J. Math. Computer Sci., 32 (2024), 94–108

Online: ISSN 2008-949X

Journal of Mathematics and Computer Science

Journal Homepage: www.isr-publications.com/jmcs

A proposed sixth order inverse polynomial method for the solution of non-linear physical models

S. E. Fadugba[a,b,c,*], Ismail Ibrahim[d], O. Adeyeye[e], A. A. Adeniji[f], M. C. Kekana[f], J. T. Okunlola[g]

[a] *Department of Physical Sciences, Mathematics Programme, Landmark University, Omu-Aran, Nigeria.*

[b] *Department of Mathematics, Ekiti State University, Ado Ekiti, 360001, Nigeria.*

[c] *Landmark University SDG 4: Quality Education Research Group, Omu-Aran, Nigeria.*

[d] *Department of Mathematics, Federal University, Dutse, Nigeria.*

[e] *School of Quantitative Sciences, Universiti Utara Malaysia (UUM), Kedah, Malaysia.*

[f] *Department of Mathematics, Tshwane University of Technology, Pretoria, South Africa.*

[g] *Department of Mathematical and Physical Sciences, Afe Babalola University, Ado Ekiti, Nigeria.*

Abstract

There are several nonlinear physical models emanated from science and technology that have always remained a challenge for numerical analysts and applied mathematicians. Various one-step numerical methods were developed to deal with these models, however, it requires the developed method to have consistency, stability, zero stability and convergence characteristics to handle the non-linearity in the model. This paper proposes a new sixth order inverse polynomial method (SOIPM) with a relative measure of stability for the solution of non-linear physical models with different flavors. Firstly, the properties of SOIPM are analyzed and investigated. Moreover, three illustrative non-linear physical models have been solved to measure the accuracy, computational performance, suitability and effectiveness of SOIPM. Furthermore, the results generated via SOIPM are compared with the existing method of the celebrated Runge-Kutta of order four (RK4) in the context of the exact value (EV). Finally, the absolute errors (ABEs) and final absolute errors (FABEs) incurred by SOIPM are computed and compared with that of RK4.

Keywords: Consistency, convergence, initial value problem, inverse polynomial method, linear stability, order of accuracy, rational interpolating polynomial, zero stability.

2020 MSC: 34A12, 65L05, 65L20, 65L70.

1. Introduction

First order initial value problems (IVPs) are known to arise in various models describing different real-life scenarios such as the learning theory model, population models, epidemic models, models describing falling bodies, models for mixture problems and chemical reactions, among many others [24]. The general form for first order IVPs considered in this article is given by the equation

$$y' = f(x, y), y(a) = y_0, x \in [a, b], -\infty < y < \infty, \tag{1.1}$$

*Corresponding author

Email address: `fadugba.sunday@lmu.edu.ng`, `sunday.fadugba@eksu.edu.ng` (S. E. Fadugba)

doi: 10.22436/jmcs.032.02.01

Received: 2022-08-22 *Revised:* 2023-06-11 *Accepted:* 2023-06-21

which will either have a unique solution, an infinite number of solutions, or no solution. In cases where the IVP fails to have a solution or the solution is difficult to obtain, numerical solutions are considered. These numerical solutions conventionally adopt methods such as Taylor series method and Runge-Kutta methods. However, due to the problem of low accuracy, researchers still venture till date to introduce numerical methods with improved accuracy than existing approaches, which is the main objective of this article. Several authors have also studied the solutions of IVPs in ODEs via developed and existing methods, see [2, 4, 6–13, 15, 19, 21, 25, 27], just to mention a few. Heydari et al. [18] implemented the piecewise spectral-variational iteration method (PSVIM) to solve the nonlinear Lane-Emden equations arising in mathematical physics and astrophysics. This method was based on a combination of Chebyshev interpolation and standard variational iteration method (VIM) and matching it to a sequence of sub-intervals. A numerical method for finding the solution of Duffing-harmonic oscillator was proposed by [17]. The approach was based on hybrid functions approximation. The properties of hybrid functions that consist of block-pulse and Chebyshev cardinal functions are also discussed. Heydari et al. [16] investigated the application of a new modification of the piecewise variational iteration method for simulating the solution of the strongly nonlinear oscillators. The proposed method was based on a combination of spectral method and variational iteration method. Tafakkori-Bafghi et al. [26] introduced an effective numerical method for solving two-point nonlinear boundary value problems. The proposed iterative scheme, called the Legendre-Picard iteration method was based on the Picard iteration technique, shifted Legendre polynomials and Legendre-Gauss quadrature formula. This article develops an inverse polynomial method of order six to solve first order initial value problems with better accuracy. Inverse polynomial methods for the solution of differential equations have been explored by various studies including [1, 20], where a fourth stage inverse polynomial scheme was developed, and [14] which introduced a fifth stage form of the inverse polynomial scheme. These highlighted studies extended the concept of Taylor series solution for first order initial value problems which was discussed by [3, 5]. The advantage of this new approach coined as the inverse polynomial method as an accurate approach for solving first order ODEs was discussed extensively by [22, 23], thus justifying the usability of this method to solve first order IVPs. This article will describe in subsequent sections, the derivation of the new sixth order inverse polynomial method, its properties in terms of convergence and stability will be identified, and some numerical examples will be considered. In the section detailing the numerical results, comparison will be made with the Runge-Kutta method of order four (RK4) in the context of the exact value (EV). This gives a fair basis of comparison to show the suitability, accuracy and effectiveness of the new method in the context of the exact value. Thereafter, the sections following will discuss the results obtained and then conclude the article.

2. Derivation of a new proposed sixth order inverse polynomial method

To derive the SOIPM for the solution of initial value problem of the form (1.1), consider the inverse interpolating polynomial

$$y_{n+1} = y_n \left[\sum_{j=0}^{k} a_j x_n^j \right]^{-1},$$

where the coefficients a_j's are to be determined from nonlinear equations that will be generated by the sixth order inverse polynomial, $k = 6$,

$$y_{n+1} = y_n \left[\sum_{j=0}^{6} a_j x_n^j \right]^{-1} = y_n \left[a_0 x_n^0 + a_1 x_n + a_2 x_n^2 + a_3 x_n^3 + a_4 x_n^4 + a_5 x_n^5 + a_6 x_n^6 \right]^{-1}. \tag{2.1}$$

Setting $a_0 = 1$, (2.1) becomes

$$y_{n+1} = y_n [1 + a_1 x_n + a_2 x_n^2 + a_3 x_n^3 + a_4 x_n^4 + a_5 x_n^5 + a_6 x_n^6]^{-1}. \tag{2.2}$$

Taking the binomial expansion of (2.2), we get

$$\begin{aligned} y_{n+1} \approx y_n[1 &- (a_1x_n + a_2x_n^2 + a_3x_n^3 + a_4x_n^4 + a_5x_n^5 + a_6x_n^6) \\ &+ (a_1x_n + a_2x_n^2 + a_3x_n^3 + a_4x_n^4 + a_5x_n^5 + a_6x_n^6)^2 \\ &- (a_1x_n + a_2x_n^2 + a_3x_n^3 + a_4x_n^4 + a_5x_n^5 + a_6x_n^6)^3 \\ &+ (a_1x_n + a_2x_n^2 + a_3x_n^3 + a_4x_n^4 + a_5x_n^5 + a_6x_n^6)^4 \\ &- (a_1x_n + a_2x_n^2 + a_3x_n^3 + a_4x_n^4 + a_5x_n^5 + a_6x_n^6)^5 \\ &+ (a_1x_n + a_2x_n^2 + a_3x_n^3 + a_4x_n^4 + a_5x_n^5 + a_6x_n^6)^6]. \end{aligned} \tag{2.3}$$

Equating the LHS of (2.3) in terms of Taylor's series expansion of the form

$$y(x_n + h) \approx y_n + hy'_n + \frac{h^2}{2!}y''_n + \frac{h^3}{3!}y'''_n + \frac{h^4}{4!}y_n^{(iv)} + \frac{h^5}{5!}y_n^{(v)} + \frac{h^6}{6!}y_n^{(vi)}$$

yields

$$\begin{aligned} &y_n + hy'_n + \frac{h^2}{2!}y''_n + \frac{h^3}{3!}y'''_n + \frac{h^4}{4!}y_n^{(iv)} + \frac{h^5}{5!}y_n^{(v)} + \frac{h^6}{6!}y_n^{(vi)} \\ &= y_n - a_1x_ny_n + (a_1^2 - a_2)x_n^2y_n + (-a_1^3 + 2a_1a_2 - a_3)x_n^3y_n + (a_1^4 - 3a_1^2a_2 + 2a_1a_3 + a_2^2 - a_4)x_n^4y_n \\ &\quad + (-a_1^5 + 4a_1^3a_2 - 3a_1a_2^2 - 3a_1a_2^2 + 2a_1a_4 + 2a_2a_3 - a_5)x_n^5y_n \\ &\quad + (a_1^6 - 5a_1^4a_2 + 4a_1^3a_3 + a_3^2 + 6a_1^2a_2^2 - a_2^3 - 3a_1^2a_4 + 2a_2a_4 + 2a_1a_5 - 6a_1a_2a_3 - a_6)x_n^6y_n. \end{aligned} \tag{2.4}$$

Equating (2.4) term by term, we obtain the values of $a_1, a_2, a_3, a_4, a_5, a_6$ as follows

$$\begin{aligned} a_1 &= \frac{-hy'_n}{x_ny_n}, \\ a_2 &= \frac{2h^2(y'_n)^2 - h^2y_ny''_n}{2x_n^2y_n^2}, \\ a_3 &= \left[\frac{-h^3y_n^2y'''_n - 6h^3(y'_n)^3 + 6h^3y_ny'_ny''_n}{6x_n^3y_n^3}\right], \\ a_4 &= \frac{1}{24x_n^4y_n^4}[24h^4(y'_n)^4 - 36h^4y_n(y'_n)^2y''_n + 8h^4y_n^2y'_ny'''_n - h^4y_n^3y_n^{(iv)} + 6h^4(y_n)^2(y'')_n)^2], \\ a_5 &= \frac{1}{120x_n^5y_n^5}[-120h^5(y'_n)^5 + 240h^5y_n(y'_n)^3y''_n - 180h^5y_n^2(y'_n)^2y''_n \\ &\quad - 90h^5y_n^2(y'_n)(y''_n)^2 + 10h^5y_n^3(y'_n)y_n^{(iv)} + 20h^5y_n^3y''_ny'''_n - h^5y_n^{(v)}], \\ a_6 &= \frac{1}{720x_n^6y_n^6}[-720h^6(y'_n)^6 - 1800h^6y'_ny_ny''_n + 480h^6(y'_n)^3(y_n)^5 + 1080h^6(y'_n)^2(y_n)^2(y''_n)^2 \\ &\quad + 90h^6(y_n)^3(y''_n)^3 + 270h^6(y'_n)^2(y_n)^5y_n^{(iv)}(y''_n)^2 - 120h^6(y'_n)^2(y_n)^3y_n^{(iv)}(y''_n)^2], \end{aligned} \tag{2.5}$$

Substituting (2.5) into (2.2), one obtains

$$\begin{aligned} y_{n+1} = y_n\Bigg[1 &- \left[\frac{hy'_n}{x_ny_n}\right]x_n + \left[\frac{2h^2(y'_n)^2 - h^2y_ny''_n}{2x_n^2y_n^2}\right]x_n^2 + \left[\frac{-h^3y_n^2y'''_n - 6h^3(y'_n)^3 + 6h^3y_ny'_ny''_n}{6x_n^3y_n^3}\right]x_n^3 \\ &+ \left[\frac{24h^4(y'_n)^4 - 36h^4y_n(y'_n)^2y''_n + 8h^4y_n^2y'_ny'''_n - h^4y_n^3y_n^{(iv)} + 6h^4(y_n)^2(y''_n)^2}{24x_n^4y_n^4}\right]x_n^4 \\ &+ \left[\frac{-120h^5(y'_n)^5 + 240h^5y_n(y'_n)^3y''_n - 180h^5y_n^2(y'_n)^2y'''_n - 90h^5y_n^2(y'_n)(y''_n)^2}{120x_n^5y_n^5}\right]x_n^5 \end{aligned}$$

$$
\begin{aligned}
&+\left[\frac{10h^5y_n^3(y_n')y_n^{(iv)}+20h^5y_n^3y_n''y_n'''-h^5y_n^{(v)}}{120x_n^5y_n^5}\right]x_n^5\\
&+\frac{1}{720x_n^6y_n^6}[-720h^6(y_n')^6-1800h^6y_n'y_ny_n''+480h^6(y_n')^3(y_n)^5+1080h^6(y_n')^2(y_n)^2(y_n'')^2\\
&+90h^6(y_n)^3(y_n'')^3+270h^6(y_n')^2(y_n)^5y_n^{(iv)}(y_n'')^2-120h^6(y_n')^2(y_n)^3y_n^{(iv)}(y_n'')^2]x_n^6\Big]^{-1},
\end{aligned}
$$

$$
\begin{aligned}
y_{n+1}=y_n\Big[1&-\left[\frac{hy_n'}{y_n}\right]+\left[\frac{2h^2(y_n')^2-h^2y_ny_n''}{2y_n^2}\right]+\left[\frac{-h^3y_n^2y_n'''-6h^3(y_n')^3+6h^3y_ny_n'y_n''}{6y_n^3}\right]\\
&+\left[\frac{24h^4(y_n')^4-36h^4y_n(y_n')^2y_n''+8h^4y_n^2y_n'y_n'''-h^4y_n^3y_n^{(iv)}+6h^4(y_n)^2(y_n'')^2}{24y_n^4}\right]\\
&+\left[\frac{-120h^5(y_n')^5+240h^5y_n(y_n')^3y_n''-180h^5y_n^2(y_n')^2y_n'''-90h^5y_n^2(y_n')(y_n'')^2}{120y_n^5}\right]\\
&+\left[\frac{10h^5y_n^3(y_n')y_n^{(iv)}+20h^5y_n^3y_n''y_n'''-h^5y_n^{(v)}}{120y_n^5}\right]\\
&+\frac{1}{720y_n^6}[-720h^6(y_n')^6-1800h^6y_n'y_ny_n''+480h^6(y_n')^3(y_n)^5+1080h^6(y_n')^2(y_n)^2(y_n'')^2\\
&+90h^6(y_n)^3(y_n'')^3+270h^6(y_n')^2(y_n)^3y_n^{(iv)}(y_n'')^2-120h^6(y_n')^2(y_n)^3y_n^{(iv)}(y_n'')^2]\Big]^{-1}.
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
y_{n+1}=720y_n^7\Big[&720y_n^6-720hy_n^5y_n'+720h^2y_n^4(y_n')^2-360y_n^5h^2y_n''-120h^3y_n^5y_n'''-720h^3y_n^3(y_n')^3\\
&+720h^3y_n^4y_n'y_n''+720y_n^2h^4(y_n')^4-1080h^4y_n^3(y_n')^2y_n''+240h^4y_n^4y_ny_n'''\\
&-30h^4y_n^5y_n^{(iv)}+180h^4(y_n)^4(y_n'')^2-760y_nh^5(y_n')^5\\
&+1440h^5y_n^2(y_n')^3y_n''-1080h^5y_n^3(y_n')^2y_n'''-540h^5y_n^3(y_n')(y_n'')^2\\
&+60h^5y_n^4(y_n')y_n^{(iv)}+120h^5y_n^4y_n''y_n'''-6h^5y_ny_n^{(v)}\\
&-720h^6(y_n')^6-1800h^6y_n'y_ny_n''+480h^6(y_n')^3(y_n)^5+1080h^6(y_n')^2(y_n)^2(y_n'')^2\\
&+90h^6(y_n)^3(y_n'')^3+270h^6(y_n')^2(y_n)^5(y_n)^{(iv)}(y_n'')^2-120h^6(y_n')^2(y_n)^3(y_n^{(iv)})(y_n'')^2\Big]^{-1}.
\end{aligned}
\tag{2.6}
$$

Equation (2.6) is the new sixth order inverse polynomial method.

3. The properties of sixth order inverse polynomial method

In this section, the consistency, stability and convergence properties of the newly constructed sixth order inverse polynomial method were analyzed and investigated in the following results.

3.1. Consistency property of SOIPM

Theorem 3.1. *Consider the SOIPM given by (2.6) for solving (1.1), then the increment function is*

$$\phi(x_n, y_n; h) = f(x_n, y_n) \quad as \quad h \to 0.$$

Proof. From the definition of general one step method we have that

$$\frac{y_{n+1}-y_n}{h} = \phi(x_n, y_n; h). \tag{3.1}$$

Using (2.6), yields

$$y_{n+1}-y_n=\frac{720y_n^7}{\begin{bmatrix}720y_n^6-720hy_n^5y_n'+720h^2y_n^4(y_n')^2-360y_n^5h^2y_n''-120h^3y_n^5y_n'''-720h^3y_n^3(y_n')^3\\ +720h^3y_n^4y_n'y_n''+720y_n^2h^4(y_n')^4-1080h^4y_n^3(y_n')^2y_n''+240h^4y_n^4y_n'y_n'''\\ -30h^4y_n^5y_n^{(iv)}+180h^4(y_n)^4(y_n'')^2-760y_nh^5(y_n')^5+1440h^5y_n^2(y_n')^3y_n''\\ -1080h^5y_n^3(y_n')^2y_n'''-540h^5y_n^3(y_n')(y_n'')^2+60h^5y_n^4(y_n')y_n^{(iv)}+120h^5y_n^4y_n''y_n'''\\ -6h^5y_ny_n^{(v)}-720h^6(y_n')^6-1800h^6y_n'y_ny_n''+480h^6(y_n')^3(y_n)^5\\ +1080h^6(y_n')^2(y_n)^2(y_n'')^2+90h^6(y_n)^3(y_n'')^3\\ +270h^6(y_n')^2(y_n)^5(y_n)^{(iv)}(y_n'')^2-120h^6(y_n')^2(y_n)^3(y_n^{(iv)})(y_n'')^2\end{bmatrix}}-y_n. \tag{3.2}$$

Simplifying further, (3.2) becomes

$$y_{n+1}-y_n=\frac{\begin{bmatrix}720hy_n^6y_n'-720h^2y_n^5(y_n')^2+360y_n^6h^2y_n''+120h^3y_n^6y_n'''+720h^3y_n^4(y_n')^3\\ -720h^3y_n^5y_n'y_n''-720y_n^3h^4(y_n')^4+1080h^4y_n^4(y_n')^2y_n''-240h^4y_n^5y_n'y_n'''\\ +30h^4y_n^6y_n^{(iv)}-180h^4(y_n)^5(y_n'')^2+760y_n^2h^5(y_n')^5-1440h^5y_n^3(y_n')^3y_n''\\ +1080h^5y_n^4(y_n')^2y_n'''+540h^5y_n^4(y_n')(y_n'')^2-60h^5y_n^5(y_n')y_n^{(iv)}\\ -120h^5y_n^5y_n''y_n'''+6h^5y_n^2y_n^{(v)}+720h^6y_n(y_n')^6+1800h^6y_n'y_n^2y_n''\\ -480h^6(y_n')^3(y_n)^6-1080h^6(y_n')^2(y_n)^3(y_n'')^2-90h^6(y_n)^4(y_n'')^3\\ -270h^6(y_n')^2(y_n)^6(y_n)^{(iv)}(y_n'')^2+120h^6(y_n')^2(y_n)^4(y_n^{(iv)})(y_n'')^2\end{bmatrix}}{\begin{bmatrix}720y_n^6-720hy_n^5y_n'+720h^2y_n^4(y_n')^2-360y_n^5h^2y_n''-120h^3y_n^5y_n'''-720h^3y_n^3(y_n')^3\\ +720h^3y_n^4y_n'y_n''+720y_n^2h^4(y_n')^4-1080h^4y_n^3(y_n')^2y_n''\\ +240h^4y_n^4y_n'y_n'''-30h^4y_n^5y_n^{(iv)}+180h^4(y_n)^4(y_n'')^2-760y_nh^5(y_n')^5\\ +1440h^5y_n^2(y_n')^3y_n''-1080h^5y_n^3(y_n')^2y_n'''-540h^5y_n^3(y_n')(y_n'')^2\\ +60h^5y_n^4(y_n')y_n^{(iv)}+120h^5y_n^4y_n''y_n'''-6h^5y_ny_n^{(v)}-720h^6(y_n')^6\\ -1800h^6y_n'y_ny_n''+480h^6(y_n')^3(y_n)^5+1080h^6(y_n')^2(y_n)^2(y_n'')^2+90h^6(y_n)^3(y_n'')^3\\ +270h^6(y_n')^2(y_n)^5(y_n)^{(iv)}(y_n'')^2-120h^6(y_n')^2(y_n)^3(y_n^{(iv)})(y_n'')^2\end{bmatrix}}. \tag{3.3}$$

This implies that

$$\frac{y_{n+1}-y_n}{h}=\frac{\begin{bmatrix}720y_n^6y_n'-720hy_n^5(y_n')^2+360y_n^6hy_n''+120h^2y_n^6y_n'''+720h^2y_n^4(y_n')^3-720h^2y_n^5y_n'y_n''\\ -720y_n^3h^2(y_n')^4+1080h^3y_n^4(y_n')^2y_n''-240h^3y_n^5y_n'y_n'''+30h^3y_n^6y_n^{(iv)}\\ -180h^3(y_n)^5(y_n'')^2+760y_n^2h^4(y_n')^5-1440h^4y_n^3(y_n')^3y_n''+1080h^4y_n^4(y_n')^2y_n'''\\ +540h^4y_n^4(y_n')(y_n'')^2-60h^4y_n^5(y_n')y_n^{(iv)}-120h^4y_n^5y_n''y_n'''+6h^4y_n^2y_n^{(v)}+720h^5y_n(y_n')^6\\ +1800h^5y_n'y_n^2y_n''-480h^5(y_n')^3(y_n)^6-1080h^5(y_n')^2(y_n)^3(y_n'')^2-90h^5(y_n)^4(y_n'')^3\\ -270h^5(y_n')^2(y_n)^6(y_n)^{(iv)}(y_n'')^2+120h^5(y_n')^2(y_n)^4(y_n^{(iv)})(y_n'')^2\end{bmatrix}}{\begin{bmatrix}720y_n^6-720hy_n^5y_n'+720h^2y_n^4(y_n')^2-360y_n^5h^2y_n''-120h^3y_n^5y_n'''-720h^3y_n^3(y_n')^3\\ +720h^3y_n^4y_n'y_n''+720y_n^2h^4(y_n')^4-1080h^4y_n^3(y_n')^2y_n''+240h^4y_n^4y_n'y_n'''\\ -30h^4y_n^5y_n^{(iv)}+180h^4(y_n)^4(y_n'')^2-760y_nh^5(y_n')^5+1440h^5y_n^2(y_n')^3y_n''\\ -1080h^5y_n^3(y_n')^2y_n'''-540h^5y_n^3(y_n')(y_n'')^2+60h^5y_n^4(y_n')y_n^{(iv)}+120h^5y_n^4y_n''y_n'''\\ -6h^5y_ny_n^{(v)}-720h^6(y_n')^6-1800h^6y_n'y_ny_n''+480h^6(y_n')^3(y_n)^5\\ +1080h^6(y_n')^2(y_n)^2(y_n'')^2+90h^6(y_n)^3(y_n'')^3\\ +270h^6(y_n')^2(y_n)^5(y_n)^{(iv)}(y_n'')^2-120h^6(y_n')^2(y_n)^3(y_n^{(iv)})(y_n'')^2\end{bmatrix}}. \tag{3.4}$$

Comparing (3.1) and (3.4), the increment function $\phi(x_n, y_n; h)$ for SOIPM is given by

$$\phi(x_n, y_n; h) = \frac{\left[\begin{array}{l} 720y_n^6 y_n' - 720hy_n^5(y_n')^2 + 360y_n^6 hy_n'' + 120h^2 y_n^6 y_n''' + 720h^2 y_n^4 (y_n')^3 \\ -720h^2 y_n^5 y_n' y_n'' - 720y_n^3 h^2 (y_n')^4 + 1080h^3 y_n^4 (y_n')^2 y_n'' - 240h^3 y_n^5 y_n' y_n''' \\ +30h^3 y_n^6 y_n^{(iv)} - 180h^3 (y_n)^5 (y_n'')^2 + 760y_n^2 h^4 (y_n')^5 - 1440h^4 y_n^3 (y_n')^3 y_n'' \\ +1080h^4 y_n^4 (y_n')^2 y_n''' + 540h^4 y_n^4 (y_n')(y_n'')^2 - 60h^4 y_n^5 (y_n') y_n^{(iv)} \\ -120h^4 y_n^5 y_n'' y_n''' + 6h^4 y_n^2 y_n^{(v)} + 720h^5 y_n (y_n')^6 + 1800h^5 y_n' y_n^2 y_n'' \\ -480h^5 (y_n')^3 (y_n)^6 - 1080h^5 (y_n')^2 (y_n)^3 (y_n'')^2 - 90h^5 (y_n)^4 (y_n'')^3 \\ -270h^5 (y_n')^2 (y_n)^6 (y_n)^{(iv)} (y_n'')^2 + 120h^5 (y_n')^2 (y_n)^4 (y_n^{(iv)})(y_n'')^2 \end{array}\right]}{\left[\begin{array}{l} 720y_n^6 - 720hy_n^5 y_n' + 720h^2 y_n^4 (y_n')^2 - 360y_n^5 h^2 y_n'' - 120h^3 y_n^5 y_n''' - 720h^3 y_n^3 (y_n')^3 \\ +720h^3 y_n^4 y_n' y_n'' + 720y_n^2 h^4 (y_n')^4 - 1080h^4 y_n^3 (y_n')^2 y_n'' + 240h^4 y_n^4 y_n' y_n''' \\ -30h^4 y_n^5 y_n^{(iv)} + 180h^4 (y_n)^4 (y_n'')^2 - 760y_n h^5 (y_n')^5 + 1440h^5 y_n^2 (y_n')^3 y_n'' \\ -1080h^5 y_n^3 (y_n')^2 y_n''' - 540h^5 y_n^3 (y_n')(y_n'')^2 + 60h^5 y_n^4 (y_n') y_n^{(iv)} + 120h^5 y_n^4 y_n'' y_n''' \\ -6h^5 y_n y_n^{(v)} - 720h^6 (y_n')^6 - 1800h^6 y_n' y_n y_n'' + 480h^6 (y_n')^3 (y_n)^5 \\ +1080h^6 (y_n')^2 (y_n)^2 (y_n'')^2 + 90h^6 (y_n)^3 (y_n'')^3 \\ +270h^6 (y_n')^2 (y_n)^5 (y_n)^{(iv)} (y_n'')^2 - 120h^6 (y_n')^2 (y_n)^3 (y_n^{(iv)})(y_n'')^2 \end{array}\right]}. \tag{3.5}$$

Taking the limit of (3.5) as $h \to 0$, this yields

$$\lim_{h\to 0} \phi(x_n, y_n; h) = \frac{720y_n^6 y_n'}{720y_n^6} = y_n' = f(x_n, y_n). \tag{3.6}$$

Equation (3.6) confirms the consistency property of (2.6). □

3.2. *Zero stability property of SOIPM*

Theorem 3.2. *Consider the linear multistep method with $k = 1$ of the form*

$$\sum_{j=0}^{1} \lambda_j y_{n+j} = h \sum_{j=0}^{1} \beta_j f_{n+j}$$

and the SOIPM given by (2.6) for solving (1.1). Let the first characteristic polynomial is defined as

$$\eta(d) = \sum_{j=0}^{1} \lambda_j d^j,$$

then SOIPM satisfies the Dahlquist root conditions:

a) *all roots of the first characteristic polynomial satisfy $|d| \leqslant 1$;*
b) *multiple roots of the first characteristic polynomial satisfy $|d| < 1$.*

Proof. From (2.6), we deduce that $\lambda_1 = 1$ and $\lambda_0 = -1$, so the first characteristic polynomial is obtained as

$$\eta\,(d) = \lambda_1 d + \lambda_0 = \eta\,(d) = d - 1 \quad \Longrightarrow \eta\,(d) = 0 \implies d - 1 = 0 \implies d = 1.$$

Since SOIPM satisfies the Dahlquist root conditions, hence it is zero stable. □

3.3. Stability property of SOIPM

3.3.1. Case 1

Theorem 3.3. *Consider the SOIPM (2.6) for solving (1.1) with $h \in (0, h_0)$. Let the initial error be given by e_0, there exist a constant M such that $h > 0$, then the ultimate error e_n satisfies the inequality*

$$e_n \leqslant Me_0, \; M \in (0, 1).$$

Proof. Using (2.1), one can write

$$y_{n+h} = y_{n+h-1}\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1}. \tag{3.7}$$

The theoretical solution $y(x)$ is given as

$$y(x_{n+h}) = y(x_{n+h-1})\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} + T_{n+h}. \tag{3.8}$$

Subtracting (3.7) from (3.8), yields

$$\begin{aligned} y(x_{n+h}) - y_{n+h} &= y(x_{n+h-1})\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} - y_{n+h-1}\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} + T_{n+h} \\ &= [y(x_{n+h-1}) - y_{n+h-1}]\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} + T_{n+h}. \end{aligned}$$

Therefore,

$$e_{n+h} = e_{n+h-1}\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} + T_{n+h}, \tag{3.9}$$

where

$$e_{n+h} = y(x_{n+h}) - y_{n+h}, e_{n+h-1} = [y(x_{n+h-1}) - y_{n+h-1}].$$

Taking the norm of both sides of (3.9), one obtains

$$\|e_{n+h}\| = \left\| e_{n+h-1}\left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} \right\| + \|T_{n+h}\|.$$

Let

$$Q = \sum_{j=0}^{6} a_j x_{n+h}^j.$$

Then

$$\left\| \left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} \right\| = \left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1}, \; \left[\sum_{j=0}^{6} a_j x_{n+h}^j\right]^{-1} = Q^{-1} = M, \;\; \|e_{n+h}\| \leqslant M\|e_{n+h-1}\| + \|T_{n+h}\|.$$

Let

$$E_{n+h} = \sup(e_{n+h}), T = \sup(T_{n+h}), E_{n+h-1} = \sup(e_{n+h-1}),\ 0 < n < \infty.$$

We then have

$$\begin{aligned}&E_{n+h} \leqslant ME_{n+h-1} + T,\\ &\text{For } h = 1, \implies E_{n+1} \leqslant ME_n + T,\\ &\text{For } h = 2, \implies E_{n+2} \leqslant ME_{n+1} + T \leqslant M(ME_n + T) + TE_{n+2} \leqslant M^2E_n + MT + T.\end{aligned}$$

Continuing this way, we have

$$E_{n+6} \leqslant M^6E_n + \sum_{\alpha=0}^{5} M^{\alpha}T, \quad |e_{n+6}| < E_{n+h} < M^6E_n + \sum_{\alpha=0}^{5} M^{\alpha}T.$$

Since $M < 1$ and as $n \to \infty$, $E_{n+h} \to 0$ and this shows that SOIPM is stable. □

Remark 3.4. For the general case, we have that

$$E_{n+h} \leqslant M^kE_n + \sum_{\alpha=0}^{h-1} M^{\alpha}T$$

and

$$|e_{n+h}| < E_{n+h} < M^kE_n + \sum_{\alpha=0}^{h-1} M^{\alpha}T.$$

3.3.2. Case 2: linear stability analysis of the method

Theorem 3.5. *Consider the linear test equation of the form*

$$y' = \tau y, y(x_0) = y_0, \tag{3.10}$$

where τ is a constant, then stability function of SOIPM (2.6) yields

$$p(z) = \sum_{q=0}^{6} \frac{z^q}{\Gamma(q+1)}.$$

Proof. From (3.10), we have that

$$\begin{aligned}y'_n = f_n = \tau y_n, y''_n = f_n^{(1)} = \tau^2 y_n, y'''_n = f_n^{(2)} = \tau^3 y_n,\\ y_n^{(iv)} = f_n^{(3)} = \tau^4 y_n, y_n^{(v)} = f_n^{(4)} = \tau^5 y_n, y_n^{(vi)} = f_n^{(5)} = \tau^6 y_n.\end{aligned}$$

Thus, (2.6) becomes

$$\frac{y_{n+1}}{y_n} = \left(1 + \tau h + \frac{(\tau h)^2}{2!} + \frac{(\tau h)^3}{3!} + \frac{(\tau h)^4}{4!} + \frac{(\tau h)^5}{5!} + \frac{(\tau h)^6}{6!}\right). \tag{3.11}$$

Setting $z = \tau h$, then (3.11) becomes

$$\frac{y_{n+1}}{y_n} = \left(1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \frac{z^4}{4!} + \frac{z^5}{5!} + \frac{z^6}{6!}\right) = p(z).$$

This completes the proof. □

The linear stability region of SOIPM is obtained and given in Figure 1.

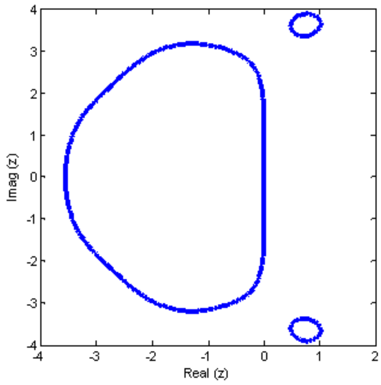

Figure 1: Linear stability region of SOIPM.

3.4. Convergence property of SOIPM

Theorem 3.6. *Let $\overline{y}_n$ be defined as a point in the interior of the interval whose end points are y_n and y_n^*. Applying the mean value theorem, then SOIPM (2.6) is convergent and the increment function (3.5) is Lipschitzian.*

Proof. Simplifying (2.6) further as

$$y_{n+1} - y_n = h\left(f_n + Af_n^{(1)} + Bf_n^{(2)} + Cf_n^{(3)} + Df_n^{(4)} + Ef_n^{(5)}\right),$$

where

$$A = \frac{h}{2}, B = \frac{h^2}{6}, C = \frac{h^3}{24}, D = \frac{h^4}{120}, E = \frac{h^5}{720},$$

thus, the increment function becomes

$$\begin{aligned}\Phi(x_n, y_n; h) &= f_n + Af_n^{(1)} + Bf_n^{(2)} + Cf_n^{(3)} + Df_n^{(4)} + Ef_n^{(5)} \\ &= f(x_n, y_n) + Af^{(1)}(x_n, y_n) + Bf^{(2)}(x_n, y_n) \\ &\quad + Cf^{(3)}(x_n, y_n) + Df^{(4)}(x_n, y_n) + Ef^{(5)}(x_n, y_n).\end{aligned} \tag{3.12}$$

Suppose

$$\begin{aligned}\Phi(x_n, y_n^*; h) = f(x_n, y_n^*) &+ Af^{(1)}(x_n, y_n^*) + Bf^{(2)}(x_n, y_n^*) \\ &+ Cf^{(3)}(x_n, y_n^*) + Df^{(4)}(x_n, y_n^*) + Ef^{(5)}(x_n, y_n^*).\end{aligned} \tag{3.13}$$

Subtracting (3.13) from (3.12), one gets

$$\begin{aligned}\Phi(x_n, y_n; h) - \Phi(x_n, y_n^*; h) = f(x_n, y_n) - f(x_n, y_n^*) &+ A[f^{(1)}(x_n, y_n) - f^{(1)}(x_n, y_n^*)] \\ &+ B[f^{(2)}(x_n, y_n) - f^{(2)}(x_n, y_n^*)] + C[f^{(3)}(x_n, y_n) - f^{(3)}(x_n, y_n^*)] \\ &+ D[f^{(4)}(x_n, y_n) - f^{(4)}(x_n, y_n^*)] + E[f^{(5)}(x_n, y_n) - f^{(5)}(x_n, y_n^*)]\end{aligned} \tag{3.14}$$

Let $\overline{y}_n$ be defined as a point in the interior of the interval whose end points are y_n and y_n^*. Applying the mean value theorem, we have

$$f(x_n, y_n) - f(x_n, y_n^*) = \sup_{(x_n, y_n)\in D} \frac{\partial f(x_n, \overline{y}_n)}{\partial y_n}(y_n - y_n^*), \tag{3.15}$$

$$f^{(1)}(x_n, y_n) - f^{(1)}(x_n, y_n^*) = \sup_{(x_n, y_n)\in D} \frac{\partial f^{(1)}(x_n, \overline{y}_n)}{\partial y_n}(y_n - y_n^*), \tag{3.16}$$

$$f^{(2)}(x_n,y_n) - f^{(2)}(x_n,y_n^*) = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(2)}(x_n,\overline{y}_n)}{\partial y_n}(y_n - y_n^*), \tag{3.17}$$

$$f^{(3)}(x_n,y_n) - f^{(3)}(x_n,y_n^*) = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(3)}(x_n,\overline{y}_n)}{\partial y_n}(y_n - y_n^*), \tag{3.18}$$

$$f^{(4)}(x_n,y_n) - f^{(4)}(x_n,y_n^*) = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(4)}(x_n,\overline{y}_n)}{\partial y_n}(y_n - y_n^*), \tag{3.19}$$

$$f^{(5)}(x_n,y_n) - f^{(5)}(x_n,y_n^*) = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(5)}(x_n,\overline{y}_n)}{\partial y_n}(y_n - y_n^*). \tag{3.20}$$

We define,

$$F = \sup_{(x_n,y_n)\in D} \frac{\partial f(x_n,\overline{y}_n)}{\partial y_n}, \qquad G = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(1)}(x_n,\overline{y}_n)}{\partial y_n}, \qquad H = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(2)}(x_n,\overline{y}_n)}{\partial y_n},$$
$$I = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(3)}(x_n,\overline{y}_n)}{\partial y_n}, \qquad J = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(4)}(x_n,\overline{y}_n)}{\partial y_n}, \qquad K = \sup_{(x_n,y_n)\in D} \frac{\partial f^{(5)}(x_n,\overline{y}_n)}{\partial y_n}.$$

Equations (3.15)-(3.20) become

$$\begin{aligned} f(x_n,y_n) - f(x_n,y_n^*) &= F(y_n - y_n^*), & f^{(1)}(x_n,y_n) - f^{(1)}(x_n,y_n^*) &= G(y_n - y_n^*),\\ f^{(2)}(x_n,y_n) - f^{(2)}(x_n,y_n^*) &= H(y_n - y_n^*), & f^{(3)}(x_n,y_n) - f^{(3)}(x_n,y_n^*) &= I(y_n - y_n^*),\\ f^{(4)}(x_n,y_n) - f^{(4)}(x_n,y_n^*) &= J(y_n - y_n^*), & f^{(5)}(x_n,y_n) - f^{(5)}(x_n,y_n^*) &= K(y_n - y_n^*). \end{aligned} \tag{3.21}$$

Substituting (3.21) into (3.14) and simplifying further yields

$$\Phi(x_n,y_n;h) - \Phi(x_n,y_n^*;h) = [F + AG + BH + CI + DJ + EK](y_n - y_n^*). \tag{3.22}$$

Taking the norm of both sides of (3.22), one gets

$$\|\Phi(x_n,y_n;h) - \Phi(x_n,y_n^*;h)\| \le \|F + AG + BH + CI + DJ + EK\| \, \|y_n - y_n^*\|, \tag{3.23}$$

where the Lipschitz constant is given by

$$L = \|F + AG + BH + CI + DJ + EK\|.$$

Therefore, (3.23) becomes

$$\|\Phi(x_n,y_n;h) - \Phi(x_n,y_n^*;h)\| \le L\,\|y_n - y_n^*\|.$$

Hence, SOIPM (2.6) is convergent and the increment function $\Phi(x_n,y_n;h)$ (3.12) is Lipschitzian. □

Remark 3.7. Alternatively, for a numerical method to be convergent, the necessary and sufficient conditions are consistency and stability. Since these conditions have been achieved previously, we can conclude that SOIPM is convergent.

Remark 3.8. The convergence of the proposed scheme is given by the following result.

Theorem 3.9. *Given any well-posed initial value problem, then the SOIPM* (2.6) *is convergent, since it satisfies the following conditions:*

a) *consistency;*
b) *stability.*

Remark 3.10. By means of the local truncation error (LTE) for (2.6) obtained as

$$LTE_{SOIPM} = \frac{h^7}{5040} f^{(6)}(x_n,y_n) + \cdots$$

it is clearly seen that the derived method (2.6) has sixth order accuracy.

4. Numerical examples

The suitability, accuracy and performance of the derived method are tested on the following non-linear physical models with different flavours. The comparative results analyses of SOIPM (2.6), RK4 given by

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4), \tag{4.1}$$

with

$$k_1 = f(x_n, y_n), k_2 = f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1\right), k_3 = f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2\right), k_4 = f(x_n + h, y_n + k_3),$$

and the exact value shall be presented. The absolute error at x_n and final absolute error on $[a, b]$ defined, respectively by

$$\text{ABE} = |y(x_n) - y_N| \quad \text{and} \quad \text{FABE} = \max_{a \leqslant n \leqslant b} |y(x_{n+1}) - y_{n+1}|$$

for SOIPM (2.6) and RK4 (4.1) shall be computed.

Example 4.1. Consider a non-linear physical model of the form

$$\frac{dy}{dt} = a_1 y - a_2 y^2, \quad y(0) = 571, \quad a_1 = \frac{7}{25}, \quad a_2 = \frac{7}{2000000},$$

with the exact value given by

$$y(t) = \frac{571 a_1 e^{a_1 t}}{a_1 + 571 a_2 (e^{a_1 t} - 1)}, \tag{4.2}$$

where a_1 is the virus transmission mechanism rate and a_2 is the quarantine rate. The comparative results analyzes of SOIPM (2.6), RK4 (4.1) and EV (4.2) and ABEs computed are displayed in Tables 1 and 2.

Example 4.2. Consider a non-linear physical model of the form

$$\frac{dy}{dx} = \frac{ry}{5}\left(\frac{80 - y}{80}\right), \quad y(0) = 4, \quad r = 25$$

with the exact value given by

$$y(x) = \frac{80e^{5x}}{e^{5x} + 19}. \tag{4.3}$$

The comparative results analyzes of SOIPM (2.6), RK4 (4.1) and EV (4.3) and ABEs computed are displayed in Table 3. The FABEs generated via SOIPM (2.6) and RK4 (4.1) are displayed in Table 4.

Example 4.3. Consider a non-linear physical model of the form

$$\frac{dy}{dx} = y - y^2, \quad y(0) = \frac{2}{5}$$

with the exact value given by

$$y(x) = \frac{\frac{2}{5}e^x}{1 + \frac{2}{5}(e^x - 1)}. \tag{4.4}$$

The comparative results analyzes of SOIPM (2.6), RK4 (4.1) and EV (4.4) and ABEs computed are displayed in Table 5. The FABEs generated via SOIPM (2.6) and RK4 (4.1) are displayed in Table 6.

Table 1: The comparative results analyses of SOIPM (2.6), RK4 (4.1), and EV (4.2) for Example 4.1 with $h = 0.1$ and different values of time t (days).

t(days)	SOIPM	RK4	EV	ABE_{SOIPM}	ABE_{RK4}
0	571.000000000000	571.000000000000	571.000000000000	0.000000000000	0.000000000000
5	2266.103420036719	2266.103405749799	2266.103420036581	0.000000000139	0.000014286782
10	8457.563886923883	8457.563798505764	8457.563886922031	0.000000001852	0.000088416267
15	25923.854096541858	25923.853848460403	25923.854096542524	0.000000000666	0.000248082120
20	52826.493984558489	52826.493719219063	52826.493984553577	0.000000004911	0.000265334515
25	70994.527830096296	70994.527692920543	70994.527830096500	0.000000000204	0.000137175957
30	77573.486837016710	77573.486784128749	77573.486837016419	0.000000000291	0.000052887670
35	79387.635604725554	79387.635586962890	79387.635604725525	0.000000000029	0.000017762635
40	79848.116894757739	79848.116889236670	79848.116894757666	0.000000000073	0.000005520997
45	79962.492437872614	79962.492436233428	79962.492437872585	0.000000000029	0.000001639157
50	79990.747480797407	79990.747480325139	79990.747480797450	0.000000000044	0.000000472312
55	79997.718158025498	79997.718157892130	79997.718158025382	0.000000000116	0.000000133252
60	79999.437292604693	79999.437292567658	79999.437292604678	0.000000000015	0.000000037020

Table 2: The comparative results analyses of SOIPM (2.6), RK4 (4.1), and EV (4.2) for Example 4.1 with $h = 0.1$.

t_n (days)	SOIPM	RK4	EV	ABE_{SOIPM}	ABE_{RK4}
0.0	571.000000000000	571.000000000000	571.000000000000	0.000000000000	0.000000000000
0.1	587.094946807399	587.094946728635	587.094946807399	0.000000000000	0.000000078764
0.2	603.640118867513	603.640118705663	603.640118867514	0.000000000000	0.000000161851
0.3	620.647913364301	620.647913114868	620.647913364302	0.000000000001	0.000000249433
0.4	638.131057592256	638.131057250567	638.131057592257	0.000000000001	0.000000341690
0.5	656.102617101494	656.102616662689	656.102617101495	0.000000000001	0.000000438806
0.6	674.576004006246	674.576003465277	674.576004006247	0.000000000002	0.000000540970
0.7	693.564985457754	693.564984809376	693.564985457756	0.000000000002	0.000000648380
0.8	713.083692282365	713.083691521129	713.083692282367	0.000000000001	0.000000761237
0.9	733.146627785476	733.146626905724	733.146627785477	0.000000000002	0.000000879754
1.0	753.768676721770	753.768675717628	753.768676721772	0.000000000002	0.000001004144

Table 3: The comparative results analyses of SOIPM (2.6), RK4 (4.1), and EV (4.3) for Example 4.2 with $h = 0.1$.

x_n	SOIPM	RK4	EV	ABE_{SOIPM}	ABE_{RK4}
0.0	4.0000000000000	4.0000000000000	4.0000000000000	0.000000000000	0.000000000000
0.1	6.3877041904300	6.3869308532960	6.3876934521450	0.000010738285	0.000762598848
0.2	10.012905523178	10.010782619961	10.012879839867	0.000025683311	0.002097219906
0.3	15.268732237921	15.264702526237	15.268711060481	0.000021177440	0.004008534244
0.4	22.400329045712	22.394184281537	22.400364973206	0.000035927494	0.006180691669
0.5	31.254602812292	31.246730899601	31.254700733190	0.000097920898	0.007969833589
0.6	41.110901769627	41.102181767458	41.110934640935	0.000032871308	0.008752873477
0.7	50.834079405603	50.825520453662	50.833986027699	0.000093377904	0.008465574037
0.8	59.347386587158	59.339647124272	59.347306972859	0.000079614299	0.007659848587
0.9	66.057222691939	66.050387768979	66.057237226231	0.000014534292	0.006849457252
1.0	70.920615221808	70.914569370827	70.920665925259	0.000050703451	0.006096554432

Table 4: The FABE generated via SOIPM (2.6), RK4 (4.1) by varying the step length h for Example 4.2.

h	$FABE_{SOIPM}$	$FABE_{RK4}$
0.1	0.000050703451	0.006096554432
0.01	0.000000000046	0.000000668329
0.001	0.000000000000	0.000000000068
0.0001	0.000000000000	0.000000000000
0.00001	0.000000000000	0.000000000000
0.000001	0.000000000000	0.000000000000
0.0000001	0.000000000000	0.000000000000
0.00000001	0.000000000000	0.000000000000

Table 5: The comparative results analyses of SOIPM (2.6), RK4 (4.1), and EV (4.4) for Example 4.3 with $h = 0.1$.

x_n	SOIPM	RK4	EV	ABE_{SOIPM}	ABE_{RK4}
0.0	0.400000000000	0.400000000000	0.400000000000	0.000000000000	0.000000000000
0.1	0.424222038729	0.424222037019	0.424222038718	0.000000000011	0.000000001700
0.2	0.448813669556	0.448813666287	0.448813669530	0.000000000026	0.000000003244
0.3	0.473658134962	0.473658130255	0.473658134918	0.000000000044	0.000000004663
0.4	0.498633726438	0.498633720377	0.498633726374	0.000000000065	0.000000005996
0.5	0.523616137863	0.523616130491	0.523616137777	0.000000000086	0.000000007286
0.6	0.548480927451	0.548480918768	0.548480927346	0.000000000105	0.000000008577
0.7	0.573105985409	0.573105975371	0.573105985287	0.000000000122	0.000000009917
0.8	0.597373904008	0.597373892527	0.597373903873	0.000000000134	0.000000011346
0.9	0.621174153740	0.621174140697	0.621174153598	0.000000000142	0.000000012901
1.0	0.644404982788	0.644404968035	0.644404982645	0.000000000143	0.000000014610

Table 6: The FABEs generated via SOIPM (2.6), RK4 (4.1) by varying the step length h for Example 4.3.

h	$FABE_{SOIPM}$	$FABE_{RK4}$
0.1	0.000000000143	0.000000014610
0.01	0.000000000000	0.000000000001
0.001	0.000000000000	0.000000000000
0.0001	0.000000000000	0.000000000000
0.00001	0.000000000000	0.000000000000
0.000001	0.000000000000	0.000000000000
0.0000001	0.000000000000	0.000000000000
0.00000001	0.000000000000	0.000000000000

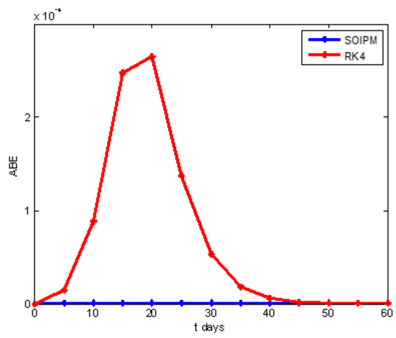

Figure 2: The plots of ABEs generated via SOIPM (2.6) and RK4 (4.1) for different values of t with $h = 0.1$ using Table 1.

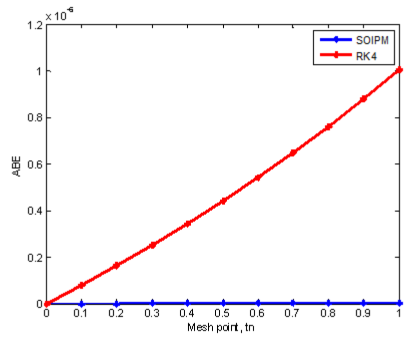

Figure 3: The plots of ABEs generated via SOIPM (2.6) and RK4 (4.1) with $h = 0.1$ using Table 2.

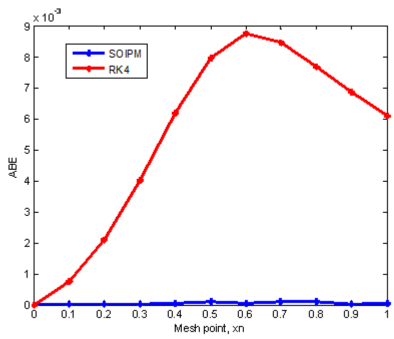

Figure 4: The plots of ABEs generated via SOIPM (2.6) and RK4 (4.1) with $h = 0.1$ using Table 3.

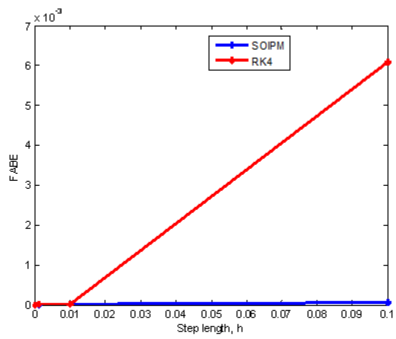

Figure 5: The plots of FABEs generated via SOIPM (2.6) and RK4 (4.1) with different step length using Table 4.

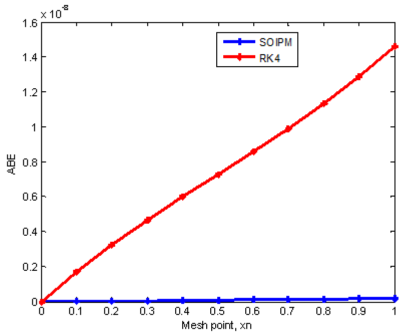

Figure 6: The plots of ABEs generated via SOIPM (2.6) and RK4 (4.1) with $h = 0.1$ using Table 5.

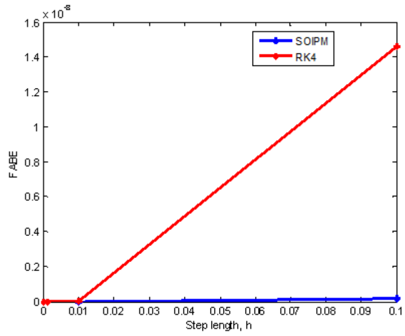

Figure 7: The plots of FABEs generated via SOIPM (2.6) and RK4 (4.1) with different step length using Table 6.

5. Discussion of results

The comparative results analyses in Tables 1 and 2 show that the numerical solution of Example 4.1 obtained via SOIPM (2.6) (with the correctly specified parameters for COVID-19 outbreak) leads to the results, which are in good agreement with the exact value for different values of t and $h = 0.1$, respectively. The ABEs generated via SOIPM (2.6) and RK4 (4.1) for different values of t and $h = 0.1$ displayed in Figures 2 and 3, respectively show that SOIPM (2.6) follows the exact value curve more elegantly and perform better than RK4 (4.1). The comparative results analyses for Examples 4.2 and 4.3 are shown in Tables 3 and 5, respectively. It is observed from Tables 3 and 5 that SOIPM (2.6) outperforms the popular RK4 (4.1). It is also observed from Figures 4 and 6 that the ABEs generated via SOIPM (2.6) are smaller than that of RK4 (4.1). The FABEs generated by SOIPM (2.6) and RK4 (4.1) are plotted in Figures 5 and 7 using Tables 4 and 6 for Examples 4.2 and 4.3, respectively. It is observed from Figures 5 and 7 that SOIPM (2.6) performs excellently and yields smaller error for every decreasing step length, h. It is also observed from Tables 4 and 6 that the sixth order accuracy/convergence of the SOIPM (2.6) has been confirmed with the step length h having a first order decrease in its magnitude, that is $h = 0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001, 0.0000001, 0.00000001$. It is clearly seen in Tables 4 and 6 that for every one-order decrease in h, there are sixth-order decrease in the magnitude of the computed final absolute errors (FABEs). The CPU time for the proposed method ranges from 3-5 secs.

6. Concluding remarks

In this paper, a new method of the family of explicit schemes with sixth order convergence/accuracy characteristics for the numerical solution of non-linear physical models with different flavours has been constructed successfully. The properties of SOIPM (2.6) have been analysed and investigated. Also the error analysis of the scheme in terms of ABEs and FABEs have been examined. Three non-linear physical models are solved via SOIPM (2.6) and RK4 (4.1). At the same time, the main computations were carried out in the context of the exact value. From the analysis of the properties, SOIPM (2.6) was found to be consistent, linearly stable, zero-stable, and thus convergent. The results show that SOIPM (2.6) compares favourably with the exact solution. The results also show that SOIPM (2.6) performs better than RK4 (4.1) and converges faster to the exact value for every one-order decrease in h which confirms the sixth order accuracy/convergence of the method. It is noteworthy to say that SOIPM (2.6) is more efficient than RK4 (4.1) because it requires less computer time unlike its counterpart that requires more computer time and four gradient evaluations for each iteration. The methodology can further be applied to two dimensional non-linear physical models emanated from real-life situations. Finally, the results were carried out via MATLAB R2014a, Version: 8.3.0.552, 32 bit (win 32) in double precision.

Acknowledgment

All helpful comments of the anonymous referees are appreciated by the authors. The authors also wish to thank Tshwane University of Technology for their financial support and the Department of Higher Education and Training, South Africa.

References

[1] O. E. Abolarin, S. W. Akingbade, *Derivation and application of fourth stage inverse polynomial scheme to initial value problems*, IAENG Int. J. Appl. Math., **47** (2017), 459–464. 1
[2] A. A. Adeniji, S. E. Fadugba, M. Y. Shatalov, *Comparative analysis of Lotka-Volterra type models with numerical methods using residuals in Mathematica*, Commun. Math. Biol. Neurosci., **2022** (2022), 1–17. 1
[3] D. Barton, *On Taylor series and stiff equations*, ACM Trans. Math. Software, **6** (1980), 280–294. 1
[4] J. C. Butcher, *Numerical methods for ordinary differential equations*, John Wiley & Sons, Chichester, (2016). 1
[5] G. Corliss, Y. F. Chang, *Solving ordinary differential equations using Taylor series*, ACM Trans. Math. Software, **8** (1982), 114–144. 1
[6] S. O. Edeki, S. E. Fadugba, V. O. Udjor, O. P. Ogundile, D. A. Dosunmu, O. Ugbenu, N. Akindele, S. C. Zelibe, T. O. Kehinde, B. A. Jonah, *Approximate-analytical solutions of the quadratic Logistic differential model via SAM*, J. Phys.: Conf. Ser., **2199** (2022), 012004. 1
[7] S. E. Fadugba, *Numerical technique via interpolating function for solving second order ordinary differential equations*, J. Math. Stat., **1** (2019) 1–6.
[8] S. E. Fadugba, *Development of an improved numerical integration method via the transcendental function of exponential form*, J. Interdiscip. Math., **23** (2020), 1347–1356.
[9] S. E. Fadugba, V. J. Shaalini, A. A. Ibrahim, *Analysis and applicability of a new quartic polynomial one-step method for solving COVID-19 model*, J. Phys.: Conf. Ser., **1734** (2021) 012019.
[10] S. E. Fadugba, *Development and analysis of a proposed scheme to solve initial value problems*, J. Math. Comput. Sci., **26** (2021), 210–221.
[11] S. E. Fadugba, J. Vinci Shaalini, O. M. Ogunmiloro, J. T. Okunlola, F. H. Oyelami, *Analysis of Exponential–Polynomial Single Step Method for Singularly Perturbed Delay Differential Equations*, *J. Phys.: Conf. Ser.* **2199** (2022), 012007.
[12] S. E. Fadugba, S. N. Ogunyebi, B. O. Falodun, *An examination of a second order numerical method for solving initial value problems*, J. Niger. Soc. Phys. Sci., **2** (2020), 120–127.
[13] S. E. Fadugba, A. E. K. Pushpam, *Development and analysis of sextic polynomial explicit method for logistic models*, Palest. J. Math., **11** (2022), 195–204. 1
[14] S. E. Fadugba, V. J. Shaalini, A. A. Ibrahim, *Development and analysis of fifth stage inverse polynomial scheme for the solution of stiff linear and nonlinear ordinary differential equations*, J. Math. Comput. Sci., **10** (2020), 2926–2942. 1
[15] M. Heydari, G. B. Loghmani, S. M. Hosseini, *Exponential Bernstein functions: an effective tool for the solution of heat transfer of a micropolar fluid through a porous medium with radiation*, Comput. Appl. Math., **36** (2017), 647–675. 1
[16] M. Heydari, G. B. Loghmani, S. M. Hosseini, *An improved piecewise variational iteration method for solving strongly nonlinear oscillators*, Comput. Appl. Math., **34** (2015), 215–249. 1
[17] M. Heydari, G. B. Loghmani, S. M. Hosseini, S. M. Karbassi,*Application of hybrid functions for solving Duffing-harmonic oscillator*, J. Differ. Equ., **2014** (2014), 9 pages. 1
[18] M. Heydari, G. B. Loghmani, A.-M. Wazwaz, *A numerical approach for a class of astrophysics equations using piecewise spectral-variational iteration method*, Int. J. Numer. Methods Heat Fluid Flow, **27** (2017), 358–378. 1
[19] S. Qureshi, A. Soomro, F. Sunday Emmanuel, E. Hincal, *A new family of L-stable block methods with relative measure of stability*, Int. J. Appl. Nonlinear Sci., **3** (2022), 197–222. 1
[20] R. B. Ogunrinde, *On a new inverse polynomial numerical scheme for the solution of initial value problems in ordinary differential equations*, Int. J. Aerosp. Mech. Eng., **9** (2015), 131–137. 1
[21] R. B. Ogunrinde, R. R. Ogunrinde, S. E. Fadugba, *Analysis of the properties of a derived one-step numerical method of a transcendental function*, J. Interdiscip. Math., **24** (2021), 2201–2213. 1
[22] K. O. Okosun, R. A. Ademiluyi, *A two-step second order inverse polynomial methods for integration of differential equations with singularities*, Res. J. Appl. Sci., **2** (2007), 13–16. 1
[23] K. O. Okosun, R. A. Ademiluyi, *Three step inverse polynomial methods for integration of differential equations with singularities*, Antarct. J. Math., **4** (2007), 121–130. 1
[24] C. Roberts, *Ordinary differential equations: applications, models, and computing*, CRC Press, (2011). 1
[25] V. J. Shaalini, S. E. Fadugba, *A new multi-step method for solving delay differential equations using Lagrange interpolation*, J. Niger. Soc. Phys. Sci., **3** (2021), 159–164. 1
[26] M. Tafakkori-Bafghi, G. B. Loghmani, M. Heydari, *Numerical solution of two-point nonlinear boundary value problems via Legendre-Picard iteration method*, Math. Comput. Simulation, **199** (2022), 133–159. 1
[27] J. Vinci Shaalini, A. E. K. Pushpam, *An application of exponential-polynomial single step method for viral model with delayed immune response*, Adv. Math.: Sci. J., **8** (2019), 154–161. 1