Hadhramout University Journal of Natural & Applied Sciences

Volume 18
Issue 1 *6/1/2021*

Article 2

2021

Seventh and Twelfth-Order Iterative Methods for Roots of Nonlinear Equations

Hassan Mohamed Bawazir
Faculty of Education, Seiyun University-Yemen., hbawazir@yahoo.com

NAS journal Hadhramout University

Contents lists available at https://digitalcommons.aaru.edu.jo/huj_nas/.

Hadhramout University Journal of Natural & Applied Science

Article

Digital Object Identifier:
Received 19 October 2020,
Accepted 1 March 2021,
Available online 13 December 2021

Seventh and Twelfth-Order Iterative Methods for Roots of Nonlinear Equations

Hassan Mohamed Bawazir [1,*]

[1]Faculty of Education, Seiyun University-Yemen.

*Corresponding author: hbawazir@yahoo.com

This is an open-access article underproduction of Hadhramout University Journal of Natural & Applied Science with eISSN xxxxxxxxx

Abstract: This study presents two iterative methods, based on Newton's method, to attain the numerical solutions of nonlinear equations. We prove that our methods have seven and twelve orders of convergence. Analytical investigation has been established to show that our schemes have higher efficiency indexes than some recent methods. Numerical examples are executed to investigate the performance of the proposed schemes. Moreover, the theoretical order of convergence is verified on the numerical examples.

Keywords: Nonlinear Equation; Iterative Method; Newton's Method; Convergence Order.

1. Introduction:

The importance of finding the roots of nonlinear equations comes out from its applications in science and engineering [15, 16, 12, 10]. Many numerical applications use high precision in their computation, so higher-order numerical methods are required [8].

Solving some differential equations and integral equations requires finding the roots of nonlinear equations [5, 11]. In this study, we introduce two new iterative methods to find a simple root λ of a nonlinear equation $f(x)=0$, where $f: I \subset R \to R$: is a scalar function on an open interval I

One of the simple well-known methods for solving nonlinear equations is the classical Newton's method (CN).

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

The method provides a sequence of approximations that converge quadratically to a simple zero of f.

In recent years, some high order iterative methods for solving nonlinear equations have been improved and investigated see [15, 4, 14, 13, 9, 1, 2, 3] and the references therein.

In this paper, we consider the double Newton's method (DN)

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

which converges in fourth order [9].

First, we present a variant of the Double-Newton's method with seventh-order convergence. Then a twelfth-order iterative method is proposed. Finally, numerical examples are given to show the performance of the two methods.

Basic definitions:

Definition 1. [6] Let λ be a simple zero of a real function $f(x)$, let $<x_n>$ be a real sequence that converges towards λ. We say that the order of convergence of the sequence is $\alpha \in R^+$ if there exists $\beta \in R^+$ such that $\lim_{n\to\infty} |x_{n+1} - \lambda| / |x_n - \lambda|^\alpha = \beta$, β is called asymptotic error constant. If $\alpha = 2$ or 3 the sequence is said to have quadratic convergence or cubic convergence, respectively.

Definition 2. [12] Let $\varepsilon_n = x_n - \lambda$ is the error in the n^{th} iteration, we call the relation

$$\varepsilon_{n+1} = \beta \varepsilon_n^{\alpha} + O(\varepsilon_n^{\alpha+1}) \tag{3}$$

as the error equation. If we can obtain the error equation for any iterative method, then the value of α is its order of convergence.

If x_{n+1}, x_n and x_{n-1} are three successive iterations closer to the root λ. Then, the computational order of convergence ρ (see [16]) is approximated by using (3) as

$$\rho = \frac{\ln |(x_{n+1} - \lambda)/(x_n - \lambda)|}{\ln |(x_n - \lambda)/(x_{n-1} - \lambda)|} \quad . \tag{4}$$

The efficiency index is $p^{1/w}$, where p is the order of the method and w is the number of function evaluations per iteration required by the method, see [7].

2. Convergence Analysis

Firstly, we introduce new seventh-order iterative method as follows

$$\begin{aligned} y_n &= x_n - f(x_n)/f'(x_n), \\ z_n &= y_n - f(y_n)(1 + A/2)/f'(y_n) \\ x_{n+1} &= z_n + \frac{f(z_n) f(y_n)(1 + A/2)}{(f(z_n) - f(y_n)) f'(y_n)} \end{aligned} \tag{5}$$

where $A = \dfrac{f(y_n)\left(f'(x_n) - f'(y_n)\right)}{f(x_n) f'(y_n)}$.

For the method (5) we have the following convergence result.

Theorem 1 Let λ be a simple zero of sufficiently differentiable function $f : I \subseteq R \to R$ for an open interval I . If x_0 is sufficiently close to λ, then the method defined by (5) is of seventh-order and satisfies the error equation

$$\varepsilon_{n+1} = c_2^4 (2c_2^2 - 1.5c_3)\varepsilon_n^7 + O(\varepsilon_n^8)$$

where $\varepsilon_n = x_n - \lambda,\ c_k = f^{(k)}(\lambda)/(k!\, f'(\lambda))$.

Proof.

Using Taylor expansion of $f(x_n)$ and $f'(x_n)$ about λ , we have

$$f(x_n) = f'(\lambda)[\varepsilon_n + c_2\varepsilon_n^2 + c_3\varepsilon_n^3 + O(\varepsilon_n^4)] , \tag{6}$$

$$f'(x_n) = f'(\lambda)[1 + 2c_2\varepsilon_n + 3c_3\varepsilon_n^2 + O(\varepsilon_n^3)], \tag{7}$$

therefore

$$\begin{aligned} & f(x_n)/f'(x_n) \\ &= \varepsilon_n - c_2\varepsilon_n^2 + 2(c_2^2 - c_3)\varepsilon_n^3 + (7c_2c_3 - 4c_2^3 - 3c_4)\varepsilon_n^4 \\ &\quad + (6c_3^2 + 10c_2c_4 + 8c_2^4 - 4c_5 - 20c_2^2c_3)\varepsilon_n^5 + (17c_3c_4 \\ &\quad + 13c_2c_5 + 52c_2^3c_3 - 28c_2^2c_4 - 5c_6 - 33c_2c_3^2 - 16c_2^5)\varepsilon_n^6 \\ &\quad + (16c_2c_6 - 6c_7 - 36c_2^2c_5 + 22c_3c_5 - 92c_2c_3c_4 + 70c_2^3c_4 \\ &\quad + 12c_4^2 + 126c_2^2c_3^2 - 18c_3^3 - 128c_2^4c_3 + 32c_2^6)\varepsilon_n^7 + O(\varepsilon_n^8) \end{aligned} \quad , \tag{8}$$

so

$$\begin{aligned} & d_n = y_n - \lambda = \varepsilon_n - f(x_n)/f'(x_n) \\ &= c_2\varepsilon_n^2 + 2(c_3 - c_2^2)\varepsilon_n^3 + (4c_2^3 + 3c_4 - 7c_2c_3)\varepsilon_n^4 + (4c_5 + 20c_2^2c_3 \\ &\quad - 6c_3^2 - 10c_2c_4 - 8c_2^4)\varepsilon_n^5 + (28c_2^2c_4 + 5c_6 + 33c_2c_3^2 + 16c_2^5 \\ &\quad - 17c_3c_4 - 13c_2c_5 - 52c_2^3c_3)\varepsilon_n^6 + (6c_7 - 16c_2c_6 + 36c_2^2c_5 \\ &\quad - 22c_3c_5 + 92c_2c_3c_4 - 70c_2^3c_4 - 12c_4^2 - 126c_2^2c_3^2 + 18c_3^3 \\ &\quad + 128c_2^4c_3 - 32c_2^6)\varepsilon_n^7 + O(\varepsilon_n^8) \end{aligned} \quad . \tag{9}$$

Taylor expansions of $f(y_n)$ and $f'(y_n)$ around λ are given as

$$f(y_n) = f'(\lambda)[d_n + c_2 d_n^2 + c_3 d_n^3 + O(d_n^4)], \tag{10}$$

$$f'(y_n) = f'(\lambda)[1 + 2c_2 d_n + 3c_3 d_n^2 + O(d_n^3)], \tag{11}$$

so, by (9), we attain

$$\begin{aligned} & f(y_n) \\ &= f'(\lambda)[c_2\varepsilon_n^2 + 2(c_3 - c_2^2)\varepsilon_n^3 + (5c_2^3 + 3c_4 - 7c_2c_3)\varepsilon_n^4 \\ &\quad + (24c_2^2c_3 + 4c_5 - 12c_2^4 - 6c_3^2 - 10c_2c_4)\varepsilon_n^5 + (28c_2^5 \\ &\quad + 34c_2^2c_4 + 5c_6 + 37c_2c_3^2 - 73c_2^3c_3 - 17c_3c_4 - 13c_2c_5)\varepsilon_n^6 \\ &\quad + (6c_7 - 16c_2c_6 + 44c_2^2c_5 - 22c_3c_5 + 104c_2c_3c_4 - 102c_2^3c_4 \\ &\quad - 12c_4^2 - 160c_2^2c_3^2 + 18c_3^3 + 206c_2^4c_3 - 64c_2^6)\varepsilon_n^7 + O(\varepsilon_n^8)] \end{aligned} \tag{12}$$

Furthermore

$$\begin{aligned} f'(y_n) = f'(\lambda)[1 + 2c_2^2\varepsilon_n^2 &+ 4c_2(c_3 - c_2^2)\varepsilon_n^3 + (8c_2^4 + 6c_2c_4 \\ & - 11c_2^2c_3)\varepsilon_n^4 + (8c_2c_5 - 20c_2^2c_4 + 28c_2^3c_3 \\ & - 16c_2^5)\varepsilon_n^5 + (60c_2^3c_4 + 10c_2c_6 + 32c_2^6 - 68c_2^4c_3 \\ & - 16c_2c_3c_4 + 12c_3^3 - 26c_2^2c_5)\varepsilon_n^6 + O(\varepsilon_n^7)] \end{aligned} , \tag{13}$$

now,

$$\begin{aligned} & f(y_n)/f'(y_n) \\ &= d_n - c_2 d_n^2 + 2(c_2^2 - c_3)d_n^3 + (7c_2c_3 - 4c_2^3 - 3c_4)d_n^4 + O(d_n^5) \end{aligned} , \tag{14}$$

so by (9), we get

$$f(y_n)/f'(y_n)$$
$$=c_2\varepsilon_n^2+2(c_3-c_2^2)\varepsilon_n^3+(3c_2^3+3c_4-7c_2c_3)\varepsilon_n^4+(4c_5+16c_2^2c_3$$
$$-4c_2^4-6c_3^2-10c_2c_4)\varepsilon_n^5+(6c_2^5-32c_2^3c_3+29c_2c_3^2+22c_2^2c_4 \quad (15)$$
$$+5c_6-17c_3c_4-13c_2c_5)\varepsilon_n^6+(6c_7-16c_2c_6+28c_2^2c_5-12c_3c_5$$
$$+80c_2c_3c_4-38c_2^3c_4-12c_4^2-98c_2^2c_3^2+18c_3^3$$
$$+68c_2^4c_3-12c_2^6)\varepsilon_n^7+O(\varepsilon_n^8)$$

Using (7) and (13), we achieve

$$f'(x_n)-f'(y_n)$$
$$=f'(\lambda)[2c_2\varepsilon_n+(3c_3-2c_2^2)\varepsilon_n^2+4(c_4-c_2c_3+c_2^3)\varepsilon_n^3$$
$$+(5c_5-8c_2^4-6c_2c_4+11c_2^2c_3)\varepsilon_n^4+(6c_6-8c_2c_5 \quad ,(16)$$
$$+20c_2^2c_4-28c_2^3c_3+16c_2^5)\varepsilon_n^5+O(\varepsilon_n^6)]$$

therefore

$$f(y_n)\left(f'(x_n)-f'(y_n)\right)$$
$$=[f'(\lambda)]^2[2c_2^2\varepsilon_n^3+(7c_2c_3-6c_2^3)\varepsilon_n^4+(10c_2c_4-28c_2^2c_3$$
$$+18c_2^4+6c_3^2)\varepsilon_n^5+(13c_2c_5-50c_2^5-40c_2^2c_4+104c_2^3c_3 \quad (17)$$
$$+17c_3c_4-41c_2c_3^2)\varepsilon_n^6+O(\varepsilon_n^7)]$$

By (6) and (13), we get

$$f(x_n)f'(y_n)$$
$$=[f'(\lambda)]^2[\varepsilon_n+c_2\varepsilon_n^2+(c_3+2c_2^2)\varepsilon_n^3$$
$$+(c_4-2c_2^3+4c_2c_3)\varepsilon_n^4 \quad ,(18)$$
$$+(c_5-5c_2^2c_3+4c_2^4+6c_2c_4)\varepsilon_n^5+O(\varepsilon_n^6)]$$

and hence, by applying (17) and (18), we attain

$$A=\frac{f(y_n)(f'(x_n)-f'(y_n))}{f(x_n)f'(y_n)}$$
$$=2c_2^2\varepsilon_n^2+(7c_2c_3-8c_2^3)\varepsilon_n^3+(10c_2c_4-37c_2^2c_3 \quad (19)$$
$$+22c_2^4+6c_3^2)\varepsilon_n^4+(13c_2c_5-52c_2^5-52c_2^2c_4$$
$$+127c_2^3c_3+17c_3c_4+28c_2c_3^2)\varepsilon_n^5+O(\varepsilon_n^6)$$

Now, from (15) and (19) we obtain

$$\frac{f(y_n)}{f'(y_n)}A$$
$$=2c_2^3\varepsilon_n^4+(11c_2^2c_3-12c_2^4)\varepsilon_n^5+(16c_2^2c_4-78c_2^3c_3 \quad ,(20)$$
$$+44c_2^5+20c_2c_3^2)\varepsilon_n^6+(21c_2^2c_5-128c_2^6-116c_2^3c_4$$
$$+354c_2^4c_3+58c_2c_3c_4-119c_2^2c_3^2+12c_3^3)\varepsilon_n^7+O(\varepsilon_n^8)$$

Furthermore, using (9) and (15) we get

$$d_n-f(y_n)/f'(y_n)$$
$$=c_2^3\varepsilon_n^4+4(c_2^2c_3-c_2^4)\varepsilon_n^5+2(3c_2^2c_4-10c_2^3c_3+5c_2^5$$
$$+2c_2c_3^2)\varepsilon_n^6+4(2c_2^2c_5+3c_2c_3c_4-8c_2^3c_4-7c_2^2c_3^2 \quad (21)$$
$$+15c_2^4c_3-5c_2^6)\varepsilon_n^7+O(\varepsilon_n^8)$$

Using (21), (20) and (9) we obtain

$$\tilde{d}_n=z_n-\lambda$$
$$=d_n-\frac{f(y_n)}{f'(y_n)}-\frac{f(y_n)}{f'(y_n)}\frac{A}{2}$$
$$=(2c_2^4-1.5c_2^2c_3)\varepsilon_n^5+(19c_2^3c_3-2c_2^4c_4-12c_2^5-6c_2c_3^2)\varepsilon_n^6 \quad (22)$$
$$+(44c_2^6+31.5c_2^2c_3^2-2.5c_2^2c_5-17c_2c_3c_4-26.5c_2^3c_4$$
$$-117c_2^4c_3-6c_3^3)\varepsilon_n^7+O(\varepsilon_n^8)$$

Expanding $f(z_n)$ about λ is given as

$$f(z_n)=f'(\lambda)[\tilde{d}_n+c_2\tilde{d}_n^2+c_3\tilde{d}_n^3+\cdots] \quad ,(23)$$

therefore, by (22) we get

$$f(z_n)$$
$$=f'(\lambda)[(2c_2^4-1.5c_2^2c_3)\varepsilon_n^5+(19c_2^3c_3-2c_2^4c_4-12c_2^5-6c_2c_3^2)\varepsilon_n^6$$
$$+(44c_2^6+31.5c_2^2c_3^2-2.5c_2^2c_5-17c_2c_3c_4-26.5c_2^3c_4 \quad .(24)$$
$$-117c_2^4c_3-6c_3^3)\varepsilon_n^7+O(\varepsilon_n^8)]$$

Using (12) and (24), we get

$$f(z_n)-f(y_n)$$
$$=f'(\lambda)[-c_2\varepsilon_n^2+2(c_2^2-c_3)\varepsilon_n^3+(7c_2c_3-5c_2^3-3c_4)\varepsilon_n^4$$
$$+(14c_2^4+6c_3^2+10c_2c_4-25.5c_2^2c_3-4c_5)\varepsilon_n^5+(16c_2^5$$
$$+34c_2^2c_4+5c_6+31c_2c_3^2-54c_2^3c_3-17c_3c_4-13c_2c_5 \quad (25)$$
$$-2c_2^4c_4)\varepsilon_n^6+(16c_2c_6-6c_7-46.5c_2^2c_5+22c_3c_5$$
$$-121c_2c_3c_4+75.5c_2^3c_4+12c_4^2+191.5c_2^2c_3^2-24c_3^3$$
$$-323c_2^4c_3+108c_2^6)\varepsilon_n^7++O(\varepsilon_n^8)]$$

therefore,

$$f(z_n)/(f(z_n)-f(y_n))$$
$$=(1.5c_2c_3-2c_2^3)\varepsilon_n^3+(2c_2^3c_4+8c_2^4+3c_3^2-12c_2^2c_3)\varepsilon_n^4$$
$$+(9c_2c_3^2-18c_2^5+2.5c_2c_5+12.5c_3c_4+32.5c_2^2c_4 \quad (26)$$
$$+55.5c_2^3c_3+4c_2^4c_4-4c_2^2c_3c_4)\varepsilon_n^5+O(\varepsilon_n^6)$$

In addition, using (15) and (20), we obtain

$$f(y_n)(1+A/2)/f'(y_n)$$
$$=c_2\varepsilon_n^2+2(c_3-c_2^2)\varepsilon_n^3+(4c_2^3+3c_4-7c_2c_3)\varepsilon_n^4 \quad (27)$$
$$+(4c_5+21.5c_2^2c_3-10c_2^4-6c_3^2-10c_2c_4)\varepsilon_n^5+O(\varepsilon_n^6)$$

From (26) and (27) we get

$$\frac{f(z_n)f(y_n)(1+A/2)}{(f(z_n)-f(y_n))f'(y_n)}$$
$$=(1.5c_2^2c_3-2c_2^4)\varepsilon_n^5+(2c_2^4c_4+12c_2^5+6c_2c_3^2-19c_2^3c_3)\varepsilon_n^6 \quad (28)$$
$$+(2.5c_2^2c_5+17c_2c_3c_4+26.5c_2^3c_4+115.5c_2^4c_3+6c_3^3$$
$$-42c_2^6-31.5c_2^2c_3^2)\varepsilon_n^7+O(\varepsilon_n^8)$$

(28)

Finally, using (22) and (28) we get the error relation:

$$\varepsilon_{n+1}=x_{n+1}-\lambda$$
$$=\tilde{d}_n+\frac{f(z_n)f(y_n)(1+A/2)}{(f(z_n)-f(y_n))f'(y_n)}$$
$$=c_2^4(2c_2^2-1.5c_3)\varepsilon_n^7+O(\varepsilon_n^8)$$

This means that the method defined by (5) is of the seventh-order.

Finally, we construct the twelfth-order iterative method as follows

$$y_n=x_n-\frac{f(x_n)}{f'(x_n)},$$
$$z_n=y_n-\frac{f(y_n)}{f'(y_n)}\left[1+\frac{f(y_n)(f'(x_n)-f'(y_n))}{2f(x_n)f'(y_n)}\right] \quad (29)$$
$$x_{n+1}=z_n-\frac{f(z_n)}{f'(z_n)}\left[1+\frac{f(z_n)(f'(y_n)-f'(z_n))}{2f(y_n)f'(z_n)}\right]$$

Theorem 2 Let λ be a simple zero of sufficiently differentiable function $f:I\subseteq R\to R$ for an open interval I. If x_0 is sufficiently close to λ, then the method defined by (29) is of twelfth-order and satisfies the error equation

$$\varepsilon_{n+1}=c_2^5(c_2^2-1.5c_3)(2c_2^2-1.5c_3)^2\varepsilon_n^{12}+O(\varepsilon_n^{13})$$

Proof.

Using Taylor expansion of $f'(z_n)$ about λ, we have

$$f'(z_n)=f'(\lambda)[1+2c_2\tilde{d}_n+3c_3\tilde{d}_n^2+\cdots],$$

(30)

therefore, by (23) and (30), we obtain

$$f(z_n)/f'(z_n)$$
$$=\tilde{d}_n-c_2\tilde{d}_n^2+2(c_2^2-c_3)\tilde{d}_n^3+(7c_2c_3-4c_2^3-3c_4)\tilde{d}_n^4+O(\tilde{d}_n^5). \quad (31)$$

From (13), (22) and (30) we get

$$f'(y_n)-f'(z_n)$$
$$=f'(\lambda)[2c_2^2\varepsilon_n^2+4c_2(c_3-c_2^2)\varepsilon_n^3+(8c_2^4+6c_2c_4$$
$$-11c_2^2c_3)\varepsilon_n^4+(8c_2c_5-20c_2^2c_4+31c_2^3c_3 \quad (32)$$
$$-20c_2^5)\varepsilon_n^5+O(\varepsilon_n^6)]$$

therefore, by using (23) and (32) we attain

$$f(z_n)(f'(y_n)-f'(z_n))$$
$$=[f'(\lambda)]^2[2c_2^2\varepsilon_n^2\tilde{d}_n+4c_2(c_3-c_2^2)\varepsilon_n^3\tilde{d}_n+(8c_2^4+6c_2c_4 \quad (33)$$
$$-11c_2^2c_3)\varepsilon_n^4\tilde{d}_n+(8c_2c_5-20c_2^2c_4+31c_2^3c_3-20c_2^5)\varepsilon_n^5\tilde{d}_n+\cdots]$$

and using (12) and (30) we attain

$$f'(z_n)f(y_n)$$
$$=[f'(\lambda)]^2[c_2\varepsilon_n^2+2(c_3-c_2^2)\varepsilon_n^3+(5c_2^3+3c_4-7c_2c_3)\varepsilon_n^4$$
$$+(24c_2^2c_3+4c_5-12c_2^4-6c_3^2-10c_2c_4)\varepsilon_n^5+O(\varepsilon_n^6)]$$

, (34)

therefore

$$\frac{f(z_n)(f'(y_n)-f'(z_n))}{2f(y_n)f'(z_n)} \quad (35)$$
$$=c_2\tilde{d}_n+(1.5c_2c_3-c_2^3)\varepsilon_n^2\tilde{d}_n+(3c_3^2-3.5c_2^2c_3)\varepsilon_n^3\tilde{d}_n+\cdots$$

Finally, using (31) and (35) we attain the error relation:

$$\varepsilon_{n+1}=x_{n+1}-\lambda$$
$$=\tilde{d}_n-[\tilde{d}_n-c_2\tilde{d}_n^2+\cdots][1+c_2\tilde{d}_n+(1.5c_2c_3-c_2^3)\varepsilon_n^2\tilde{d}_n$$
$$+(3c_3^2-3.5c_2^2c_3)\varepsilon_n^3\tilde{d}_n+\cdots]$$
$$=(c_2^3-1.5c_2c_3)\varepsilon_n^2\tilde{d}_n^2+(3.5c_2^2c_3-3c_3^2)\varepsilon_n^3\tilde{d}_n^2+\cdots$$
$$=c_2^5(c_2^2-1.5c_3)(2c_2^2-1.5c_3)^2\varepsilon_n^{12}+O(\varepsilon_n^{13})$$

This means that the method defined by (29) is of the twelfth-order.

The method (5), requires 5 function evaluations, 3 of f and 2 of f', whereas the method (29), requires 6 function evaluations, 3 of f and 3 of f'. The methods (5) and (29) have the efficiency indexes $7^{1/5}=1.4757$ and $12^{1/6}=1.5131$, respectively, which are better than the efficiency index $2^{1/2}=1.4142$ of the Newton's method (1) and the double Newton's method (2) and the efficiency index $10^{1/6}=1.4678$ of the tenth-order (TO) method [13].

3. Numerical Examples and Conclusion

In this section, we employ the new methods defined by (5) and (29) to solve some nonlinear equations and compare them with Classical Newton's method (CN) (1), double Newton's method (DN) (2) and the tenth-order (TO) method [13].

We use the following functions, [9]:

$f_1(x) = x^3 + 4x^2 - 10,\ \lambda = 1.365230013414096887913 73,$

$f_2(x) = x^5 + x^4 + 4x^2 - 20,\ \lambda = 1.4662790738647226707058 7,$

$f_3(x) = e^{x^2+7x-30} - 1,\ \lambda = 3,$

$f_4(x) = (sinx)^2 - x^2 + 1,\ \lambda = 1.4044916482153411152467 0,$

$f_5(x) = e^x sinx + \ln(x^2 + 1),\ \lambda = 0,$

$f_6(x) = x^3 - sin^2 x + 3cosx + 5,\ \lambda = -1.5826870457520698654008 1,$

$f_7(x) = x^3 - e^{-x},\ \lambda = 0.7728829591492101247496 29.$

All numerical examples were performed in MatlabR2017b, using 200 digits floating point (digits: = 200), and variable precision arithmetic. We have computed the root of each test function for two different initial guesses x_0 for 7 real functions, listed above, while the iterative schemes were stopped when $|x_{n+1} - x_n| + |f(x_n)| < 10^{-200}$.

Table (1): The sequence of the approximation zeros of the function f_3 using Method (5) starting with $x_0 = 3.5$ under the stopping criterium $|x_{n+1} - x_n| + |f(x_n)| < 10^{-200}$

n	x_n
3	3.0000000000000033032238007803684548451677746278 4989199036522099 98001042550586337229548182508732656279594893865 13249888387965294 10685311997043489221669499959953226699301090133 91834034711552290 071213117
4	3.0000000000000000000000000000000000000000000000 0000000000000000 00000000000000000000000000000000003433312778739 46721040786731977 95834359895322406594017981907818223730490127338 03275296801347097 125849397
5	3.0000000000000000000000000000000000000000000000 000000000000000 00000000000000000000000000000000000000000000000 0000000000000000 00000000000000000000000000000000000000000000000 0000000000000000 000000000000
6	3.0000000000000000000000000000000000000000000000 000000000000000 00000000000000000000000000000000000000000000000 0000000000000000 00000000000000000000000000000000000000000000000 0000000000000000 000000000000

Table 2. Numerical results for different methods with stopping criterion $|x_{n+1} - x_n| + |f(x_n)| < 10^{-200}$

$f(x)$	x_0	IT CN (1)	DN (2)	TO [13]	Eq.(5)	Eq.(29)
f_1	1.9	9	5	3	3	3
	1	10	5	3	3	3
f_2	1.2	10	5	3	3	3
	2	11	6	3	3	3
f_3	3.5	17	9	5	6	5
	4	24	12	8	8	7
f_4	1.6	10	5	3	3	2
	2.5	11	6	3	3	3
f_5	3	11	5	4	3	3
	4.2	11	6	4	4	3
f_6	-1	10	5	3	3	3
	-3	11	6	3	3	3
f_7	0	11	6	3	3	3
	1.5	11	6	3	3	3

Table 3. Numerical results for different methods with stopping criterion $|x_{n+1} - x_n| + |f(x_n)| < 10^{-200}$

$f(x)$	x_0	CN (1) (IT, ρ)	DN (2) (IT, ρ)	TO [13] (IT, ρ)	Eq.(5) (IT, ρ)	Eq.(29) (IT, ρ)
f_1	1.9	(9, 2.0037)	(5, 4.0221)	(3, 10.2305)	(3, 7.3951)	(3, 12.4333)
	1	(10, 2.0018)	(5, 4.0213)	(3, 10.3010)	(3, 7.3912)	(3, 12.4010)
f_2	1.2	(10, 1.9999)	(5, 3.9992)	(3, 9.4298)	(3, 6.9656)	(3, 12.0802)
	2	(11, 2.0000)	(6, 3.9997)	(3, 9.5343)	(3, 6.9549)	(3, 12.1162)
f_3	3.5	(17,1.9957)	(9, 3.9742)	(5, 7.8713)	(6, 6.6614)	(5, 11.4713)
	4	(24, 1.9947)	(12, 3.9379)	(8, 9.3547)	(8, 6.0293)	(7, 11.5014)
f_4	1.6	(10, 1.8269)	(5, 4.0057)	(3, 9.9212)	(3, 7.0665)	(2,______)
	2.5	(11, 2.0006)	(6, 4.0033)	(3, 9.7953)	(3, 7.1723)	(3, 12.2018)
f_5	3	(10, 2.0002)	(5, 4.0024)	(3, 9.6135)	(3, 6.9992)	(3, 12.0332)
	4.2	(11, 2.0002)	(6, 4.0012)	(4, 9.9558)	(4, 6.9998)	(3, 07868)
f_6	-1	(10, 2.0029)	(5, 4.0353)	(3, 10.8454)	(3, 7.7287)	(3, 12.7012)
	-3	(11, 2.0022)	(6, 4.0134)	(3, 10.7783)	(3, 8.1064)	(3, 13.1207)
f_7	0	(11, 2.0002)	(6, 4.0010)	(3, 9.6018)	(3, 7.0394)	(3, 12.1509)
	1.5	(11, 2.0002)	(6, 4.0012)	(3, 9.6159)	(3, 7.0649)	(3, 12.2641)

Table 1 shows an example of the sequence of the approximation zeros of the function f_3 using the method (5) starting with $x_0 = 3.5$ under the stopping criterion $|x_{n+1} - x_n| + |f(x_n)| < 10^{-200}$.

Displayed in Table 2 are the numbers of iterations (IT) required such that $|x_{n+1} - x_n| + |f(x_n)| < 10^{-200}$. Table 3 shows the computational order ρ for all considered examples.

The computational results presented in Table 2 show that, the presented methods, (5) and (29) converge more rapidly than Classical Newton's method (1) and double Newton's method (2) and they require less number of iterations. Therefore, the new methods (5) and (29) have better convergence efficiency. Table 3 shows the computational orders of 5 methods, CN (1), DN (2), TO [13], (5), and (29). It can be seen from the numerical results displayed in Tables 2 that the numerical results of the proposed methods support the theoretical results proved in Section 2.

Finally, we conclude that the new iterative methods (5) and (29), presented in this paper, can compete with other efficient equation solvers, such as the Classical Newton's method (1), and the double Newton's method (2), and the tenth-order method [13]. The results reflect the efficiency indexes 1.4757, 1.5131, 1.4142, 1.4142 and 1.4678 of the methods (5), (29), (1), (2), and TO [13], respectively.

References:

[1] H. M. S. Bawazir, "Fifth and Eleventh-Order Iterative Methods for Roots of Nonlinear Equations," *Hadhramout University Journal of Natural & Applied Sciences*, vol. 17, pp. 209 – 219, 2020.

[2] H. M. S. Bawazir, "Eighth and Twelfth-Order Iterative Methods for Solving Nonlinear Equations" *International Journal of Multidisciplinary Sciences and Advanced Technology*. vo. 1(12), pp. 36 – 42, 2020.

[3] H. M. S. Bawazir, "Seventh and Eleventh-Order Iterative Methods for Solving Nonlinear Equation," *International Journal of Multidisciplinary Sciences and Advanced Technology*. vol. 1(10), pp. 56 – 63, 2020.

[4] R. Behl, V. Kanwar, and K. Sharma, "New modified optimal families of King's and Traub-Ostrowski's methods," *Numerical Analysis and Applications*, vol. 7, pp. 26-35, 2014.

[5] M. M. Bhatti, T. Abbas, and M. M. Rashidi, "Entropy generation as a practical tool of optimisation for non-Newtonian nanofluid flow through a permeable stretching surface using SLM," *Journal of Computational Design and Engineering*, vol. 4, pp. 21-28, 2017.

[6] R. L.Burden and J. D. Faires, *Numerical Analysis*. Thomson Brooks/Cole, 8th Edition, 2005.

[7] W. Gautschi, Numerical *Analysis: An Introduction*. Birkhäuser Boston Inc. Boston, 1997.

[8] M. Grau and J. L. Díaz-Barrero, "An improvement to Ostrowski root-finding method," *Applied Mathematics and Computation*, vol. 173, pp. 450-456, 2006.

[9] Z. Hu, L. Guocai, and L. Tian, "An iterative method with ninth-order convergence for solving nonlinear equations," *International Journal of Contemporary Mathematical Sciences*, vol. 6, pp. 17-23, 2011.

[10] D. R. Jackett, T. J. McDougall, R. Feistel, D. G. Wright, and S. M. Griffies, "Algorithms for density, potential temperature, conservative temperature, and the freezing temperature of seawater," *Journal of Atmospheric and Oceanic Technology*, vol. 23, pp. 1709-1728, 2006.

[11] K. Maleknejad, K. Nouri, and L. Torkzadeh, "Operational matrix of fractional integration based on the shifted second kind Chebyshev polynomials for solving fractional differential equations," *Mediterranean Journal of Mathematics*, vol. 13, pp. 1377-1390, 2016.

[12] T. J. McDougall, D. R. Jackett, D. G. Wright, and R. Feistel, "Accurate and computationally efficient algorithms for potential temperature and density of seawater," *Journal of Atmospheric and Oceanic Technology*, vol. 20, pp. 730-741, 2003.

[13] K. Nouri, H. Ranjbar, and L. Torkzadeh, "Two High Order Iterative Methods for Roots of Nonlinear Equations," *Punjab University Journal of Mathematics*, vol. 51, pp. 47-59, 2019.

[14] M. Saqib and M. Iqbal, "Sequence of Multi-Step Higher Order Iteration Schemes for Nonlinear Scalar Equations," Punjab Univ. j. math., vol. 49, pp. 51-58, 2017.

[15] J. R. Sharma, "Improved Chebyshev–Halley methods with sixth and eighth order convergence," *Applied Mathematics and Computation*, vol. 256, pp. 119-124, 2015.

[16] J. F. Traub, *Iterative methods for the solution of equations*, American Mathematical Soc., 1964.

[17] S. Weerakoon and T. Fernando, "A variant of Newton's method with accelerated third-order convergence," *Applied mathematics letters*, vol. 13, pp. 87-93, 2000.

طريقتان تكراريتان من الرتبة السابعة والرتبة الثانية عشرة لإيجاد جذور المعادلات غير الخطية

حسن محمد سعيد باوزير
كلية التربية، جامعة سيئون–حضرموت– اليمن

ملخص: تقدم هذه الدراسة طريقتين تكراريتين تعتمدان على طريقة نيوتن، للوصول إلى الحلول العددية للمعادلات غير الخطية. نثبت أن الطريقتين لها سبع واثنتا عشرة رتبة تقارب. تم إجراء تحقيق تحليلي لتوضيح أن الطريقتين بها مؤشرات كفاية أعلى من بعض الطرائق الحديثة. تم تنفيذ أمثلة عددية للتحقق من أداء الطرقتيين المقترحتين. علاوة على ذلك، يتم التحقق من الترتيب النظري للتقارب على الأمثلة العددية.

كلمات مفتاحية: معادلة غير خطية، طريقة تكرارية، طريقة نيوتن، رتبة التقارب.