ระเบียบวิธีการทำซ้ำอันดับเจ็ดสำหรับการแก้สมการไม่เชิงเส้น

The Seventh-Order Iterative Methods for Solving Nonlinear Equations

ไพรัชช์ จันทร์งาม[1], วรัชยา ทองแสน[1] และ เฉลิมวุฒิ คำเมือง[2*]
Pairat Janngam [1], Waratchaya Tongsan [1] and Chalermwut Comemuang [2*]
[1]สาขาวิชาคณิตศาสตร์ คณะครุศาสตร์ มหาวิทยาลัยราชภัฏบุรีรัมย์ ประเทศไทย
[2]สาขาวิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏบุรีรัมย์ ประเทศไทย
[1]Department of Mathematics, Faculty of Education, Buriram Rajabhat University, Thailand
[2]Department of Mathematics, Faculty of Science, Buriram Rajabhat University, Thailand

Received : 5 March 2023
Revised : 22 August 2023
Accepted : 11 September 2023

บทคัดย่อ

บทความนี้ได้นำเสนอระเบียบวิธีใหม่ที่มี 3 ขั้นตอน ซึ่งสองขั้นตอนแรกใช้หลักการระเบียบวิธี Shengfeng อันดับที่สี่ ทำการวิเคราะห์การลู่เข้าของระเบียบวิธีใหม่เป็นอันดับที่ 7 ได้นำเสนอตัวอย่างของระเบียบวิธีการใหม่นี้กับระเบียบวิธีที่มีอันดับเจ็ดรูปแบบอื่น

คำสำคัญ : สมการไม่เชิงเส้น ; ระเบียบวิธีการทำซ้ำ ; วิธีนิวตัน-ราฟสัน ; อันดับการลู่เข้า

Abstract

The study presents a new scheme of three steps, of which the first two steps are based on the fourth order Shengfeng method. The proposed method has order seven. Numerical tests show that the new methods are comparable with the well-known existing methods and give better results.

Keywords : Nonlinear equations ; Iterative method ; Newton-Raphson method ; order of convergence

*Corresponding author. E-mail : chalermwut.cm@bru.ac.th

Introduction

The most significant issue in numerical analysis is solving a root-finding problem of nonlinear equations. Among the relevant factors of the iterative method are convergence and computation efficiency, where the convergent sequence evinces the relative speed of convergence. We will consider these two relevant factors and construct an efficient iterative method to find a simple root for nonlinear equations. The classical iterative method for solving a nonlinear equation is Newton's method (Abbasbandy,2003), (Cordero & Torregrosa, 2007), (Muhaijir, Imran & Gamal, 2016).

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \; n = 0,1,2,\ldots \tag{1}$$

This is an important and basic method for finding the approximate root of which has quadratic convergence.

Many researchers have developed existing methods by increasing the order of convergence or decreasing the function estimation in each iteration. Numerous papers discuss nonlinear algebraic equations, including the seventh-order iterative methods developed by using weight function methods and using an approximation for the last derivative (Fardi, Ghasemi, & Davari, 2012), a new family of iterative methods for nonlinear equations (Noor, 2007), iterative methods for solving scalar equations (Shin, Faisal, & Arif, 2016) and other methods (Kong-ied, 2021; Chun, 2007; Chun, 2007; Chun, 2008; Cordero & Torregrosa, 2007; Edwar, Imran, & Deswita, 2016; Weerakoon & Fernando, 2000). Shengfeng (2019) presented a fourth-order iterative method without calculating the higher derivatives for nonlinear equations (Shengfeng & Dong, 2019). Among these methods, we emphasize developing the concept of Shengfeng to construct an iterative method.

In this paper, we present a new method, for Algorithm 2.1, combining the Shengfeng method using linear interpolation. In the next section, we conduct discussions on the proposed iterative methods. Then, we prove that the new methods are seventh-order convergence, which used the Taylor polynomial to prove convergence analysis. Finally, numerical examples are given to show the performance of the new methods.

Methods

In the following section, we suggest a three-step iterative method for resolving the nonlinear equation derived from fourth-order Newton-type composition by Shengfeng(2019). This method is given as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ x_{n+1} &= x_n - \frac{\big(f(x_n) - f(y_n)\big) f(x_n)}{\big(f(x_n) - 2f(y_n)\big) f'(x_n)} \end{aligned} \tag{2}$$

By including a Newton-like step, the fourth-order method (2) can be extended to create a seventh-order iterative method as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ z_n &= x_n - \frac{\big(f(x_n) - f(y_n)\big) f(x_n)}{\big(f(x_n) - 2f(y_n)\big) f'(x_n)} \\ x_{n+1} &= z_n - \frac{f(z_n)}{f'(z_n)} \end{aligned} \tag{3}$$

Using the following formula for linear interpolation on points $\big(x_n, f'(x_n)\big)$ and $\big(y_n, f'(y_n)\big)$ for approximating $f'(z_n)$ as follows:

$$f'(z_n) \approx \frac{z_n - x_n}{y_n - x_n} f'(y_n) + \frac{z_n - y_n}{x_n - y_n} f'(x_n), \text{ when } y_n = x_n - \frac{f(x_n)}{f'(x_n)} \tag{4}$$

This simplification gives:

$$f'(z_n) \approx \frac{f'(x_n)f(y_n) - f'(y_n)f(x_n) + f'(y_n)f(y_n)}{2f(y_n) - f(x_n)} \tag{5}$$

Substituting (5) in (3), the new three-step seventh-order method is given as:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ z_n &= x_n - \frac{\big(f(x_n) - f(y_n)\big) f(x_n)}{\big(f(x_n) - 2f(y_n)\big) f'(x_n)} \\ x_{n+1} &= z_n - \frac{f(z_n)\big(2f(y_n) - f(x_n)\big)}{f'(x_n)f(y_n) - f'(y_n)f(x_n) + f'(y_n)f(y_n)} \end{aligned} \tag{6}$$

Algorithm 2.1 For a given x_0, the iteration is generated by the following iterative scheme.

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

The discussion of the suggested iteration schemes' convergence criteria is included in this section.

Theorem 3.1 Let $f : I \subset R \to R$ have a simple root x^* in I , for an open interval I. If $f(x)$ to the root x^* to be sufficiently smooth, then the order of convergence given by algorithm 2.1 is seven.

Proof. Assume x^* is a solution of the equation $f(x)$, so $f(x^*) = 0$. Let e_k be the error at k-th iteration, then one has $e_k = x_k - x^*$. Using Taylor polynomial for $f(x)$ expanded about x^*, we obtain:

$$\begin{aligned} f(x_k) \;&=\; f'(x^*)e_k + \frac{1}{2!}f''(x^*)e_k^2 + \frac{1}{3!}f'''(x^*)e_k^3 + \frac{1}{4!}f^{(4)}(x^*)e_k^4 + \frac{1}{5!}f^{(5)}(x^*)e_k^5 \\ &\quad + \frac{1}{6!}f^{(6)}(x^*)e_k^6 + O(e_k^7) \end{aligned} \tag{7}$$

$$f(x_k) \;=\; f'(x^*)\left(e_k + \alpha_2 e_k^2 + \alpha_3 e_k^3 + \alpha_4 e_k^4 + \alpha_5 e_k^5 + \alpha_6 e_k^6 + O(e_k^7)\right) \tag{8}$$

$$f'(x_k) \;=\; f'(x^*)\left(71 + 2\alpha_2 e_k + 3\alpha_3 e_k^2 + 4\alpha_4 e_k^3 + 5\alpha_5 e_k^4 + 6\alpha_6 e_k^5 + 7\alpha_7 e_k^6 + O(e_k^7)\right) \tag{9}$$

where $\alpha_i \;=\; \frac{1}{i!}\frac{f^{(i)}(x^*)}{f'(x^*)}, \quad i \;=\; 1, 2, \ldots$

From equations (8) and (9), we have:

$$\begin{aligned} \frac{f(x_k)}{f'(x_k)} &= e_k - \alpha_2 e_k^2 + (2\alpha_2^2 - 2\alpha_3)e_k^3 + (7\alpha_2\alpha_3 - 4\alpha_2^3 - 3\alpha_4)e_k^4 \\ &\quad + \left(10\alpha_2\alpha_4 + 8\alpha_2^4 - 20\alpha_2^2\alpha_3 + 6\alpha_3^2 - 4\alpha_5\right)e_k^5 \\ &\quad - \left(16\alpha_2^5 - 52\alpha_2^3\alpha_3 + 28\alpha_2^2\alpha_4 + 33\alpha_2\alpha_3^2 - 13\alpha_2\alpha_5 - 17\alpha_3\alpha_4 + 5\alpha_6\right)e_k^6 + O(e_k^7) \end{aligned} \tag{10}$$

With the help of (8) – (10), we have:

$$\begin{aligned} y_k &= x^* + \alpha_2 e_k^2 - \left(2\alpha_2^2 - 2\alpha_3\right)e_k^3 - \left(7\alpha_2\alpha_3 - 4\alpha_2^3 - 3\alpha_4\right)e_k^4 \\ &\quad - \left(10\alpha_2\alpha_4 + 8\alpha_2^4 - 20\alpha_2^2\alpha_3 + 6\alpha_3^2 - 4\alpha_5\right)e_k^5 + O\left(e_k^6\right) \\ &\quad + \left(16\alpha_2^5 - 52\alpha_2^3\alpha_3 + 28\alpha_2^2\alpha_4 + 33\alpha_2\alpha_3^2 - 13\alpha_2\alpha_5 - 17\alpha_3\alpha_4 + 5\alpha_6\right)e_k^6 + O(e_k^7) \end{aligned} \tag{11}$$

Using Taylors series for $f(y_k)$ expanded about x^*, we have:

$$\begin{aligned} f(y_k) = f'(x^*)[&\alpha_2 e_k^2 - (2\alpha_2^2 - 2\alpha_3)e_k^3 - (7\alpha_2\alpha_3 - 5\alpha_2^3 - 3\alpha_4)e_k^4 \\ &-\left(10\alpha_2\alpha_4 + 12\alpha_2^4 - 24\alpha_2^2\alpha_3 + 6\alpha_3^2 - 4\alpha_5\right)e_k^5 \\ &+\left(28\alpha_2^5 - 73\alpha_2^3\alpha_3 + 34\alpha_2^2\alpha_4 + 37\alpha_2\alpha_3^2 - 13\alpha_2\alpha_5 - 17\alpha_3\alpha_4 + 5\alpha_6\right) + O(e_k^7)] \end{aligned} \quad (12)$$

With the help of (8) – (12), we have:

$$\begin{aligned} z_k = x^* &+ (2\alpha_2^3 - \alpha_2\alpha_3)e_k^4 + (-6\alpha_2^4 + 12\alpha_2^2\alpha_3 - 2\alpha_2\alpha_4 - 2\alpha_3^2)e_k^5 \\ &+ (10\alpha_2^5 - 38\alpha_2^3\alpha_3 + 18\alpha_2^2\alpha_4 + 22\alpha_2\alpha_3^2 - 3\alpha_2\alpha_5 - 7\alpha_3\alpha_4)e_k^6 + O(e_k^7) \end{aligned} \quad (13)$$

Using Taylors series for $f(z_k)$ expanded about x^*, we have:

$$\begin{aligned} f(z_k) = f'(x^*)[&(2\alpha_2^3 - \alpha_2\alpha_3)e_k^4 - (6\alpha_2^4 - 12\alpha_2^2\alpha_3 + 2\alpha_2\alpha_4 + 2\alpha_3^2)e_k^5 \\ &+ (10\alpha_2^5 - 38\alpha_2^3\alpha_3 + 18\alpha_2^2\alpha_4 + 22\alpha_2\alpha_3^2 - 3\alpha_2\alpha_5 - 7\alpha_3\alpha_4)e_k^6 + O(e_k^7)] \end{aligned} \quad (14)$$

Using equations (9) – (14) in (6), we have:

$$x_{k+1} = x^* + (-3\alpha_2^4\alpha_3 + 3\alpha_2^2\alpha_3^2)e_k^7 + O(e_k^8) \quad (15)$$

That is:

$$e_{k+1} = (-3\alpha_2^4\alpha_3 + 3\alpha_2^2\alpha_3^2)e_k^7 + O(e_k^8) \quad (16)$$

From equation (16) it is shown that the order of convergence of Algorithm 2.1 is seven.

Results

Now, we present some numerical test results to verify the efficiency of the seventh-order methods defined by Algorithm 2.1 (AL). Also, we compare their result with method of Hafiz and Goria (2012) (HG), the method of Srisarakham and Thongmoon (2016) (ST), the method of Bawazir (2021) (BA) and the method of Thota and Shanmugasundaram (2022) (TS), which are the same seventh-order convergence. Starting from a predetermined first estimate x_0, where x^* is the exact root computed with 500 significant digits. The Maple software is used for all numerical computations, with a maximum number of computations of 100. For $\varepsilon = 10^{-15}$, so that the repeated calculations programs in the computers are stopped, we decide on the stopping standards $\left|x_{n+1} - x_n\right| < \varepsilon$ or $\left|f(x_{n+1})\right| < \varepsilon$.

The test equations are listed as follows.

$f_1\left(x\right) = x^3 - 4x^2 - 10,\ x_0 = 1$ $\qquad$ $f_2(x) = \sin^2 x - x^2 + 1,\ x_0 = -1$

$f_3\left(x\right) = x^3 - 10,\ x_0 = 1.5$ $\qquad$ $f_4(x) = 10e^{x^2} - 1,\ x_0 = 1$

$f_5(x) = \left(\sin x - \cos x\right)^2,\ x_0 = 1$ $\qquad$ $f_6(x) = \left(e^x - 4x^2\right)^2,\ x_0 = 1.7$

Discussion

We observe that our iterative method (AL) is comparable with all the methods cited in the Table 1 and gives better results in terms of a smaller number of iterations /speed and convergence to approximation value very near to the root of the problems. As shown in Table 1 the proposed methods are preferable to the other known methods of the same order with seventh-order convergence.

Conclusions

In this work we presented new iterative methods for solving nonlinear equations. The new methods have the seventh order of convergence. Algorithm 2.1 are developed from a concept of Shengfeng, it requires three evaluations of the function. For Algorithm 2.1, it does not require to compute the second or higher derivatives. Then we give numerical results to show presented methods and compare it with other seventh-order methods. We have proved the proposed iterative methods behaved a lesser of iteration for the test examples, that shows the efficiency of the two new developed methods.

Table 1 Numerical experiments and comparison of different iterative methods.

Method	n	x_n	$\lvert f(x_n)\rvert$	$\lvert x_n - x_{n-1}\rvert$
f_1				
GH	3	1.4142135623730950488016887	6.6×10^{-42}	4.8×10^{-11}
ST	2	1.4142135623730950488016887	2.3×10^{-34}	2.3×10^{-5}
BA	2	1.4142135623730950488016887	1.2×10^{-33}	3.0×10^{-5}
TS	2	1.4142135623730950488016887	3.9×10^{-45}	1.6×10^{-5}
AL	2	1.4142135623730950488016887	4.0×10^{-48}	2.1×10^{-6}
f_2				
GH	3	-1.404491648215341228795927	6.9×10^{-18}	4.2×10^{-5}
ST	2	-1.404491648215341225457270	1.4×10^{-18}	2.9×10^{-3}
BA	3	-1.404491648215341226035086	2.5×10^{-110}	2.2×10^{-16}
TS	3	-1.404491648215341226035086	1.5×10^{-82}	6.9×10^{-10}
AL	2	-1.404491648215341226034674	1.0×10^{-21}	1.3×10^{-3}
f_3				
GH	3	2.1544346900318837217665007	1.0×10^{-19}	1.4×10^{-5}
NT	2	2.1544346900318837217297697	4.1×10^{-19}	2.9×10^{-3}
BA	2	2.1544346900318837213396066	5.8×10^{-18}	4.3×10^{-3}
TS	3	2.1544346900318837177273322	5.6×10^{-17}	1.8×10^{-2}
AL	2	2.1544346900318837217592423	7.1×10^{-22}	1.3×10^{-3}
f_4				
GH	3	1.5174271293851463508566949	1.9×10^{-20}	6.7×10^{-6}
NT	2	1.5174271293851463506945032	5.1×10^{-19}	1.7×10^{-3}
BA	2	1.5174271293851463503400350	1.6×10^{-18}	2.0×10^{-3}
TS	2	1.5174271293851463508629722	3.4×10^{-25}	1.2×10^{-3}
AL	2	1.5174271293851463508629723	4.0×10^{-27}	1.3×10^{-4}
f_5				
GH	14	0.7853981742515395086600038	2.4×10^{-16}	2.5×10^{-8}
NT	9	0.7853981754374789977898933	2.9×10^{-16}	6.5×10^{-8}
BA	9	0.7853981726236354115037489	1.7×10^{-16}	5.1×10^{-8}
TS	9	0.7853981679591610866655284	4.2×10^{-17}	2.8×10^{-8}
AL	8	0.7853981765630794885491064	3.5×10^{-16}	9.2×10^{-8}
f_6				
GH	17	0.7148059157126977588322505	1.5×10^{-16}	7.8×10^{-9}
NT	11	0.7148059155434670325426786	1.4×10^{-16}	1.7×10^{-8}
BA	11	0.7148059146564629942154286	7.1×10^{-17}	1.3×10^{-8}
TS	10	0.7148059189331188497840683	5.8×10^{-16}	4.0×10^{-8}
AL	8	0.7148059135083798411261142	1.8×10^{-17}	8.0×10^{-9}

References

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

Bawazir, H.M. (2021) Seventh and Twelfth-Order Iterative Methods for Roots of Nonlinear Equations, *Hadhramout University Journal of Natural & Applied Sciences*, *18*(1), 9-15.

Chun, C. (2007). A family of composite fourth-order iterative methods for solving non-linear equations. *Applied Mathematics and Computation*, *187,* 951–956.

Chun, C. (2007). Some variants of Kings fourth-order family of methods for nonlinear equations. *Applied Mathematics and Computation, 190,* 57–62.

Chun, C. (2018). Some fourth-order iterative methods for solving non-linear equations*. Applied Mathematics and Computation, 195,* 454–459.

Cordero, A., & Torregrosa, J R. (2007). Variants of Newton's method using fifth-order formulas. *Applied Mathematics and Computation*, *190,* 686–698.

Edwar, I. P., Imran, M., & Deswita, L. (2016). A Sixth-Order Derivative-Free Iteration Method for Solving Nonlinear Equations. *Bulletin of Mathematics*, *8,* 1–8.

Fardi, M., Ghasemi, M., & Davari, A. (2012). New Iterative Methods with Seventh-Order Convergence For Solving Nonlinear Equations. *International Journal of Nonlinear Analysis and Applications*, *3,* 31-37.

Hafiz, M. A., & Al-Goria, S. M. H. (2012). New ninth– and seventh–order methods for solving nonlinear equations. *European Scientific Journal*, *8*(27), 83-95.

Kong-ied, B. (2021). Two new eighth and twelfth order iterative methods for solving nonlinear equations. *International Journal of Mathematics and Computer Science*, *16,* 333- 334.

Muhaijir, M. N., Imran, M., & Gamal, M. D. H. (2016). Variants of Chebyshev's Method with Eighth-Order Convergence for Solving Nonlinear Equations. *Applied and Computational Mathematics*, *5,* 247–251.

Noor, M.A. (2007). *New family of Iterative methods for nonlinear equations. Applied Mathematics and Computation,* 190, 553–558.

Shengfeng, L., & Dong, Y. (2019). Viscovatov-Like Algorithm of Thiele–Newton's Blending Expansion for a Bivariate Function. *Institute of Applied Mathematics*, *7,* 696.

Shengfeng, L. (2019). Fourth-order iterative method without calculating the higher derivatives for nonlinear equation. *Journal of Algorithms & Computational Technology*, *13,* 1-8.

Shin, M. K., Faisal, A., & Arif, R. (2016). Iterative methods for solving scalar equations. *Journal of Nonlinear Sciences and Applications, 9,* 1035-1042.

Srisarakham, N., & Thongmoon, M. (2016). A Note on Three-Step Iterative Method with Seventh Order of Convergence for Solving Nonlinear Equations. *Thai Journal of Mathematics, 14*(3), 565-573.

Thota, S., & Shanmugasundaram, P. (2022). On new sixth and seventh order iterative methods for solving non-linear equations using homotopy perturbation technique. *BMC Research Note, 15,* 267

Weerakoon, S., & Fernando, T. G. I. (2000). A variant of Newton's method with accelerated third-order Convergence. *Applied Mathematics Letters, 13*(8), 87–93.