org/10.3934/mfc.2023005>.
29. Pandey S, Rajawat RS, Mishra VN. Approximation properties of modified Jain-Gamma operators preserving linear function. *Palestine J Math.* 2023;12(2):169–82.
30. Raiz M, Kumar A, Mishra VN, Rao N. Dunkl analogue of Szasz Schurer beta operators and their approximation behavior. *Math Found Comput.* 2022;5(4):315–30. <https://doi.org/10.3934/mfc.2022007>.
31. Ayoade AA, Thota S. Functional education as a nexus between agricultural and industrial revolution: an epidemiological modelling approach. *Uniciencia.* 2023;37(1):1–16. <https://doi.org/10.15359/ru.37-1.12>.
32. Dubey R, Mishra LN, Ali R. Special class of second-order nondifferentiable symmetric duality problem with (G, α_r) -pseudobonvexity assumptions. *Mathematics.* 2019;7(8):763–78. <https://doi.org/10.3390/math7080763>.
33. Auwalu A, Hincal E, Mishra LN. On some fixed point theorems for expansive mappings in dislocated cone metric spaces with banach algebras. *J Math Appl.* 2019;42:21–33.
34. Mishra LN, Raiz M, Rathour L, Mishra VN. Tauberian theorems for weighted means of double sequences in intuitionistic fuzzy normed spaces. *Yugoslav J Oper Res.* 2022;32(3):377–88. <https://doi.org/10.2298/YJOR210915005M>.
35. Sharma MK, Dhiman N, Kumar S, Rathour L, Mishra VN. Neutrosophic Monte Carlo simulation approach for decision making in medical diagnostic process under uncertain environment. *Int J Neutrosophic Sci.* 2023;22(1):08–16. <https://doi.org/10.54216/IJNS.220101>.
36. Babolian E, Biazar J. On the order of convergence of adomain method. *Appl Math Comput.* 2002;130:383–7.
37. Traub JF. Iterative methods for the solution of equations. New York: Chelsea; 1982.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Prof. Srinivasarao Thota completed his M.Sc. in Mathematics from Indian Institute of Technology (IIT) Madras, India and Ph.D. in Mathematics from Motilal Nehru National Institute of Technology

(NIT) Allahabad, India. Prof. Thota's area of research interests are Computer Algebra (symbolic methods for differential equations), Numerical Analysis (root finding algorithms), Mathematical Modeling (ecology). He has published more than 40 research papers in various international journals and presented his research work at several international conferences as oral presenter and invited/keynote/guest speaker in different countries. Presently working at the Department of Mathematics, Amrita Vishwa Vidyapeetham, Amaravati, Andhra Pradesh, India.

Prof. Mohamed M. Awad is working at Faculty of Engineering, Mansoura University, Mansoura, Egypt. He received his PhD from Memorial University of Newfoundland in 2007. His research is focused on the modeling of complex fluid dynamics and heat transfer problems in internal flows. He is the author of 3 book chapters. He is a member of ASME. He was a recipient of the ASME IPTI Award in 2005 and 2006. He won silver medal at 45th International Exhibition of Inventions, Geneva, Swiss, 29 March–2 April 2017.

Prof. P. Shanmugasundaram completed M.Sc. in Mathematics from Sri Vasavi College, M.Phil in Mathematics from Madurai Kamraj University, Madurai, India and Ph.D. in Mathematics Anna University, Chennai, India. He has published more than 25 research papers in various international journals. Presently, working at Department of Mathematics, College of Natural & Computational sciences, Mizan?Tepi University, Ethiopia.

Ms. Laxmi Rathour born in Anuppur, India. She received her Master's degree from Department of Mathematics, Indira Gandhi National Tribal University, Amarkantak, M.P., India. She has research interests in the areas of pure and applied mathematics specially Approximation theory, Nonlinear analysis and optimization, Fixed Point Theory and applications, Operation Research etc. In the meantime, she has published several scientific and professional papers in the country and abroad. Moreover, she serves voluntary as reviewer for Mathematical Reviews (USA) and Zentralblatt Math (Germany).

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

