Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 7(10): 18616–18631.
DOI: 10.3934/math.20221024
Received: 22 June 2022
Revised: 31 July 2022
Accepted: 03 August 2022
Published: 19 August 2022

Research article

Some new numerical schemes for finding the solutions to nonlinear equations

Awais Gul Khan[1], **Farah Ameen**[1], **Muhammad Uzair Awan**[1] **and Kamsing Nonlaopon**[2,*]

[1] Department of Mathematics, Government College University, Faisalabad 38023, Pakistan
[2] Department of Mathematics, Faculty of Science, Khon Kaen University, Khon Kaen 40002, Thailand

* **Correspondence:** Email: nkamsi@kku.ac.th; Tel: +66866421582.

Abstract: We introduce a sequence of third and fourth-order iterative schemes for finding the roots of nonlinear equations by using the decomposition technique and Simpson's one-third rule. We also discuss the convergence analysis of our suggested iterative schemes. With the help of different numerical examples, we demonstrate the validity, efficiency and implementation of our proposed schemes.

Keywords: convergence analysis; iterative method; decomposition technique; Simpson's one-third rule
Mathematics Subject Classification: 49J40, 90C33

1. Introduction

One of the oldest and most basic problems in mathematics is that of solving nonlinear equations. To solve these equations, we can use iterative methods such as Newton's method and its variants. Newton's method is one of the most powerful and well-known iterative methods known to converge quadratically.

In the Adomian decomposition method, the solution is considered in terms of an infinite series, which converges to an exact solution. Chun [5] and Abbasbandy [4] constructed and investigated different higher-order iterative methods by applying the decomposition technique of Adomian [3]. Darvishi and Barati [6] also applied the Adomian decomposition technique to develop Newton-type methods that are cubically convergent for the solution of the system of non-linear equations. Implementation of this Adomian decomposition technique required higher-order derivatives evaluation, which is the major pitfall of this method.

To overcome this drawback, several new techniques have been suggested and analyzed by many researchers. Daftardar-Gejji and Jafari [7] have used different modifications of the Adomian decomposition method [3] and proposed a simple technique that does not need the derivative evaluation of the Adomian polynomial, which is the major advantage of using this technique over Adomian decomposition method. Saqib and Iqbal [13] and Ali et al. [1, 2] have used this decomposition technique and developed a family of iterative methods with better efficiency and convergence order for solving the nonlinear equations. Heydari et al. [15, 16] proposed several iterative methods including derivative free methods and discussed their convergence. For finding multiple roots of nonlinear equations and iterative schemes using homotopy perturbation techniques, see [17, 18].

Weerakon and Fernando [14] improved the convergence of the Newton method using the quadrature rule. Later on, Ozban [11] investigated some new variant forms of the Newton method by using the concept of harmonic mean and mid-point rule. Noor [10] developed the fifth-order convergent iterative method using the Gaussian quadrature formula and investigated its efficacy compared to the existing methods in the literature.

In this paper, we consider the well know fixed-point iterative method in which we rewrite the nonlinear equation $\Lambda(\wp) = 0$ as $\wp = \Upsilon(\wp)$. We present and introduce some new iterative methods. We also determine the convergence analysis of proposed methods. Some numerical examples are presented to make a comparative study of newly constructed methods with known third and fourth-order convergent iterative algorithms.

2. Construction of iterative methods

This section comprises some new multi-step third and fourth-order convergent iterative methods in view of Simpson's one-third rule and decomposition technique [7].

2.1. *Simpson's one third rule*

Consider the nonlinear equation

$$\Lambda(\wp) = 0, \tag{2.1}$$

which is equivalent to

$$\wp = \Upsilon(\wp). \tag{2.2}$$

Assume that α is the simple root of nonlinear Eq (2.1) and γ is the initial guess sufficiently close to the root. Using fundamental theorem of calculus and Simpson's one-third quadrature formula, we have

$$\int_{\gamma}^{\wp} \Upsilon'(\wp)\, d\wp = \frac{\wp-\gamma}{6}\left\{\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right\},$$

$$\Upsilon(\wp) = \Upsilon(\gamma) + \frac{\wp-\gamma}{6}\left\{\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right\}. \tag{2.3}$$

Now, from (2.2) and (2.3), we have

$$\wp = \Upsilon(\gamma) + \frac{1}{6}(\wp-\gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right). \tag{2.4}$$

Now, using the technique of He [8], the nonlinear Eq (2.1) can be written as an equivalent coupled system of equations

$$\wp = \Upsilon(\gamma) + \frac{1}{6}(\wp - \gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right) + H(\wp),$$

and

$$\begin{aligned} H(\wp) &= \Upsilon(\wp) - \Upsilon(\gamma) - \frac{1}{6}(\wp - \gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right) \\ &= \wp\left[1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right)\right] \\ &\quad + \frac{\gamma}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right) - \Upsilon(\gamma), \end{aligned} \tag{2.5}$$

from which, it follows that

$$\begin{aligned} \wp &= \frac{H(\wp)}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right)} + \frac{\Upsilon(\gamma) - \frac{\gamma}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right)}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right)} \\ &= c + M(\wp), \end{aligned} \tag{2.6}$$

where

$$c = \gamma, \tag{2.7}$$

and

$$M(\wp) = \frac{H(\wp)}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right)} + \frac{\Upsilon(\gamma) - \gamma}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp+\gamma}{2}\right) + \Upsilon'(\wp)\right)}. \tag{2.8}$$

It is clear that $M(\wp)$ is nonlinear operator. Now, we establish the sequence of higher-order iterative methods implementing the decomposition technique presented by Daftardar-Gejji and Jafari [7]. In this technique, the solution of (2.1) can be represented in terms of infinite series.

$$\wp = \sum_{i=0}^{\infty} \wp_i. \tag{2.9}$$

Here, the operator $M(\wp)$ can be decomposed as:

$$M(\wp) = M(\wp_\circ) + \sum_{i=1}^{\infty}\left\{M\left(\sum_{j=0}^{i}\wp_j\right) - M\left(\sum_{j=0}^{i-1}\wp_j\right)\right\}. \tag{2.10}$$

Thus, from (2.6), (2.9) and (2.10), we have

$$\sum_{i=1}^{\infty}\wp_i = c + M(\wp_\circ) + \sum_{i=1}^{\infty}\left\{M\left(\sum_{j=0}^{i}\wp_j\right) - M\left(\sum_{j=0}^{i-1}\wp_j\right)\right\}, \tag{2.11}$$

which generates the following iterative schemes:

$$\begin{aligned}\wp_o &= c,\\ \wp_1 &= M(\wp_o),\\ \wp_2 &= M(\wp_0+\wp_1)-M(\wp_0),\\ &\vdots\\ \wp_{n+1} &= M\left(\sum_{j=0}^{n}\wp_j\right)-M\left(\sum_{j=0}^{n-1}\wp_j\right), \quad n=0,1,2,\ldots.\end{aligned} \tag{2.12}$$

Consequently, it follows that

$$\wp_1+\wp_2+\cdots+\wp_{n+1} = M(\wp_0+\wp_1+\wp_2+\cdots+\wp_n),$$

and

$$\wp = c+\sum_{i=1}^{\infty}\wp_i. \tag{2.13}$$

It is noted that $\wp$ is approximated by

$$U_n = (\wp_0+\wp_1+\wp_2+\cdots+\wp_n),$$

and

$$\lim_{x\to\infty} U_n = \wp.$$

For $n=0$,

$$\wp \approx U_0 = \wp_0 = c = \gamma. \tag{2.14}$$

From (2.5), it can easily be computed as:

$$H(\wp_0) = 0.$$

Using (2.8), we get

$$\begin{aligned}\wp_1 = M(\wp_0) &= \frac{H(\wp_0)}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\gamma}{2}\right)+\Upsilon'(\wp_0)\right)}\\ &\quad+\frac{\Upsilon(\gamma)-\gamma}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\gamma}{2}\right)+\Upsilon'(\wp_0)\right)}\\ &= \frac{\Upsilon(\gamma)-\gamma}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\gamma}{2}\right)+\Upsilon'(\wp_0)\right)}.\end{aligned}$$

For $n=1$,

$$\begin{aligned}\wp \approx U_1 &= \wp_0+\wp_1 = \wp_0+M(\wp_0)\\ &= \gamma+\frac{\Upsilon(\gamma)-\gamma}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\gamma}{2}\right)+\Upsilon'(\wp_0)\right)}.\end{aligned}$$

Using (2.14), we have

$$\wp = \frac{\Upsilon(\gamma) - \gamma\Upsilon'(\gamma)}{1 - \Upsilon'(\gamma)}. \tag{2.15}$$

This fixed point formulation is used to suggest the following algorithm for solving the nonlinear equation $\Lambda(\wp) = 0$.

Algorithm 2.1. *For a given $\wp_0$ (initial guess), approximate solution $\wp_{n+1}$ is computed by the following iterative scheme*

$$\wp_{n+1} = \frac{\Upsilon(\wp_n) - \gamma\Upsilon'(\wp_n)}{1 - \Upsilon'(\wp_n)}. \tag{2.16}$$

Kang et al. [9] developed this algorithm and proved that Algorithm 2.1 has quadratic convergence.

Form (2.5) and (2.8), we have

$$\begin{aligned} H(\wp_0 + \wp_1) &= \Upsilon(\wp_0 + \wp_1) - \Upsilon(\gamma) \\ &\quad - \frac{1}{6}(\wp_0 + \wp_1 - \gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0 + \wp_1 + \gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right). \end{aligned}$$

Thus

$$\begin{aligned} \wp_1 + \wp_2 &= M(\wp_0 + \wp_1) \\ &= \frac{H(\wp_0 + \wp_1)}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)} + \frac{\Upsilon(\gamma) - \gamma}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)} \\ &= \frac{1}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)} \\ &\quad \times \left(\Upsilon(\wp_0 + \wp_1) - \Upsilon(\gamma) - \frac{1}{6}(\wp_0 + \wp_1 - \gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0 + \wp_1 + \gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)\right) \\ &\quad + \frac{\Upsilon(\gamma) - \gamma}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)} \\ &= \frac{1}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)} \\ &\quad \times \left(\Upsilon(\wp_0 + \wp_1) - \frac{1}{6}(\wp_0 + \wp_1 - \gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0 + \wp_1 + \gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right) - \gamma\right). \end{aligned}$$

For $n = 2$,

$$\begin{aligned} \wp \approx U_2 &= \wp_0 + \wp_1 + \wp_2 = c + M(\wp_0 + \wp_1) \\ &= \gamma + \frac{1}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right)} \\ &\quad \times \left(\Upsilon(\wp_0 + \wp_1) - \frac{1}{6}(\wp_0 + \wp_1 - \gamma)\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0 + \wp_1 + \gamma}{2}\right) + \Upsilon'(\wp_0 + \wp_1)\right) - \gamma\right) \\ &= \frac{1}{1 - \frac{1}{6}\left(\Upsilon'(\gamma) + 4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right) + \Upsilon'(\wp_0)\right)} \end{aligned}$$

$$\times\left(\Upsilon(\wp_0+\wp_1)-\frac{1}{6}(\wp_0+\wp_1)\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1)\right)\right).$$

Take

$$\begin{aligned}\wp_0+\wp_1=v&=\frac{\Upsilon(\gamma)-\gamma\Upsilon'(\gamma)}{1-\Upsilon'(\gamma)}\\&=\frac{\Upsilon(v)-\frac{1}{6}v\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{v+\gamma}{2}\right)+\Upsilon'(v)\right)}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{v+\gamma}{2}\right)+\Upsilon'(v)\right)}.\end{aligned}$$

This relation yields the following two-step method for solving the nonlinear equation $\Lambda(\wp)=0$.

Algorithm 2.2. *For a given initial guess $\wp_0$, the approximated solution $\wp_{n+1}$ can be computed by the following iterative schemes:*

$$v_n=\frac{\Upsilon(\wp_n)-\wp_n\Upsilon'(\wp_n)}{1-\Upsilon'(\wp_n)},\tag{2.17}$$

and

$$\wp_{n+1}=\frac{\Upsilon(v_n)-\frac{1}{6}v_n\left(\Upsilon'(\wp_n)+4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right)+\Upsilon'(v_n)\right)}{1-\frac{1}{6}\left(\Upsilon'(\wp_\wp)+4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right)+\Upsilon'(v_n)\right)},\tag{2.18}$$

where $n=0,1,2,\ldots$.

It is noted that

$$\begin{aligned}\wp_0+\wp_1+\wp_2=w&\\&=\frac{1}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right)+\Upsilon'(\wp_0)\right)}\Bigg[\Upsilon(\wp_0+\wp_1)\\&\quad-\frac{1}{6}(\wp_0+\wp_1)\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1)\Bigg].\end{aligned}\tag{2.19}$$

Using (2.5) and (2.8), we can write

$$\begin{aligned}H(\wp_0+\wp_1+\wp_2)&=\Upsilon(\wp_0+\wp_1+\wp_2)-\Upsilon(\gamma)\\&\quad-\frac{1}{6}(\wp_0+\wp_1+\wp_2-\gamma)\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right),\end{aligned}$$

and

$$\begin{aligned}\wp_1+\wp_2+\wp_3&=M(\wp_0+\wp_1+\wp_2)\\&=\frac{H(\wp_0+\wp_1+\wp_2)}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}\\&\quad+\frac{\Upsilon(\gamma)-\gamma}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}\\&=\frac{1}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}\end{aligned}$$

$$\times\left(\Upsilon(\wp_0+\wp_1+\wp_2)-\Upsilon(\gamma)-\frac{1}{6}(\wp_0+\wp_1+\wp_2-\gamma)\Upsilon'(\gamma)\right.$$
$$\left.+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)$$
$$+\frac{\Upsilon(\gamma)-\gamma}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}$$
$$=\frac{1}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}\left[\Upsilon(\wp_0+\wp_1+\wp_2)\right.$$
$$\left.-\frac{1}{6}(\wp_0+\wp_1+\wp_2-\gamma)\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)-\gamma\right].$$

For $n=3$,

$$\wp\approx U_3=\wp_0+\wp_1+\wp_2+\wp_3=\wp_0+M(\wp_0+\wp_1+\wp_2).$$
$$=\gamma+\frac{1}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}\left[\Upsilon(\wp_0+\wp_1+\wp_2)\right.$$
$$\left.-\frac{1}{6}(\wp_0+\wp_1+\wp_2-\gamma)\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)-\gamma\right]$$
$$=\frac{1}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)}$$
$$\times\left[\Upsilon(\wp_0+\wp_1+\wp_2)-\frac{1}{6}(\wp_0+\wp_1+\wp_2)\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{\wp_0+\wp_1+\wp_2+\gamma}{2}\right)+\Upsilon'(\wp_0+\wp_1+\wp_2)\right)\right].$$

Using (2.19), we have

$$\wp=\frac{\Upsilon(w)-\frac{1}{6}w\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{w+\gamma}{2}\right)+\Upsilon'(w)\right)}{1-\frac{1}{6}\left(\Upsilon'(\gamma)+4\Upsilon'\left(\frac{w+\gamma}{2}\right)+\Upsilon'(w)\right)}.$$

Using this relation, we suggest the following three-step method for solving nonlinear Eq (2.1).

Algorithm 2.3. *For a given initial guess $\wp_0$, compute the approximate solution $\wp_{n+1}$ by the following iterative scheme.*

$$v_n=\frac{\Upsilon(\wp_n)-\wp_n\Upsilon'(\wp_n)}{1-\Upsilon'(\wp_n)},$$

$$w_n=\frac{\Upsilon(v_n)}{1-\frac{1}{6}\left(\Upsilon'(\wp_\wp)+4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right)+\Upsilon'(v_n)\right)}$$
$$-\frac{\frac{1}{6}v_n\left(\Upsilon'(\wp_n)+4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right)+\Upsilon'(v_n)\right)}{1-\frac{1}{6}\left(\Upsilon'(\wp_n)+4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right)+\Upsilon'(v_n)\right)},\tag{2.20}$$

and

$$\wp_{n+1}=\frac{\Upsilon(w_n)}{1-\frac{1}{6}\left(\Upsilon'(\wp_\wp)+4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)+\Upsilon'(w_n)\right)}$$

$$-\frac{\frac{1}{6}w_n\left(\Upsilon'(\wp_n)+4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)+\Upsilon'(w_n)\right)}{1-\frac{1}{6}\left(\Upsilon'(\wp_n)+4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)+\Upsilon'(w_n)\right)}, \tag{2.21}$$

where $n = 0, 1, 2, \ldots$.

3. Convergence analysis of proposed iterative methods

This section comprises convergence analysis of proposed Algorithms 2.2 and 2.3. It is shown that these methods are third and fourth-order convergent, respectively.

Theorem 3.1. *Let* $I \subset \mathbb{R}$ *be an open interval and* $\Lambda : I \to \mathbb{R}$ *is differential function. If* $\beta \in I$ *be the simple root of* $\Lambda(\wp) = 0$ *and* $\wp_0$ *is sufficiently close to* β *then multi-step method defined by Algorithm 2.2 has third order of convergence.*

Proof. Let β be the root of nonlinear equation $\Lambda(\wp) = 0$, or equivalently $\wp = \Upsilon(\wp)$. Let e_n and e_{n+1} be the errors at *nth* and $(n+1)$ iterations, respectively.

Now, expanding $\Upsilon(\wp)$ and $\Upsilon'(\wp)$ by using Taylor series about β, we have

$$\Upsilon(\wp_n) = \beta + e_n\Upsilon'(\beta) + \frac{e_n^2\Upsilon''(\beta)}{2} + \frac{e_n^3\Upsilon'''(\beta)}{6} + O\left(e_n^4\right), \tag{3.1}$$

and

$$\Upsilon'(\wp_n) = \Upsilon'(\beta) + e_n\Upsilon''(\beta) + \frac{e_n^2\Upsilon'''(\beta)}{2} + \frac{e_n^3\Upsilon^{iv}(\beta)}{6} + O\left(e_n^4\right). \tag{3.2}$$

Thus, we have

$$\begin{aligned}\Upsilon(\wp_n) - \wp_n\Upsilon'(\wp_n) &= \beta - \beta\Upsilon'(\beta) - \beta\Upsilon''(\beta)e_n - \frac{1}{2}\left(\Upsilon''(\beta) + \beta\Upsilon'''(\beta)\right)e_n^2 - \frac{1}{6}\left(2\Upsilon'''(\beta) + \beta\Upsilon^{iv}(\beta)\right)e_n^3 \\ &\quad + O\left(e_n^4\right),\end{aligned}$$

$$1 - \Upsilon'(\wp_n) = 1 - \Upsilon'(\beta) - e_n\Upsilon''(\beta) - \frac{e_n^2\Upsilon'''(\beta)}{2} - \frac{e_n^3\Upsilon^{iv}(\beta)}{6} + O\left(e_n^4\right),$$

and

$$\frac{\Upsilon(\wp_n) - \wp_n\Upsilon'(\wp_n)}{1 - \Upsilon'(\wp_n)} = \beta + \frac{\Upsilon''(\beta)}{2(-1+\Upsilon'(\beta))}e_n^2 - \frac{2\Upsilon'''(\beta) - 2\Upsilon'''(\beta)\Upsilon'(\beta) + 3\Upsilon''^2(\beta)}{6(-1+\Upsilon'(\beta))^2}e_n^3 + O\left(e_n^4\right).$$

From (2.16), we have

$$\begin{aligned}\wp_{n+1} &= \beta + \frac{\Upsilon''(\beta)}{2(-1+\Upsilon'(\beta))}e_n^2 - \frac{2\Upsilon'''(\beta) - 2\Upsilon'''(\beta)\Upsilon'(\beta) + 3\Upsilon''^2(\beta)}{6(-1+\Upsilon'(\beta))^2}e_n^3 \\ &\quad + \frac{1}{12(-1+\Upsilon'(\beta))^3}\Big(2\Upsilon^{iv}(\beta) - 4\Upsilon^{iv}(\beta)\Upsilon'(\beta) + 2\Upsilon^{iv}(\beta)\Upsilon'^2(\beta) \\ &\quad + 7\Upsilon''(\beta)\Upsilon'''(\beta) - 7\Upsilon''(\beta)\Upsilon'''(\beta)\Upsilon'(\beta) + 6\Upsilon''^3(\beta)\Big)e_n^4 + O\left(e_n^5\right).\end{aligned}$$

Using (2.17), we obtain

$$v_n = \beta + \frac{\Upsilon''(\beta)}{2(-1+\Upsilon'(\beta))}e_n^2 - \frac{2\Upsilon'''(\beta) - 2\Upsilon'''(\beta)\Upsilon'(\beta) + 3\Upsilon''^2(\beta)}{6(-1+\Upsilon'(\beta))^2}e_n^3$$
$$+ \frac{1}{12(-1+\Upsilon'(\beta))^3}\left(2\Upsilon^{iv}(\beta) - 4\Upsilon^{iv}(\beta)\Upsilon'(\beta) + 2\Upsilon^{iv}(\beta)\Upsilon'^2(\beta)\right.$$
$$\left.+7\Upsilon''(\beta)\Upsilon'''(\beta) - 7\Upsilon''(\beta)\Upsilon'''(\beta)\Upsilon'(\beta) + 6\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right).$$

Expanding $\Upsilon(v_n)$ in terms of Taylor series about β, we get

$$\Upsilon(v_n) = \beta + \frac{1}{2(-1+\Upsilon'(\beta))}\Upsilon'(\beta)\Upsilon''(\beta)e_n^2 - \frac{1}{6(-1+\Upsilon'(\beta))^2}\Upsilon'(\beta)\left(2\Upsilon'''(\beta) - 2\Upsilon'''(\beta)\Upsilon'(\beta)\right.$$
$$\left.+3\Upsilon''^2(\beta)\right)e_n^3 + \frac{1}{12(-1+\Upsilon'(\beta))^3}\left(4\Upsilon^{iv}(\beta)\Upsilon'(\beta) - 8\Upsilon^{iv}(\beta)\Upsilon'^2(\beta)\right.$$
$$+4\Upsilon^{iv}(\beta)\Upsilon'^3(\beta) - 14\Upsilon''(\beta)\Upsilon'''(\beta)\Upsilon'^2(\beta) + 14\Upsilon''(\beta)\Upsilon'''(\beta)\Upsilon'(\beta)$$
$$\left.+15\Upsilon''^3(\beta)\Upsilon'(\beta) - 3\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right). \tag{3.3}$$

Expanding $\Upsilon'(v_n)$ in terms of Taylor series about β, we get

$$\Upsilon'(v_n) = \Upsilon'(\beta) + \frac{1}{2(-1+\Upsilon'(\beta))}\Upsilon''(\beta)e_n^2 + \frac{1}{6(-1+\Upsilon'(\beta))^2}\left(-2\Upsilon'''(\beta) + 2\Upsilon'''(\beta)\Upsilon'(\beta)\right.$$
$$\left.-3\Upsilon''^2(\beta)\right)e_n^3 + \frac{1}{24(-1+\Upsilon'(\beta))^3}\Upsilon''(\beta)\left(4\Upsilon^{iv}(\beta) - 8\Upsilon^{iv}(\beta)\Upsilon'(\beta)\right.$$
$$+4\Upsilon^{iv}(\beta)\Upsilon'^2(\beta) + 11\Upsilon''(\beta)\Upsilon'''(\beta) - 11\Upsilon''(\beta)\Upsilon'''(\beta)\Upsilon'(\beta)$$
$$\left.+12\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right). \tag{3.4}$$

Expanding $\Upsilon'\left(\frac{v_n+\wp_n}{2}\right)$ in terms of Taylor series about β, we get

$$\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) = \Upsilon'(\beta) + \frac{\Upsilon''(\beta)}{2}e_n + \frac{1}{8(-1+\Upsilon'(\beta))}\left(2\Upsilon''^2(\beta) + \Upsilon'''(\beta)\Upsilon'(\beta) - \Upsilon'''(\beta)\right)e_n^2$$
$$+ \frac{1}{48(-1+\Upsilon'(\beta))^2}\left(-14\Upsilon''(\beta)\Upsilon'''(\beta) + 14\Upsilon''(\beta)\Upsilon'''(\beta)\Upsilon'(\beta)\right.$$
$$\left.+\Upsilon^{iv}(\beta) - 12\Upsilon''^3(\beta) - 2\Upsilon^{iv}(\beta)\Upsilon'(\beta) + \Upsilon^{iv}(\beta)\Upsilon'^2(\beta)\right)e_n^3 + \frac{1}{96(-1+\Upsilon'(\beta))^3}$$
$$\left(11\Upsilon^{iv}(\beta)\Upsilon''(\beta) - 22\Upsilon^{iv}(\beta)\Upsilon'(\beta)\Upsilon''(\beta) + 11\Upsilon^{iv}(\beta)\Upsilon'^2(\beta)\Upsilon''(\beta)\right.$$
$$+37\Upsilon''^2(\beta)\Upsilon'''(\beta) - 37\Upsilon''^2(\beta)\Upsilon'''(\beta)\Upsilon'(\beta) + 24\Upsilon''^4(\beta) + 8\Upsilon'''^2(\beta)$$
$$\left.-16\Upsilon'''^3(\beta)\Upsilon'(\beta) + 8\Upsilon'^2(\beta)\Upsilon'''^2(\beta)\right)e_n^4 + O\left(e_n^5\right). \tag{3.5}$$

From (3.2), (3.4) and (3.5), we have

$$\frac{1}{6}\left[\Upsilon'(v_n) + 4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right]$$

$$= \Upsilon'(\beta) + \frac{1}{2}\Upsilon''(\beta)e_n + \frac{1}{12}\Upsilon'''(\beta)e_n^2 + \frac{1}{12(-1+\Upsilon'(\beta))}5\Upsilon''^2(\beta)e_n^2$$
$$+ \frac{1}{36}\Upsilon^{iv}(\beta)e_n^3 + \frac{1}{36(-1+\Upsilon'(\beta))^2}5\Upsilon''(\beta)\left(2\Upsilon'(\beta)\Upsilon'''(\beta) - 2\Upsilon'''(\beta) - 3\Upsilon''^2(\beta)\right)e_n^3$$
$$+ \frac{1}{144}\Upsilon^{v}(\beta)e_n^4 + \frac{1}{144(-1+\Upsilon'(\beta))^3}5\Upsilon''(\beta)\left(3\Upsilon^{iv}(\beta) + 12\Upsilon''^3(\beta) - 6\Upsilon'(\beta)\Upsilon^{iv}(\beta)\right.$$
$$\left.+3\Upsilon'^2(\beta)\Upsilon^{iv}(\beta) + 14\Upsilon''(\beta)\Upsilon'''(\beta) - 14\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right)e_n^4 + O\left(e_n^5\right). \quad (3.6)$$

Now,

$$\frac{v_n}{6}\left[\Upsilon'(v_n) + 4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right]$$
$$= \beta\Upsilon'(\beta) + \frac{1}{2}\beta\Upsilon''(\beta)e_n + \frac{1}{12}\beta\Upsilon'''(\beta)e_n^2 + \frac{1}{12(-1+\Upsilon'(\beta))}5\beta\Upsilon''^2(\beta)e_n^2$$
$$+ \frac{1}{36}\beta\Upsilon^{iv}(\beta)e_n^3 + \frac{1}{36(-1+\Upsilon'(\beta))^2}5\beta\Upsilon''(\beta)\left(2\Upsilon'(\beta)\Upsilon'''(\beta) - 2\Upsilon'''(\beta) - 3\Upsilon''^2(\beta)\right)e_n^3$$
$$+ \frac{1}{144}\beta\Upsilon^{v}(\beta)e_n^4 + \frac{1}{144(-1+\Upsilon'(\beta))^3}5\beta\Upsilon''(\beta)\left(3\Upsilon^{iv}(\beta) + 12\Upsilon''^3(\beta) - 6\Upsilon'(\beta)\Upsilon^{iv}(\beta)\right.$$
$$\left.+3\Upsilon'^2(\beta)\Upsilon^{iv}(\beta) + 14\Upsilon''(\beta)\Upsilon'''(\beta) - 14\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right)e_n^4 + \frac{1}{2(-1+\Upsilon'(\beta))}\Upsilon'(\beta)\Upsilon''(\beta)e_n^2$$
$$+ \frac{1}{12(-1+\Upsilon'(\beta))^2}\left(\Upsilon'(\beta)2\Upsilon'(\beta)\Upsilon'''(\beta) - 2\Upsilon'''(\beta) - 3\Upsilon''^2(\beta)\right)e_n^4$$
$$+ \frac{1}{24(-1+\Upsilon'(\beta))}\Upsilon''(\beta)\Upsilon'''(\beta)e_n^4 + \frac{1}{24(-1+\Upsilon'(\beta))^2}5\Upsilon''^3(\beta)e_n^4 + O\left(e_n^5\right). \quad (3.7)$$

Using (3.3) and (3.7), we have

$$\Upsilon(v_n) - \frac{v_n}{6}\left[\Upsilon'(v_n) + 4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right]$$
$$= -\beta(-1+\Upsilon'(\beta)) - \frac{1}{2}\beta\Upsilon''(\beta)e_n - \frac{1}{12}\beta\Upsilon'''(\beta)e_n^2$$
$$+ \frac{1}{12(-1+\Upsilon'(\beta))}5\beta\Upsilon''^2(\beta)e_n^2 + \frac{1}{36}\beta\Upsilon^{iv}(\beta)e_n^3$$
$$+ \frac{1}{36(-1+\Upsilon'(\beta))^2}5\beta\Upsilon''(\beta)\left(2\Upsilon'(\beta)\Upsilon'''(\beta) - 2\Upsilon'''(\beta) - 3\Upsilon''^2(\beta)\right)e_n^3$$
$$+ \frac{1}{144}\beta\Upsilon^{v}(\beta)e_n^4 + \frac{1}{144(-1+\Upsilon'(\beta))^3}5\beta\Upsilon''(\beta)\left(3\Upsilon^{iv}(\beta) + 12\Upsilon''^3(\beta)\right.$$
$$\left.-6\Upsilon'(\beta)\Upsilon^{iv}(\beta) + 3\Upsilon'^2(\beta)\Upsilon^{iv}(\beta) + 14\Upsilon''(\beta)\Upsilon'''(\beta) - 14\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right)e_n^4$$
$$- \frac{1}{(-1+\Upsilon'(\beta))}c_2^2e_n^3 + \frac{1}{12(-1+\Upsilon'(\beta))^2}\Upsilon''(\beta)(2\Upsilon'(\beta)\Upsilon'''(\beta) - 2\Upsilon'''(\beta)$$
$$\left.-3\Upsilon''^2(\beta)\right)e_n^4 - \frac{1}{24(-1+\Upsilon'(\beta))^2}\Upsilon''(\beta)\Upsilon'''(\beta)e_n^4 - \frac{1}{24(-1+\Upsilon'(\beta))^2}5\Upsilon''^3(\beta)e_n^4 + O\left(e_n^5\right). \quad (3.8)$$

Using (3.6), we have

$$1 - \frac{1}{6}\left[\Upsilon'(v_n) + 4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right]$$

$$= 1 - \Upsilon'(\beta) - \frac{1}{2}\Upsilon''(\beta)e_n - \frac{1}{12}\Upsilon'''(\beta)e_n^2 - \frac{1}{12(-1+\Upsilon'(\beta))}5\Upsilon''^2(\beta)e_n^2$$
$$- \frac{1}{36}\Upsilon^{iv}(\beta)e_n^3 - \frac{1}{36(-1+\Upsilon'(\beta))^2}5\Upsilon''(\beta)\left(2\Upsilon'(\beta)\Upsilon'''(\beta) - 2\Upsilon'''(\beta) - 3\Upsilon''^2(\beta)\right)e_n^3$$
$$- \frac{1}{144}\Upsilon^{v}(\beta)e_n^4 - \frac{1}{144(-1+\Upsilon'(\beta))^3}5\Upsilon''(\beta)\left(3\Upsilon^{iv}(\beta) + 12\Upsilon''^3(\beta) - 6\Upsilon'(\beta)\Upsilon^{iv}(\beta)\right.$$
$$\left.+3\Upsilon'^2(\beta)\Upsilon^{iv}(\beta) + 14\Upsilon''(\beta)\Upsilon'''(\beta) - 14\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right)e_n^4 + O\left(e_n^5\right). \quad (3.9)$$

Dividing (3.8) and (3.9), we have

$$\frac{\Upsilon(v_n) - \frac{v_n}{6}\left[\Upsilon'(v_n) + 4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right]}{1 - \frac{1}{6}\left[\Upsilon'(v_n) + 4\Upsilon'\left(\frac{v_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right]}$$
$$= \beta + \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon''^2(\beta)e_n^3 + \frac{1}{144(-1+\Upsilon'(\beta))^3}\left(30\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon''(\beta)\right.$$
$$\left.-30\Upsilon''(\beta)\Upsilon'''(\beta) - 24\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right).$$

Using (2.18), we have

$$\wp_{n+1} = \beta + \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon''^2(\beta)e_n^3 + \frac{1}{144(-1+\Upsilon'(\beta))^3}\left(30\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right.$$
$$\left.-30\Upsilon''(\beta)\Upsilon'''(\beta) - 24\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right). \quad (3.10)$$

Therefore,

$$e_{n+1} = \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon''^2(\beta)e_n^3 + \frac{1}{144(-1+\Upsilon'(\beta))^3}\left(30\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right.$$
$$\left.-30\Upsilon''(\beta)\Upsilon'''(\beta) - 24\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right).$$

This shows that Algorithm 2.2 has third-order of convergence. □

Theorem 3.2. *Let $I \subset \mathbb{R}$ be an open interval and $\Lambda : I \to \mathbb{R}$ is differential function. If $\beta \in I$ be the simple root of $\Lambda(\wp) = 0$ and $\wp_0$ is sufficiently close to β then multi-step method defined by Algorithm 2.3 has fourth order of convergence.*

Proof. From (3.10), we have

$$w_n = \beta + \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon''^2(\beta)e_n^3 + \frac{1}{144(-1+\Upsilon'(\beta))^3}\left(30\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right.$$
$$\left.-30\Upsilon''(\beta)\Upsilon'''(\beta) - 24\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right).$$

Expanding $\Upsilon(w_n)$, in terms of Taylor's series

$$\Upsilon(w_n) = \beta + \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon'(\beta)\Upsilon''^2(\beta)e_n^3 + \frac{1}{144(-1+\Upsilon'(\beta))^3}\Upsilon'(\beta)\left(30\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta)\right.$$
$$\left.-30\Upsilon''(\beta)\Upsilon'''(\beta) - 24\Upsilon''^3(\beta)\right)e_n^4 + O\left(e_n^5\right). \quad (3.11)$$

Expanding $\Upsilon'(w_n)$, in terms of Taylor's series

$$\Upsilon'(w_n) = \Upsilon'(\beta) + \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon'(\beta)\Upsilon''^3(\beta)e_n^3 + \frac{1}{144(-1+\Upsilon'(\beta))^3}\Upsilon'(\beta)\Big(30\Upsilon'(\beta)\Upsilon''^2(\beta)\Upsilon'''(\beta) \\ -30\Upsilon''^2(\beta)\Upsilon'''(\beta) - 24\Upsilon''^4(\beta)\Big)e_n^4 + O\left(e_n^5\right). \tag{3.12}$$

Expanding $\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)$ in terms of Taylor's series

$$\Upsilon'\left(\frac{w_n+\wp_n}{2}\right) = \Upsilon'(\beta) + \frac{1}{2}\Upsilon''(\beta)e_n + \frac{1}{4(-1+\Upsilon'(\beta))^2}\Upsilon''^3(\beta)e_n^3 \\ + \frac{1}{48(-1+\Upsilon'(\beta))^3}\Upsilon''(\beta)(15\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta) \\ -15\Upsilon''(\beta)\Upsilon'''(\beta) - 12\Upsilon''^3\Big)(\beta)e_n^4 + O\left(e_n^5\right). \tag{3.13}$$

Using (3.2), (3.12) and (3.13), we have

$$\frac{1}{6}\left[\Upsilon'(w_n) + 4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right] \\ = \Upsilon'(\beta) + \frac{1}{2}\Upsilon''(\beta)e_n + \frac{1}{12}\Upsilon'''(\beta)e_n^2 + \frac{1}{24}\Upsilon^{iv}(\beta)e_n^3 + \frac{1}{24(-1+\Upsilon'(\beta))^2}5\Upsilon''^3(\beta)e_n^3 \\ + \frac{1}{144}\Upsilon^{v}(\beta)e_n^4 + \frac{1}{36(-1+\Upsilon'(\beta))^3}5\Upsilon''(\beta)(15\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta) - 15\Upsilon''(\beta)\Upsilon'''(\beta) \\ -12\Upsilon''^3(\beta)\Big)e_n^4 + O\left(e_n^5\right). \tag{3.14}$$

Now,

$$\frac{w_n}{6}\left[\Upsilon'(w_n) + 4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right] \\ = \beta\Upsilon'(\beta) + \frac{1}{2}\beta\Upsilon''(\beta)e_n + \frac{1}{12}\beta\Upsilon'''(\beta)e_n^2 + \frac{1}{24}\beta\Upsilon^{iv}(\beta)e_n^3 + \frac{1}{24(-1+\Upsilon'(\beta))^2}5\beta\Upsilon''^3(\beta)e_n^3 \\ + \frac{1}{144}\beta\Upsilon^{v}(\beta)e_n^4 + \frac{1}{36(-1+\Upsilon'(\beta))^3}5\Upsilon''(\beta)(15\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta) \\ -15\Upsilon''(\beta)\Upsilon'''(\beta) - 12\Upsilon''^3(\beta)\Big)e_n^4 + \frac{1}{8(-1+\Upsilon'(\beta))^2}\Upsilon''^3(\beta)e_n^4 + O\left(e_n^5\right). \tag{3.15}$$

Using (3.11) and (3.15), we have

$$\Upsilon(w_n) - \frac{w_n}{6}\left[\Upsilon'(w_n) + 4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right) + \Upsilon'(\wp_n)\right] \tag{3.16}$$

$$= \beta - \beta\Upsilon'(\beta) - \frac{1}{2}\beta\Upsilon''(\beta)e_n - \frac{1}{12}\beta\Upsilon'''(\beta)e_n^2 - \frac{1}{24}\beta\Upsilon^{iv}(\beta)e_n^3 \\ - \frac{1}{24(-1+\Upsilon'(\beta))^2}5\beta\Upsilon''^3(\beta)e_n^3 - \frac{1}{144}\beta\Upsilon^{v}(\beta)e_n^4 \\ - \frac{1}{36(-1+\Upsilon'(\beta))^3}5\Upsilon''(\beta)(15\Upsilon'(\beta)\Upsilon''(\beta)\Upsilon'''(\beta) - 15\Upsilon''(\beta)\Upsilon'''(\beta)$$

$$-12\Upsilon''^{3}(\beta)\Big)e_n^4-\frac{1}{8\left(-1+\Upsilon'(\beta)\right)^2}\Upsilon''^{3}(\beta)\,e_n^4+O\left(e_n^5\right). \tag{3.17}$$

Using (3.14), we have

$$\begin{aligned}&1-\frac{1}{6}\left[\Upsilon'(w_n)+4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)+\Upsilon'(\wp_n)\right]\\&=1-\Upsilon'(\beta)-\frac{1}{2}\Upsilon''(\beta)\,e_n-\frac{1}{12}\Upsilon'''(\beta)\,e_n^2-\frac{1}{24}\Upsilon^{iv}(\beta)\,e_n^3+O\left(e_n^4\right).\end{aligned} \tag{3.18}$$

Dividing (3.17) and (3.18), we have

$$\frac{\Upsilon(w_n)-\frac{w_n}{6}\left[\Upsilon'(w_n)+4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)+\Upsilon'(\wp_n)\right]}{1-\frac{1}{6}\left[\Upsilon'(w_n)+4\Upsilon'\left(\frac{w_n+\wp_n}{2}\right)+\Upsilon'(\wp_n)\right]}=\beta+\frac{1}{8\left(-1+\Upsilon'(\beta)\right)^2}\Upsilon''^{3}(\beta)\,e_n^4+O\left(e_n^5\right).$$

Using (2.21), we have

$$\wp_{n+1}=\beta+\frac{1}{8\left(-1+\Upsilon'(\beta)\right)^2}\Upsilon''^{3}(\beta)\,e_n^4+O\left(e_n^5\right).$$

Therefore,

$$e_{n+1}=\frac{1}{\left(-1+\Upsilon'(\beta)\right)^2}\Upsilon''^{3}(\beta)\,e_n^4+O\left(e_n^5\right).$$

This shows that Algorithm 2.3 has fourth order of convergence. □

4. Numerical examples and comparison results

This section elaborates on the efficacy of algorithms introduced in this paper with the support of examples. We obtain an estimated simple root rather than the exact based on the exactness of the computer. We utilize $\varepsilon = 10^{-5}$, for computational work we use the following stopping criteria $|\wp_{n+1}-\wp_n|<\varepsilon$.

We make a comparative representation Newton method (NM), Halley method (HM), Algorithms 2A [12], 2B [12] and 2C [12] with Algorithms 2.2 and 2.3. In tables, we also displayed the number of iterations (IT), the approximate root $\wp_n$ and the value of $\Lambda(\wp_n)$.

We solved all test problems and calculated the CPU time consumptions in second with the aid of the computer software Maple 17.

Example 4.1. *For* $\Lambda(\wp)=\wp^3-10, \Upsilon(\wp)=\sqrt{\frac{10}{\wp}}$ *and* $\wp_0=1.5$.

Methods	*IT*	$\wp_n$	$\Lambda(\wp_n)$	$\delta=\mid\wp_n-\wp_{n-1}\mid$	*CPU Time*
NM [12]	5	2.15443469003191858890	$4.85518e^{-13}$	$2.740789522304e^{-7}$	0.218
HM [12]	5	2.15443469003191858890	$4.85518e^{-13}$	$2.740789522304e^{-7}$	0.453
Algorithm 2A [12]	4	2.15443469003188834557	$-3.7048e^{-15}$	$4.78781787078e^{-8}$	0.390
Algorithm 2B [12]	3	2.15443469003188837218	$1.0e^{-18}$	$3.6257326037e^{-9}$	0.312
Algorithm 2C [12]	3	2.15443469003188837217	$-8.0e^{-19}$	$1.457e^{-16}$	0.140
Algorithm 2.2	3	2.15443469003188837216	$-2.2e^{-18}$	$4.4395190876e^{-9}$	0.500
Algorithm 2.3	3	2.15443469003188837217	$-8.0e^{-19}$	$2.645e^{-16}$	0.281

Example 4.2. *For* $\Lambda(\wp) = \cos(\wp) - \wp, \Upsilon(\wp) = \cos(\wp)$ *and* $\wp_0 = 1.7$.

Methods	*IT*	$\wp_n$	$\Lambda(\wp_n)$	$\delta = \mid \wp_n - \wp_{n-1} \mid$	*CPU Time*
NM [12]	4	0.7390851332151608761 4	$-3.9244e^{-16}$	$3.258805388731e^{-8}$	0.359
HM [12]	4	0.73908513321516087614	$3.9244e^{-16}$	$3.258805388731e^{-18}$	0.250
Algorithm 2A [12]	4	0.73908513321516087612	$-3.9240e^{-16}$	$3.258805388731e^{-8}$	0.578
Algorithm 2B [12]	3	0.73908513321516064166	$-1.0e^{-20}$	$8.63747112426e^{-9}$	0.203
Algorithm 2C [12]	3	0.73908513321516064166	$-1.0e^{-20}$	$3.72234e^{-15}$	0.187
Algorithm 2.2	3	0.73908513321516064166	$-1.0e^{-20}$	$6.04698535295e^{-9}$	0.218
Algorithm 2.3	3	0.73908513321516064168	$-4.0e^{-20}$	$1.41006e^{-15}$	0.062

Example 4.3. *For* $\Lambda(\wp) = (\wp - 1)^3 - 1, \Upsilon(\wp) = 1 + \sqrt{\frac{1}{(\wp-1)}}$ *and* $\wp_0 = 3.5$.

Methods	*IT*	$\wp_n$	$\Lambda(\wp_n)$	$\delta = \mid \wp_n - \wp_{n-1} \mid$	*CPU Time*
NM [12]	6	2.0000000000	$8.63270019e^{-11}$	$-0.53642860080240e - 5$	0.859
HM [12]	6	2.0000000000	$8.63270019e^{-11}$	$0.53642860080240e^{-5}$	0.531
Algorithm 2A [12]	5	1.9999999999	$-3.0e^{-19}$	$5.15552777e^{-11}$	0.500
Algorithm 2B [12]	3	2.0000000000	$1.95e^{-17}$	$-0.47082591839347e - 5$	0.734
Algorithm 2C [12]	3	1.9999999999	$-3.0e^{-19}$	$1.408550753e^{-11}$	0.468
Algorithm 2.2	3	2.0000000000	$3.3e^{-18}$	$0.25348368459034e^{-5}$	0.546
Algorithm 2.3	3	2.0000000000	$6.0e^{-19}$	$3.64102778e^{-11}$	0.828

Example 4.4. *For* $\Lambda(\wp) = e^{\wp^2+7\wp-30} - 1, \Upsilon(\wp) = \frac{1}{7}\left(30 - \wp^2\right)$ *and* $\wp_0 = 3.5$.

Methods	*IT*	$\wp_n$	$\Lambda(\wp_n)$	$\delta = \mid \wp_n - \wp_{n-1} \mid$	*CPU Time*
NM [12]	10	3.0000001961	$0.2550069197e^{-18}$	$0.1725678800759161e^{-3}$	2.031
HM [12]	10	3.0000001961	$0.2550069197e^{-5}$	$0.1725678800759161e^{-3}$	1.640
Algorithm 2A [12]	4	3.0000000000	0	$4.60291802e^{-11}$	1.015
Algorithm 2B [12]	3	3.0000000000	0	$1.7047845e^{-12}$	0.984
Algorithm 2C [12]	3	2.9999999999	0.9999999999	$1.0e^{-19}$	0.781
Algorithm 2.2	3	2.9999999999	$-2.00e^{-18}$	$1.7047846e^{-12}$	0.203
Algorithm 2.3	3	2.9999999999	$-2.00e^{-18}$	$1.0e^{-19}$	1.656

Example 4.5. *For* $\Lambda(\wp) = \sin^2(\wp) - \wp^2 + 1, \Upsilon(\wp) = \sin(\wp) + \frac{1}{\sin(\wp)+\wp}$ *and* $\wp_0 = 1$.

Methods	*IT*	$\wp_n$	$\Lambda(\wp_n)$	$\delta = \mid \wp_n - \wp_{n-1} \mid$	*CPU Time*
NM [12]	5	1.40449164821564703 49	$7.591622e^{-13}$	$6.247205954873e^{-7}$	4.500
HM [12]	5	1.40449164821564703 49	$7.591622e^{-13}$	$6.247205954873e^{-7}$	2.703
Algorithm 2A [12]	4	1.40449164821534122 69	$-2.1e^{-18}$	$1.8258042740e^{-9}$	1.734
Algorithm 2B [12]	3	1.40449164821534122 61	$-2.0e^{-19}$	$4.830530998e^{-10}$	1.656
Algorithm 2C [12]	3	1.40449164821534122 61	$-2.0e^{-19}$	$3.27e^{-17}$	1.546
Algorithm 2.2	3	1.40449164821534122 59	$3.8e^{-19}$	$3.039794094e^{-10}$	1.203
Algorithm 2.3	3	1.40449164821534122 60	$1.0e^{-19}$	$9.9e^{-18}$	1.015

Example 4.6. *For* $\Lambda(\wp) = e^{\wp} - 3\wp^2, \Upsilon(\wp) = \sqrt{\frac{e^{\wp}}{3}}$ *and* $\wp_0 = 0.8$.

Methods	*IT*	$\wp_n$	$\Lambda(\wp_n)$	$\delta = \mid \wp_n - \wp_{n-1} \mid$	*CPU Time*
NM [12]	4	0.91000757248870906	$-2.3e^{-18}$	$1.13400774851e^{-9}$	4.796
HM [12]	4	0.91000757248870906	$-2.3e^{-18}$	$1.13400774848e^{-9}$	3.015
Algorithm 2A [12]	3	0.91000757248844157	$7.959612e^{-13}$	$0.113206254671831e^{-5}$	2.281
Algorithm 2B [12]	3	0.91000757248870906	$-1.0e^{-19}$	$5.97372e^{-15}$	1.953
Algorithm 2C [12]	2	0.91000757248870906	$2.0e^{-19}$	$0.114201199652787e^{-5}$	1.828
Algorithm 2.2	3	0.91000757248870906	0	$5.89807e^{-15}$	1.484
Algorithm 2.3	2	0.91000757248870906	$2.0e^{-19}$	$0.113215100426461e^{-5}$	0.234

5. Conclusions

We have established two new algorithms of third and fourth-order convergence by using a modified decomposition technique for the coupled systems. We have discussed the convergence analysis of these newly established algorithms. With the help of test examples, the computational comparison has been made with well-known third and fourth-order convergent iterative methods. It has been observed from Examples 4.1 to 4.6 that our CPU time and accuracy are much better than the existing algorithms in some cases.

Acknowledgements

This research was supported by the Fundamental Fund of Khon Kaen University, Thailand.

Conflict of interest

The authors declare that they have no competing interests.

References

1. F. Ali, W. Aslam, K. Ali, M.A. Anwar, A. Nadeem, New family of iterative methods for solving nonlinear models, *Discrete Dyn. Nat. Soc.*, **2018** (2018), 9619680. https://doi.org/10.1155/2018/9619680
2. F. Ali, W. Aslam, I. Khalid, A. Nadeem, Iteration methods with an auxiliary function for nonlinear equations, *J. Math.*, **2020** (2020), 7356408. https://doi.org/10.1155/2020/7356408
3. G. Adomian, *Nonlinear stochastic systems theory and applications to physics*, Kluwer Academic Publishers, 1989.
4. S. Abbasbandy, Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method, *Appl. Math. Comput.*, **145** (2003), 887–893. https://doi.org/10.1016/S0096-3003(03)00282-0
5. C. Chun, Iterative methods improving Newton's method by the decomposition method, *Comput. Math. Appl.*, **50** (2005), 1559–1568. https://doi.org/10.1016/j.camwa.2005.08.022

6. M. T. Darvishi, A. Barati, A third-order Newton-type method to solve systems of nonlinear equations, *Appl. Math. Comput.*, **187** (2007), 630–635. https://doi.org/10.1016/j.amc.2006.08.080

7. V. Daftardar-Gejji, H. Jafari, An iterative method for solving nonlinear functional equations, *J. Math. Anal. Appl.*, **316** (2006), 753–763. https://doi.org/10.1016/j.jmaa.2005.05.009

8. J. H. He, A new iteration method for solving algebraic equations, *Appl. Math. Comput.*, **135** (2003), 81–84. https://doi.org/10.1016/S0096-3003(01)00313-7

9. S. M. Kang, A. Rafiq, Y. C. Kwun, A new second-order iteration method for solving nonlinear equations, *Abstr. Appl. Anal.*, **2013** (2013), 487062. https://doi.org/10.1155/2013/487062

10. M. A. Noor, Fifth-order convergent iterative method for solving nonlinear equations using quadrature formula, *JMCSA*, **4** (2018), 0974–0570.

11. A. Y. Ozban, Some new variants of Newton's method, *Appl. Math. Lett.*, **17** (2004), 677–682. https://doi.org/10.1016/S0893-9659(04)90104-8

12. G. Sana, M. A. Noor, K. I. Noor, Some multistep iterative methods for nonlinear equation using quadrature rule, *Int. J. Anal. Appl.*, **18** (2020), 920–938.

13. M. Saqib, M. Iqbal, Some multi-step iterative methods for solving nonlinear equations, *Open J. Math. Sci.*, **1** (2017), 25–33. https://doi.org/10.30538/oms2017.0003

14. S. Weerakoon, T. Fernando, A variant of Newton's method with accelerated third-order convergence, *Appl. Math. Lett.*, **13** (2000), 87–93. https://doi.org/10.1016/S0893-9659(00)00100-2

15. M. Heydari, S. M. Hosseini, G. B. Loghmani, Convergence of a family of third-order methods free from second derivatives for finding multiple roots of nonlinear equations, *World Appl. Sci. J.*, **11** (2010), 507–512.

16. M. Heydari, S. M. Hosseini, G. B. Loghmani, On two new families of iterative methods for solving nonlinear equations with optimal order, *Appl. Anal. Discrete Math.*, **5** (2011), 93–109. https://doi.org/10.2298/AADM110228012H

17. M. Heydari, G. B. Loghmani, Third-order and fourth-order iterative methods free from second derivative for finding multiple roots of nonlinear equations, *CJMS*, **3** (2014), 67–85.

18. M. M. Sehati, S. M. Karbassi, M. Heydari, G. B. Loghmani, Several new iterative methods for solving nonlinear algebraic equations incorporating homotopy perturbation method (HPM), *Int. J. Phys. Sci.*, **7** (2012), 5017–5025. https://doi.org/10.5897/IJPS12.279