African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 5, Issue 2, 2022 (pp. 43-60)
www.abjournals.org

ON COMPARISON OF ALGORITHMS TECHNIQUES FOR NUMERICAL SOLUTIONS OF HIGHER ORDER DIFFERENTIAL EQUATIONS

Falade Kazeem Iyanda[1], Tiamiyu Abd'gafar Tunde[2], Edogbanya Helen Olaronke[3], Ibrahim Mujtaba Bashir[4] and Abubakar Abdullahi Shuaibu[5]

[1]Department of Mathematics, Faculty Computing and Mathematical Sciences, Kano University of Science and Technology, Wudil Kano State Nigeria
Corresponding e-mail: faladekazeem2016@kustwudil.edu.ng

[2]Department of Mathematics, Federal University of Technology Minna Niger State Nigeria

[3]Department of Mathematics, Federal University Lokoja, Kogi State Nigeria

[4]Department of Mathematics, Faculty Computing and Mathematical Sciences, Kano University of Science and Technology, Wudil Kano State Nigeria

[5]Rabiu Musa Kwankwaso, College of Advanced and Remedial Studies, School of Sciences, Department of Mathematics, Tudun Wada Kano State Nigeria

Cite this article:

Falade K.I., Tiamiyu A.T., Edogbanya H.O., Ibrahim M.B., Abubakar A.S. (2022), On Comparison of Algorithms Techniques for Numerical Solutions of Higher Order Differential Equations. African Journal of Mathematics and Statistics Studies 5(2), 43-60. DOI: 10.52589/AJMSS-52OUBAHN.

Manuscript History

Received: 5 May 2022

Accepted: 26 May 2022

Published: 16 June 2022

ABSTRACT: *Higher order differential equations appear in many different fields such as engineering, biology, chemistry, and the physical sciences. Obtaining numerical analysis solutions for different models in different applied mathematics plays an important role in explaining various applied phenomena such as beam analysis in civil engineering, pendulum mathematics in physics, thermo-liquid dynamics, stochastic modelling, plasma and nuclear physics, and nonlinear optics. In this paper, we build and implement four computational algorithms (exponential fitted collocation algorithm, differential transformation algorithm, homotopy perturbation algorithm and new iterative algorithm) for the numerical solutions of higher order differential equations from applied mathematics. The resulting numerical solutions were presented as 2D functional graphs using MAPLE 18 software to demonstrate the real world significance of the presented equations. The proposed algorithms required less computation time and the obtained results demonstrated that the algorithms are suitable for use for other higher order differential equations in applied mathematics and are efficient in terms of applications use.*

KEYWORDS: Higher order differential equations, analytic-numerical solutions, four computational algorithms, 2D function plots and 2D log function plots, MAPLE software.

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

INTRODUCTION

Differential equations are among the most important mathematical tools used in creating models in the sciences, engineering, economics, mathematics, physics, aeronautics, astronomy, dynamics, biology, chemistry, medicine, environmental sciences, social sciences, banking and many other areas [1]. The generic form of nth order differential equations may be represented as

$$\frac{d^n y(t)}{dt^n} = f\left(t, y, \frac{dy}{dt}, \frac{d^2y}{dt^2}, \frac{d^3y}{dt^3}, \frac{d^4y}{dt^4}, \dots, \frac{d^{n-1}y}{dt^{n-1}}\right) \qquad n > 1 \qquad (1)$$

with initial conditions

$$y(t_0) = \alpha\ y^{/}(t_0) = \beta\ y^{//}(t_0) = \gamma\ \vdots\ y^{(n-1)}(t_0) = \rho \qquad (2)$$

where $\alpha.\beta,\gamma,..,\rho$ are real constants.

Numerical methods provide a powerful alternative tool for solving the differential equations under the prescribed initial or boundary conditions. In the last two decades, several researchers have tried to propose and apply several numerical techniques to solve ordinary differential equations of higher orders. As such are authors [2] proposed efficient spectral Galerkin method to solve direct solvers of second and fourth order equations using Legendre polynomials; [3] presented a Jacobi spectral Galerkin method for the integrated forms of fourth order elliptic differential equations. Numerical solution of elliptic partial differential equations using radial basis function neural networks has been presented in [4]; authors [5] discussed comparison of generalizing ability on solving differential equations using back propagation and reformulated radial basis function network, [6] solved differential equations with genetic programming and evolvable machines language, [7] utilized unsupervised kernel least mean square algorithm for solving ordinary differential equations, [8] presented and applied neural algorithm for solving differential equations, [9] applied integrals of Bernstein polynomials for the solution of high even order differential equation. Approximate solution of a differential equation arising in astrophysics using the variational iteration method was presented in [10], and a new method for solving singular initial value problems in the second order ordinary differential equations was presented in [11]. The aim of this paper will be on formulation of computational algorithms using some numerical approaches developed in literature for the numerical solutions of higher order equations and obtaining maximum absolute errors for each propose technique. We outlined this paper as follows: In Section 1 brief introduction on ordinary differential equations was presented, and discussion on formulation of four algorithms was presented in Section 2. The numerical experiment on model problems and numerical results are presented in Section 3, and discussion and conclusion on the performance of the proposed algorithms are presented in Section 4.

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

African Journal of Mathematics and Statistics Studies

ISSN: 2689-5323

Volume 5, Issue 2, 2022 (pp. 43-60)

www.abjournals.org

DERIVATION OF COMPUTATIONAL ALGORITHMS

In this section, we outline the formulations of four algorithms (exponentially fitted collocation algorithm, differential transformation algorithm, homotopy perturbation algorithm and new iterative algorithm) for the numerical solution of higher order differential equation (1).

Exponentially Fitted Collocation Algorithm (EFCA)

Authors [12-13] proposed an exponentially fitted collocation approximation method using the power series as the basis function and the nth derivative of the $y(t)$ function was considered. The expansion and collection of similar terms of the equation also take into account collocation and perturbation respectively. Perturbation term $H(t) = T(N)\tau_1 + T(N-1)\tau_2 +..+T(0)\tau_n$ where $T(N)$ is a Chebyshev polynomial of degrees p and $\tau_1, \tau_2, .., \tau_n$ are free tau parameters to be determined and coupled with initial conditions (2), which eventually leads to $(N+n+1)$ algebraic linear equations in $(N+n+1)$ unknown constants. Considering the above approaches, here we construct an exponentially fitted collocation algorithm in order to improve exponentially fitted collocation approximation method for more accuracy.

restart:

Step 1:

$Digits := R^+;$

$$Computelength\ (H) := R^+;$$

Step 2:

$basis\ function := t^i;$

$$v := sum\ (y[i] * basis\ function, i = 0.., H);$$

$z[0] := eval(v, t = t_0);$

$z[1] := eval(diff(v, t), t = t_0);$

$z[2] := eval(diff(v, t\$2), t = t_0);$

$z[3] := eval(diff(v, t\$3), t = t_0);$

$$\vdots \qquad\qquad \vdots$$

$z[n-1] := eval(diff(v, t\$n-1), t = t_0);$

Step 3:

$$T[0] := 1;$$

$$T[1] := 2 * t + 1;$$

$$for\ n\ from\ 1\ to\ \ p\ do$$

$$T[n+1] := simplify(2 * (2 * t - 1) * T[n] - T[n-1]);$$

end do

Step 4:

$$Y := t^N;$$

$$n := order\ \ of\ IVP;$$

$$for\ i\ from\ 0\ to\ \ N\ do$$

$V := \left(\frac{d^n y(t)}{dt^n} - f\left(t, y, \frac{dy}{dt}, \frac{d^2y}{dt^2}, \frac{d^3y}{dt^3}, \frac{d^4y}{dt^4}, \dots, \frac{d^{n-1}y}{dt^{n-1}}\right)\right) * y(i);$

$A[i] := eval(V, N = i);$

do

$X := \big(sum(A[j], j = 0 \dots H) - sum(T[\,p-k] * \tau[k+1], k = 0 \dots n-1)\big) = f(t);$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

Step 5:

$$for\ m\ from\ 1\ to\ \ N+1\ do$$
$$eqn[m] := eval\left(T, t = \frac{m}{H+2}\right);$$
$$end\ do$$
$$eqn[H+2] := z[0] + e^{t_0}\tau[n] = \alpha;$$
$$eqn[H+3] := z[1] + e^{t_0}\tau[n] = \beta; \vdots$$
$$eqn[H+n+1] := z[n-2] + e^{t_0}\tau[n] = \rho;$$

Step 6:

$$sys := seq(eqn[i], i = 1 \ldots\ H+n+1);$$
$$sol := evalf\big(solve(\{sys\})\big);$$
$$Q := eval\big(\big[seq(y[i], i = 0 \ldots\ H), \tau[n]\big], sol\big);$$
$$for\ r\ from\ 1\ to\ \ H\ do$$
$$y[\,r] := Q[r+1];$$
$$end\ do$$
$$\tau[\,n] := Q[p+2];$$

Step 7:

$$Solution := sum\big(y[j] * t^j, j = 0 \ldots H\big) + \tau[n] * t;$$
$$for\ t\ from\ 0\ by\ 0.1\ to\ 1\ do$$
$$EFCA[t] := evalf\ (eval(Solution, t)$$
$$end\ do$$
$$[2Dplot] := plot([EFCA], t = 0 \ldots 1, color[green, red, yellow, plackpurple], axes = boxed, title = IVPs);$$

where n is the order of the IVP

Differential Transformation Algorithm (DTA)

Zhou first introduced the concept of the differential transformation method. This is an iterative procedure to obtain the Taylor series analytical solutions of the ordinary differential equation (ODE), the system of ordinary differential equations (SODE), and the partial differential equation (PDE). The commendable point of this method is that it can reduce the size of the computational work while the accuracy and convergence speed of the serial solution are still provided [14-16]. The basic math operations are performed by the method of differential transformation as shown below:

Table 1: One dimensional differential transformation function $y(t)$

Functional Form	Transformed Form
$y(t) = w(t) \pm z(t)$	$Y(k) = W(k) \pm Z(k)$
$y(t) = \dfrac{dy(t)}{dt}$	$Y(k) = (k+1)Y(k+1)$
$y(t) = \dfrac{d^n y(t)}{dt^n}$	$Y(k) = (k+1)\cdots(k+n)Y(k+n)$
$y(t) = e^t$	$Y(k) = \dfrac{1}{k!}$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

$y(t) = w(t) + v(t)$	$Y(k) = \sum_{r=0}^{\infty} W(r)V(k-r)$
$y(t) = y^p$	$Y(k) = \delta(k-p) = \{1, \quad k = n\ 0, \quad otherwise$
$y(t) = \epsilon y(t)$	$Y(k) = \epsilon Y(k) \quad where\ \epsilon\ is\ a\ constant$
$y(t) = \{cos\ (\varepsilon y + \omega)\ sin\ (\varepsilon y + \omega)$	$Y(k) = \frac{\varepsilon^k}{k!} cos\left(\frac{k\pi}{2} + \omega\right), X_1(k) = \frac{\varepsilon^k}{k!} sin\left(\frac{k\pi}{2} + \omega\right)\ where\ \epsilon, are\ constants$

Reducing computational time taken in simplifying and evaluating the derivatives involves transformation differential method (DTM) while the accuracy is desired; hence, formulation of differential transform algorithm goes thus:

restart:

Step 1:

$$Digits := R^+$$
$$n := order\ \ of\ IVP;$$
$$Y(0) := \alpha;$$
$$Y(1) := \beta;$$
$$Y(2) := \gamma;$$
$$\vdots \qquad \vdots$$
$$Y(n-1) := \rho;$$
$$\delta[0] := 1;$$
$$\delta[-1] := 1;$$
$$for\ n\ from\ 1\ to\ N\ do$$
$$\delta[n] := 0$$
$$end\ do:$$

Step 2:

$$for\ k\ from\ 0\ to\ N\ do$$
$$Y[k+n] := -\frac{1}{(k+1)(k+2)(k+3)\cdots(k+n)} * (k+1)Y[k+1] + \delta[k] + f(t));$$
$$end\ do;$$

Step 3:

$$y(t) := sum\big(Y[j] * t^j, j = 0, \cdots, N+1\big);$$
$$DTA := y(t);$$
$$for\ i\ from\ 0\ by\ 0.1\ to\ 1\ do$$
$$y(i) := evalf\big(eval(DTA, t = i)\big)$$
$$end\ do;$$
$$[2Dplot] := plot([DTA], t = 0 \ldots 100, color[green, red, yellow, plackpurple], axes = boxed, title = IVPs);$$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

Homotopy Perturbation Algorithm (HPA)

Homotopy perturbation method was proposed by JiHuan He and used by scientists and engineers to solve many problems in engineering and science; it has been found to be extremely effective in solving these problems. To illustrate the basic ideas of the isomorphic perturbation method, there is a coupling between the traditional perturbation method and the homomorphism, which is a very interesting and useful concept in topology and repeatedly turns into a simple, easy-to-solve problem [21-23]. To reduce the computational time required to simplify and evaluate the integral use in the homotopy perturbation method (HPM), while precision is desired, the formula of the homotopy perturbation algorithm as follows:

restart:

Step 1:

$$Digits := R^{+};$$
$$n := order\ \ of\ IVP;$$
$$N := Computational\ lenght;$$

$f := f(t);$

$$HPA(0) := \alpha + \beta * t + \frac{\gamma * t^2}{2!} + \cdots + \frac{\rho * t^{n-1}}{(n-1)!};$$

$HPA[1] := -value\big(int(diff(HPA[0], t\$n) + (HPA[0], t\$n - 1) + \cdots + HPA[0] - f\big), [\,t = 0 \ldots t, t = 0 \ldots t, \ldots, t = 0 \ldots t]);$

Step 2:

$$for\ n\ from\ 2\ to\ N\ do$$
$$HPA[n] := -Value\left(int\big(diff(HPA[n-1], [\,t = 0 \ldots t, t = 0 \ldots t, \ldots, t = 0 \ldots t])\big)\right);$$

$end\ do;$

Step 3:

$$sol1 := sum\big((HPA[j]), j = 0..N+1\,\big);$$
$$for\ t\ \ from\ 0\ by\ 0.1\ to\ 1\ do$$
$$HPA[t] := evalf\big(eval(sol1)\big)$$
$$end\ do;$$
$$[2Dplot] := plot([HPA], t = 0 \ldots 100, color[green, red, yellow, plackpurple], axes = boxed, title = IVPs);$$

New Iterative Algorithm (NIA)

A new iterative method was proposed by Daftardar-Gejji and Jafari [24], it is an efficient technique for solving linear and nonlinear functional differential equations which arises in a wide application in nonlinear problems without linearization or small perturbation. NIM is simple in principles and easy to implement; it gives results which are in good agreement with other methods and in many cases with few iterations [25-28]. The formulation of new iterative algorithm can be stated as follows:

restart:

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 5, Issue 2, 2022 (pp. 43-60)

Step 1:

$$Digits \coloneqq R^+;$$
$$n \coloneqq order\ \ of\ IVP;$$
$$N \coloneqq Computational\ length;$$

$f \coloneqq f(t);$

$$NIA[0] \coloneqq \alpha + \beta * t + \frac{\gamma * t^2}{2!} + \cdots + \frac{\rho * t^{n-1}}{(n-1)!};$$

Step 2:

$$for\ n\ from\ to\ N\ do$$

$$NIA[n+1] \coloneqq -value\left(\left(int\big((NIA[n])\big)\right)\right), [\,t = 0 \dots t, t = 0 \dots t, \dots, t = 0 \dots t]);$$

$$sol1 \coloneqq sum\big((NIA[j]), j = 0..N+1\,\big);$$
$$end\ do;$$

Step 3:

$$for\ t\ \ from\ 0\ by\ 0.1\ to\ 1\ do$$
$$NIA[t] \coloneqq evalf\big(eval(sol1)\big)$$
$$end\ do;$$
$$[2Dplot] \coloneqq plot([NIA], t = 0 \dots 100, color[green, red, yellow, plackpurple], axes = boxed, title = IVPs);$$

Numerical Experiments

To illustrate the effectiveness of the proposed algorithms, four test problems are considered in this section and presentations of graphical simulations of all the problems are obtained in 2D plots.

Problem 1

Consider second order initial value problem of the form [29]

$$\frac{d^2y}{dt^2} + \frac{2\Phi dy}{\Psi dt} + \frac{1}{\Psi^2}y = \frac{\varpi}{\Psi^2}u(t) \tag{3}$$

With initial conditions,

$$y(0) = y^{/}(0) = 0 \tag{4}$$

Here, Φ, Ψ, ϖ are parameter constants and $u(t) = sin\ sin\ (\Omega t)$ is a sinusoidal input. In order to apply the four proposed algorithms discussed in the last section for the numerical solutions of equations (3) and (4), the following parameters are considered:

$$\{\Psi = \frac{2}{3}\ \varpi = \frac{5}{7}\ \Omega = 1\ \Phi = \left[\frac{1}{2}, \frac{1}{4}, \frac{1}{6}, \frac{1}{8}, \frac{1}{10}\right]\ and\ \Phi = \left[\frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}, \frac{1}{11}\right]$$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 5, Issue 2, 2022 (pp. 43-60)

Use of computational length as $N = 15$ was considered, and numerical solutions obtained with absolute errors are presented in Table 2.

Table 2: Numerical solutions and absolute errors when $\boldsymbol{\Phi = \frac{1}{2}}$ even and $\boldsymbol{\Phi = \frac{1}{3}}$ odd fractional parameter constants

T		$\Psi = \frac{2}{3}, \varpi = \frac{5}{7}, \Omega = 1, \Phi = \frac{1}{2}$	E_t	$\Psi = \frac{2}{3}, \varpi = \frac{5}{7}, \Omega = 1, \Phi = \frac{1}{3}$	E_t
0	Exact	0.000000000000000000000	0.00-E	0.000000000000000000000	0.00-E
	EFCA	0.000000000000000000000	0.00-E	0.000000000000000000000	0.00-E
	DTA	0.000000000000000000000	0.00-E	0.000000000000000000000	0.00-E
	HPA	0.000000000000000000000	0.00-E	0.000000000000000000000	0.00-E
	NIA	0.000000000000000000000	0.00-E	0.000000000000000000000	0.00-E
0.1	Exact	0.000257689321705731538 5		0.00026086946484147379 58	
	EFCA	0.0002576893217057276200	3.9E-18	0.0002608694648414691600	4.6E-18
	DTA	0.0002576893217057275300	4.0E-18	0.0002608694648414691300	4.7E-18
	HPA	0.0002576893217057276600	3.8E-18	0.0002608694648414691800	4.8E-18
	NIA	0.0002576893217057277700	3.9E-16	0.0002608694648414691900	4.9E-18
0.2	Exact	0.0019785365268224204020		0.002026719487765158689 3	
	EFCA	0.0019785365268221993800	2.2-16E	0.0020267194877650133000	1.5-16E
	DTA	0.0019785365268221897300	2.3-16E	0.0020267194877650141000	1.6-16E
	HPA	0.0019785365268221723400	2.5-16E	0.0020267194877650154000	1.4-16E
	NIA	0.00197853652682216234400	2.6-16E	0.0020267194877650176000	1.4-16E
0.3	Exact	0.0063936201518928491164 0		0.0066239083151803176044 0	
	EFCA	0.0063936201518911375100 0	1.6-15E	0.0066239083151783327700 0	1.9-15E
	DTA	0.0063936201518911376300 0	1.7-15E	0.0066239083151783338000 0	2.0-15E
	HPA	0.0063936201518911378200 0	1.8-15E	0.0066239083151783349000 0	2.1-15E
	NIA	0.0063936201518911379800 0	1.9-15E	0.0066239083151783356700 0	2.2-15E
0.4	Exact	0.0144766246975923140700 0		0.0151616630340572235480 0	
	EFCA	0.0144766246975869409400 0	5.3-15E	0.0151616630340362663600 0	2.1-14E
	DTA	0.0144766246975869408200 0	5.4-15E	0.0151616630340362664500 0	2.2-14E
	HPA	0.0144766246975869407600 0	5.5-15E	0.0151616630340362666400 0	2.3-14E
	NIA	0.0144766246975869406500 0	5.6-15E	0.0151616630340362668500 0	2.4-14E
0.5	Exact	0.0269449892463144661680 0		0.0285142796898842878590 0	
	EFCA	0.0269449892463129382600 0	1.5-15E	0.0285142796897205157800 0	1.6-13E
	DTA	0.0269449892463129384500 0	1.6-15E	0.0285142796897205158800 0	1.7-13E
	HPA	0.0269449892463129386700 0	1.7-15E	0.0285142796897205159900 0	1.8-13E
	NIA	0.0269449892463129388200 0	1.8-15E	0.0285142796897205159800 0	1.9-13E
0.6	Exact	0.0442666808894155540070 0		0.0473104893177442848450 0	
	EFCA	0.0442666808894803747700 0	6.4-14E	0.0473104893167924378100 0	9.5-13E
	DTA	0.0442666808894803748200 0	6.5-14E	0.0473104893167924378900 0	9.6-13E
	HPA	0.0442666808894803748900 0	6.6-14E	0.0473104893167924379400 0	9.7-13E
	NIA	0.0442666808894803749700 0	6.7-14E	0.0473104893167924379900 0	9.8-13E
0.7	Exact	0.0666718076746958227680 0		0.0719296781607998557160 0	
	EFCA	0.0666718076750612527900 0	3.6-13E	0.0719296781564510671800 0	4.3-12E
	DTA	0.0666718076750612528200 0	3.7-13E	0.0719296781564510672200 0	4.4-12E

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 5, Issue 2, 2022 (pp. 43-60)
www.abjournals.org

	HPA	0.066671807675061252890000	3.8-13E	0.071929678156451067340000	4.5-12E
	NIA	0.066671807675061252990000	3.9-13E	0.071929678156451067450000	4.6-12E
0.8	Exact	0.094168288444036273946000		0.102504527595096951290000	
	EFCA	0.094168288445303187620000	1.3-13E	0.102504527578640377360000	1.6-11E
	DTA	0.094168288445303187740000	1.2-13E	0.102504527578640377430000	1.6-11E
	HPA	0.094168288445303187860000	1.3-13E	0.102504527578640377520000	1.6-11E
	NIA	0.094168288445303187960000	1.3-13E	0.102504527578640377670000	1.7-11E
0.9	Exact	0.126560814206139667290000		0.138929536122944160160000	
	EFCA	0.126560814209373437870000	3.2-12E	0.138929536122944160240000	1.0-0E
	DTA	0.126560814209373437920000	3.2-12E	0.138929536122944160360000	1.0-0E
	HPA	0.126560814209373437950000	3.2-12E	0.138929536122944160450000	1.0-0E
	NIA	0.126560814209373437990000	3.2-12E	0.138929536122944160570000	1.0-0E
1.0	Exact	0.163472365255381710840000		0.180874802496680178458070	
	EFCA	0.163472365261346274180000	5.9-12E	0.180874802496680178459100	0.0-0E
	DTA	0.163472365261346274220000	5.9-12E	0.180874802496680178459300	0.0-0E
	HPA	0.163472365261346274310000	5.9-12E	0.180874802496680178459420	0.0-0E
	NIA	0.163472365261346274400000	5.9-12E	0.180874802496680178459660	0.0-0E

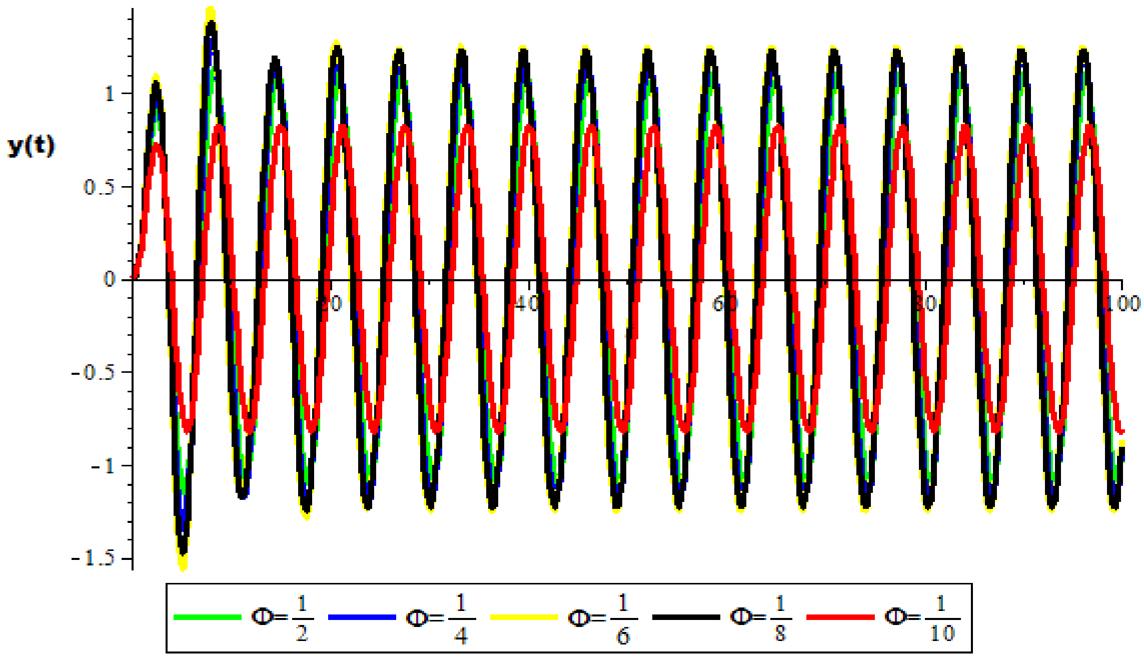

Figure 1 depicts the numerical simulation of y(t) obtained when varied fractional constants are even $\Phi = \left[\frac{1}{2}, \frac{1}{4}, \frac{1}{6}, \frac{1}{8}, \frac{1}{10}\right]$ and other two parameters remain fixed $\Psi = \frac{2}{3}, \varpi = \frac{5}{7}$ on interval $0 \leq t \leq 100$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 5, Issue 2, 2022 (pp. 43-60) www.abjournals.org

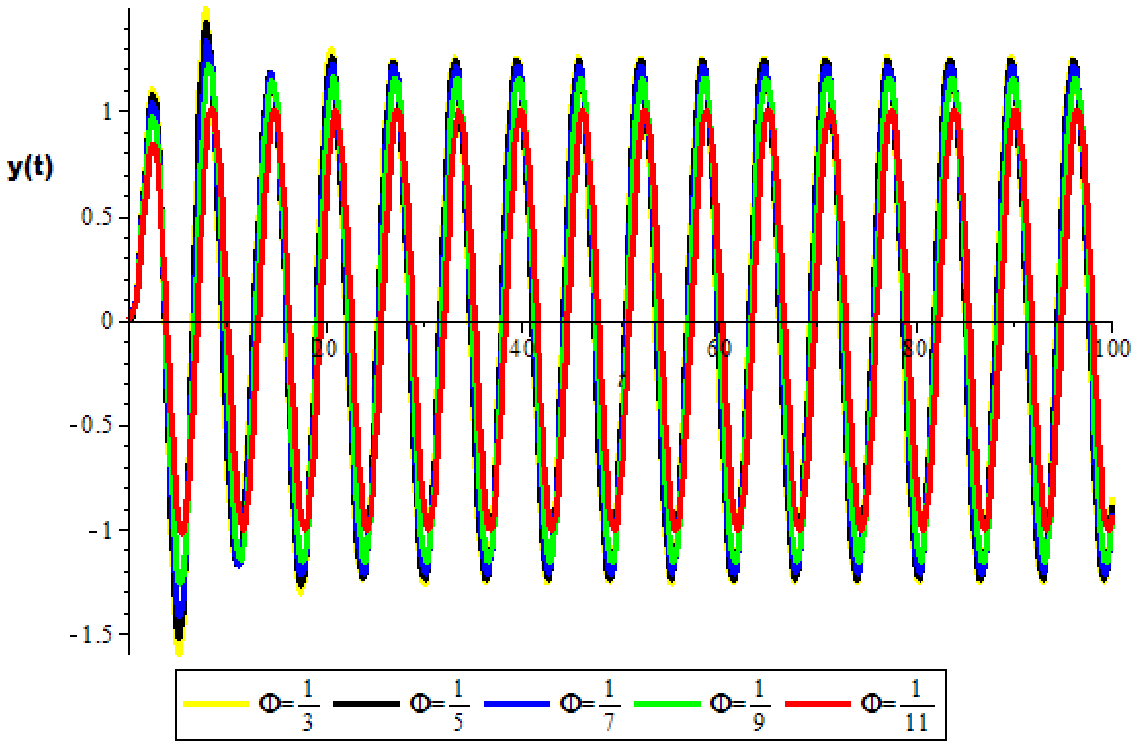

Figure 2 depicts the numerical simulation of y(t) obtained when varied fractional constants are even $\Phi = \left[\frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}, \frac{1}{11}\right]$ and other two parameters remain fixed $\Psi = \frac{2}{3}, \varpi = \frac{5}{7}$ on interval $0 \leq t \leq 100$

Problem 2

Consider third order initial value problem of the form [30]

$$\frac{d^3y}{dt^3} = -e^t \tag{5}$$

with initial conditions

$$y(0) = 1\ y^{/}(0) = -1\ y^{//}(0) = 3 \tag{6}$$

Exact solution is

$$y(t) = 2 + 2t^2 - e^{-t} \tag{7}$$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 5, Issue 2, 2022 (pp. 43-60)
www.abjournals.org

Table 3: Numerical solutions and absolute errors for problem 2

t	Exact solution	$EFCA_e$	DTA_e	HPA_e	NIA_e	[30]	[31]
0	1.00000000000000000000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.1	0.91482908192435237507	1.3E-19	1.4E-18	1.3E-19	1.8E-18	3.4E-14	2.5E-13
0.2	0.85859724183983016925	3.5E-18	2.3E-18	2.9E-18	2.7E-18	3.3E-13	6.5E-11
0.3	0.83014119242399698479	8.9E-17	1.6E-17	1.7E-18	1.6E-17	3.7E-14	3.4E-09
0.4	0.82817530235873050246	8.2E-16	2.2E-15	1.8E-18	3.6E-17	5.8E-13	8.2E-09
0.5	0.85127872929987656116	4.7E-15	1.9E-15	5.9E-16	1.6E-16	3.6E-13	6.0E-09
0.6	0.89788119960951132335	2.0E-14	2.1E-14	5.1E-15	1.4E-15	1.2E-12	9.9E-09
0.7	0.96624729252959539300	7.2E-14	1.3E-14	4.6E-15	3.4E-14	1.2E-12	1.5E-08
0.8	1.05445907150775263 50	2.2E-13	2.4E-12	1.6E-13	4.1E-13	2.5E-12	2.2E-08
0.9	1.16039688884365241 70	6.0E-13	1.0E-12	5.1E-13	3.1E-13	2.5E-12	3.5E-08
1.0	1.28171817154245745 09	1.5E-12	1.2E-12	3.4E-12	1.7E-12	4.9E-12	6.0E-08

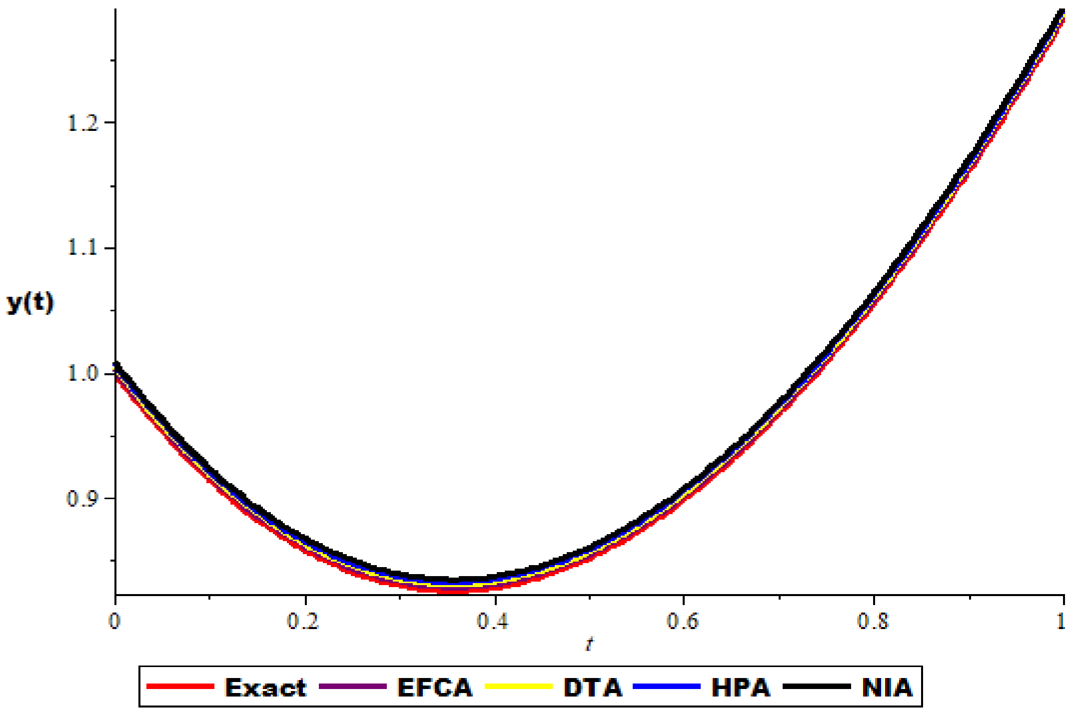

Figure 3 depicts the numerical solution on interval $0 \leq t \leq 1$

Problem 3

The ill-posed problem has an important engineering application in viscoelastic flow, a beam on elastic foundation. Fourth-order ODEs can be used to describe the ill-posed problem of a beam on elastic foundation as follows [32]:

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

$$\frac{d^4y}{dt^4} + \Psi y = f(t) \tag{8}$$

with initial conditions

$$y(0) = y^{/}(0) = y^{//}(0) = y^{///}(0) = y^{///}(0) = 0 \tag{9}$$

Here, Ψ is a beam constant.

Considering $\Psi = [1, 2,3,4,5]$ and $f(t) = 1$, we obtained numerical solutions presented in Table 4

Table 4: Numerical solutions and absolute errors

T		$\Psi = 1$	E_t	$\Psi = 2$	E_t
0	Exact	0.000000000000000000000	0.0E-00	0.000000000000000000000	0.0E-00
	EFCA	0.000000000000000000000	0.0E-00	0.000000000000000000000	0.0E-00
	DTA	0.000000000000000000000	0.0E-00	0.000000000000000000000	0.0E-00
	HPA	0.000000000000000000000	0.0E-00	0.000000000000000000000	0.0E-00
	NIA	0.000000000000000000000	0.0E-00	0.000000000000000000000	0.0E-00
0.1	Exact	0.000004166664186508100		0.000004166666170634930	
	EFCA	0.000004166664186507975	1.2E-20	0.000004166666170634951	2.2E-20
	DTA	0.000004166664186507957	1.4E-20	0.000004166666170634928	2.1E-21
	HPA	0.000004166664186507959	1.4E-20	0.000004166666170634944	1.4E-20
	NIA	0.000004166664186507960	1.4E-20	0.000004166666170634932	2.3E-21
0.2	Exact	0.000066666031746117200		0.000066666539682573860	
	EFCA	0.000066666031746117590	4.1E-20	0.000066666539682574290	4.3E-21
	DTA	0.000066666031746117257	1.1E-20	0.000066666539682573887	3.1E-20
	HPA	0.000066666031746117259	1.1E-20	0.000066666539682573896	4.1E-21
	NIA	0.000066666031746117356	2.1E-20	0.000066666539682573899	4.1E-20
0.3	Exact	0.000337498372768966300		0.000337496745540152180	
	EFCA	0.000337498372768966926	1.1E-19	0.000337496745540153283	1.1E-18
	DTA	0.000337498372768966293	0.7E-19	0.000337496745540153191	1.0E-18
	HPA	0.000337498372768966433	0.0E-20	0.000337496745540153341	1.1E-18
	NIA	0.000337498372768966563	1.1E-19	0.000337496745540153456	1.3E-18
0.4	Exact	0.001066650412733438050 0		0.001066634158870260110	
	EFCA	0.001066650412733438028 2	0.0E-00	0.001066634158870260198	0.0E-00
	DTA	0.001066650412733438074 5	0.0E-00	0.001066634158870260274	0.0E-00
	HPA	0.001066650412733438084 1	0.0E-00	0.001066634158870260354	0.0E-00
	NIA	0.001066650412733438090 0	0.0E-00	0.001066634158870260434	0.0E-00
0.5	Exact	0.002604069785975955580 0		0.002603972906304612980	
	EFCA	0.002604069785975955976 1	0.0E-00	0.002603972906304629971	1.7E-17
	DTA	0.002604069785975955291 2	1.1E-18	0.002603972906304628815	1.6E-17
	HPA	0.002604069785975955321 2	1.1E-18	0.002603972906304642811	3.0E-17
	NIA	0.002604069785975955351 1	1.1E-18	0.002603972906304634515	2.2E-17

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

0.6	Exact	0.0053995834331158306512		0.005399166875320411610	
	EFCA	0.0053995834331158336744	3.1E-18	0.005399166875320447650	3.6E-17
	DTA	0.0053995834331158441558	1.3E-17	0.005399166875320519480	1.6E-16
	HPA	0.0053995834331158461657	1.5E-17	0.005399166875320529991	1.2E-16
	NIA	0.0053995834331158451559	1.4E-17	0.005399166875320534850	1.2E-16
0.7	Exact	0.0100027369334098489406		0.010001307257944316169	
	EFCA	0.0100027369334098546777	0.0E-00	0.010001307257944384478	6.1E-17
	DTA	0.0100027369334098582287	1.1E-17	0.010001307257944384828	6.1E-17
	HPA	0.0100027369334098583281	1.1E-17	0.010001307257944384327	6.1E-17
	NIA	0.0100027369334098583117	1.1E-17	0.010001307257943848451	4.7E-17
0.8	Exact	0.0170625057942562871000		0.017058345208765799034	
	EFCA	0.0170625057942562970579	1.1E-17	0.017058345208765918286	1.1E-14
	DTA	0.0170625057942573324095	1.0E-15	0.017058345208776561384	1.1E-14
	HPA	0.0170625057942574323435	1.1E-15	0.017058345208776551238	1.1E-14
	NIA	0.0170625057942574568915	1.2E-15	0.017058345208776561388	1.1E-14
0.9	Exact	0.0273268243195231377600		0.027316149818783478146	
	EFCA	0.0273268243195231356717	0.0E-00	0.027316149818753499991	2.9E-14
	DTA	0.0273268243195331245673	9.9E-15	0.027316149818764532173	1.9E-14
	HPA	0.0273268243195332456772	1.1E-14	0.027316149818767634535	1.6E-14
	NIA	0.0273268243195333354632	1.0E-14	0.027316149818306548298	4.7E-13
1.0	Exact	0.0416418671669929837800		0.041617071842383935610	
	EFCA	0.0416418671669928765321	1.0E-16	0.041617071842383954632	1.1E-17
	DTA	0.0416418671669995432215	6.6E-15	0.041617071842388332456	4.4E-16
	HPA	0.0416418671667846512854	2.1E-15	0.041617071842387673576	3.7E-15
	NIA	0.0416418671667542389432	2.4E-15	0.041617071842345765417	3.8E-15

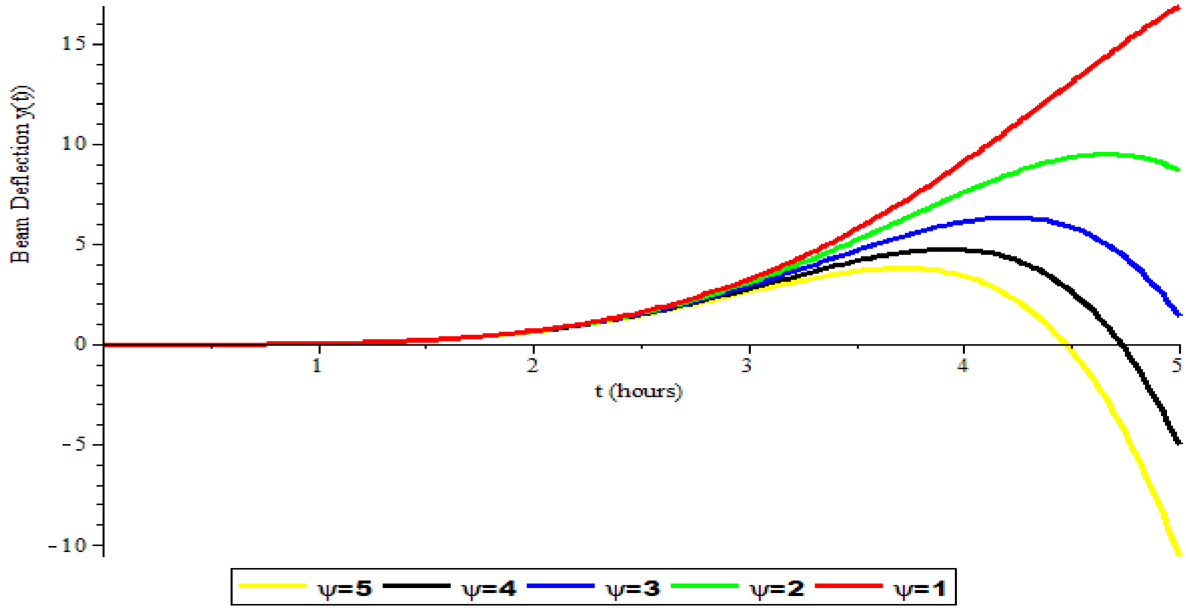

Figure 4 depicts deflection distance obtain when variable $\Psi = [1,2,3,4,5]$ on time interval $0 \leq t \leq 5$

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

Problem 4

Consider fifth order homogenous initial value problem of the form [33],

$$\frac{d^5y}{dt^5} - \Omega\frac{d^3y}{dt^3} + (\Omega - 1)y = 0 \tag{10}$$

With initial conditions

$$y(0) = 3\ y^{/}(0) = -5\ y^{//}(0) = 11\ y^{///}(0) \ = -23\ y^{/V}(0) \ = 47 \tag{11}$$

Here, Ω is a constant.

Considering $\Omega = [5,6,7,8,9,10]$ for numerical simulation of equation (10) coupled with initial conditions given in equation 2, we obtained numerical solutions presented in Table 5.

Table 5: Numerical solutions and absolute errors for problem 4

t	Exact solution $\Omega = 5$	$EFCA_e$	DTA_e	HPA_e	NIA_e	[33]	[34]
0	3.0000000000000000000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.1	2.5513548411979858000	1.5E-16	2.7E-15	2.8E-15	2.3E-15	2.6E-15	3.1E-15
0.2	2.1922293850289365000	1.3E-15	2.4E-15	3.8E-15	2.3E-15	1.9E-14	2.4E-14
0.3	1.9056166876003631000	2.2E-15	2.5E-14	4.5E-14	2.9E-13	2.2E-12	6.9E-12
0.4	1.6776668463160274000	3.7E-12	1.6E-13	2.7E-13	3.6E-12	1.2E-11	1.5E-10
0.5	1.4971076638016956000	3.8E-12	4.4E-11	3.9E-11	6.8E-11	4.6E-11	1.7E-09
0.6	1.3547709996425796000	4.7E-12	2.6E-10	1.9E-11	4.8E-11	1.4E-10	1.2E-08
0.7	1.2432055880334085000	3.8E-11	4.2E-10	3.4E-10	3.9E-10	3.4E-10	6.6E-08
0.8	1.1563605898667424000	3.7E-11	6.5E-10	7.7E-10	6.6E-10	7.4E-10	2.8E-07
0.9	1.0893270049241579000	2.7E-11	3.8E-10	2.7E-10	4.7E-10	1.5E-09	1.0E-06
1.0	1.0381264085383930000	3.6E-10	5.9E-10	3.6E-09	3.6E-09	2.8E-09	3.6E-06

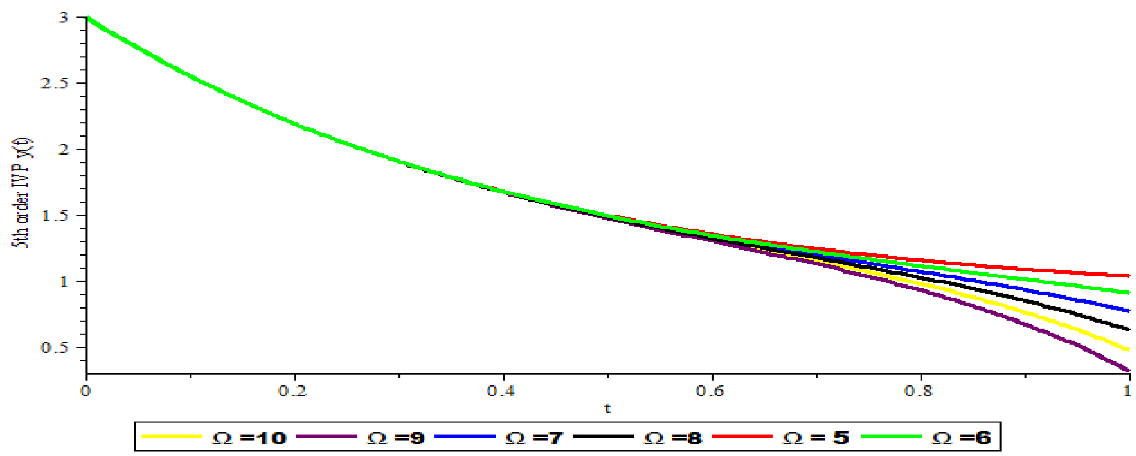

Figure 5 depicts the numerical simulation results obtained when variable $\Omega = [5,6,7,8,9,10]$ on interval $0 \leq t \leq 1$.

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

DISCUSSION AND CONCLUSION

Discussion of Results

The present paper is aimed to present numerical comparison of four proposed computational algorithms for the numerical solution of higher order differential equations which occur in applied mathematics. Four problems are considered to test the feasibility and practicability of the techniques and the following observations are noted:

i. **Problem 1**: The proposed techniques demonstrated significantly with exact solutions and highest results are obtained when constant parameter Φ is odd (see Table 2).

ii. Figure 1 depicts the frequency pattern of the even constant parameter Φ and the highest and lowest values are obtained at $\Phi = \frac{1}{6}$ (see Figure 1).

iii. Figure 2 depicts the frequency pattern of the odd constant parameter Φ and the highest and lowest values are obtained at $\Phi = \frac{1}{3}$ (see Figure 2).

iv. **Problem 2**: The absolute errors obtained for all proposed algorithms are better compared with exact solutions and the available literature (see Table 3 and Figure 3).

v. **Problem 3**: The algorithms demonstrated good agreement with exact solutions and increased the constant parameter Ψ which yielded less beam deflection distance (see Table 3 and Figure 4).

vi. **Problem 4**: Better absolute errors are obtained and the constant parameter Ω demonstrated uniform decrease in numerical solutions until small changes at $\Omega = 9$ (Figure 5).

CONCLUSION

The present work is aimed to formulate and apply effective computation algorithms: exponential fitted collocation algorithm, differential transformation algorithm, homotopy perturbation algorithm and new iterative algorithm to solve higher order initial values problems which arise in applied mathematics. Numerical solutions and graphical presentations of the numerical results demonstrated effectiveness and good accuracy when compared with exact solutions. From the computational viewpoint, the proposed algorithms are feasible, easy, fast and effective and are good approaches in teaching and learning in applied computational mathematics.

Article DOI: 10.52589/AJMSS-52OUBAHN

DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

Abbreviations

The following abbreviations are used in this paper:

$EFCA_e$ = Absolute error in exponentially fitted collocation algorithm

DTA_e = Absolute error in differential transformation algorithm

HPA_e = Absolute error in Homotopy perturbation algorithm

NIA_e = Absolute error in new iterative algorithm.

REFERENCES

[1]. Butcher, J. Numerical Methods for Ordinary Differential Equations. West Sussex: John Wiley & Sons Ltd.(2003), 45-95.

[2] Shen, J. Efficient spectral-Galerkin method. I. Direct solvers of second- and fourth-order equations using Legendre polynomials. SIAM Journal on Scientific Computing, 15(6).(1994), pp 1489-1505.

[3] Doha, E.H. & Bhrawy, A.H. A Jacobi spectral Galerkin method for the integrated forms of fourth-order elliptic differential equations. Numerical Methods for Partial Differential Equations, 25(3).(2009), pp 712-739.

[4] Jianyu, L., Siwei, L, Yingjian, Q. & Yaping, H. Numerical solution of elliptic partial differential equation using radial basis function neural networks. Neural Networks, 16(5-6). (2003), pp 729-734.

[5] Choi, B. & Lee, J.H. Comparison of generalization ability on solving differential equations using backpropagation and reformulated radial basis function networks. Neurocomputing, 73(1-3). (2009), pp 115-118

[6] Tsoulos, I.G. & Lagaris, I.E. Solving differential equations with genetic programming. and evolvable machines language, 7(1). (2006), pp 33-54

[7] Yazdi, H.S., Pakdaman, M. & Modaghegh, H. Unsupervised kernel least mean square algorithm for solving ordinary differential equations. Neurocomputing, 74(12-13). (2011), pp 2062-2071.

[8] Lee. H. & Kang, I.S. Neural algorithm for solving differential equations. Journal of Computational Physics, 91(1).(1990), pp 110-131.

[9] Doha, E.H., Bhrawy, A.H. & Saker, M.A.Integrals of Bernstein polynomials: an application for the solution of high even-order differential equations. Applied Mathematics Letters, 24(4). (2011),pp 559-565.

[10] Dehghan. M, Shakeri. F, Approximate solution of a differential equation arising in astrophysics using the variational iteration method, New Astron. 13 (2008) 53–59.

[11] Wazwaz. A.M, A new method for solving singular initial value problems in the second-order ordinary differential equations, Appl. Math. Comput. 128 (2002) 45–57.

[12] Falade K.I, Abubakar A.S. Solving Bessel differential equation of order zero using exponentially fitted collocation approximation method, Research Journal of Mathematical and Statistical Sciences Vol. 7(2), 21-26, May (2019).

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN

[13] Falade K.I, Baoku I.G, A, Tiamiyu A.T and Isyaku I On numerical computational solution of
seventh order boundary value problems, Journal of the Nigerian Mathematical Society, Vol. 39, Issue 2, (2020) pp. 255-268.
[14] Zhou J. K Differential Transformation and Its Applications for Electrical Circuits, Huazhong University Press, Wuhan, China. 1986.
[15] Farshid M. I Differential Transform Method for Solving Linear and Nonlinear Systems of Ordinary Differential Equations, Applied Mathematical Sciences, Vol. 5, no. 70, (2011) 3465 – 3472.
[16] Biazar, J. Eslami M, Differential Transform Method for Quadratic Riccati Differential Equation, International Journal of Nonlinear Science, 9 (4) (2010) 444–447
[17] He,J.-H. Homotopy Perturbation Technique. Computer Methods in Applied Mechanics and Engineering, 178, (1999), 257-262.
[18] He, J.-H A Coupling Method of a Homotopy Technique and a Perturbation Technique for Non-Linear Problems. International Journal of Non-Linear Mechanics, 35, . (2000) 37-43.
[19] He, J.-H. Homotopy Perturbation Method: A New Nonlinear Analytical Technique. Applied Mathematics and Computation, 135, (2003) 73-79
[20] He, J.-H An Elementary Introduction to the Homotopy Perturbation Method. Computers and Mathematics with Applications, 57, (2009) 410-412.
[21] Ghorbani, A. and Saberi-Nadjafi, J. He's Homotopy Perturbation Method for Calculating Adomian Polynomials. International Journal of Nonlinear Sciences and Numerical Simulation, 8, (2007), 229-232.
[22] Khan, Y. and Wu, Q. Homotopy Perturbation Transform Method for Solving Nonlinear Equations Using He's Polynomials. Computers & Mathematics with Applications, 61, (2011) 1963-1967.
[23] Kumar, J.S.D. and Rathore, S. Application of Homotopy Perturbation Transform Method for Solving Linear and Nonlinear Klein-Gordon Equations. Journal of Information and Computing Science, 7, (2012), 131-139.
[24] Daftardar-Gejji. V and Jafari H, An iterative method for solving nonlinear functional equations, Journal of Mathematical Analysis and Applications, vol. 316, no. 2, (2006) pp. 753–763.
[25] Bhalekar S., Daftardar-Gejji V. Proving Convergence of the New Iterative Method, (sub-
mitted to Comput. Math. Appl..
[26] Bhalekar S., Daftardar-Gejji V. New Iterative Method: Application to Partial Differential
Equations. Appl. Math. Comput. 203 (2008) 778-783.
[27] Daftardar-Gejji V., Bhalekar S. Solving fractional diffusion-wave equations using the New Iterative Method. Frac. Calc. Appl. Anal. 11(2) (2008) 193–202.
[28] Sachin Bhalekar and Varsha Daftardar-Gejji Solving evolution equations using a new iterative method, Numerical Methods for Partial Differential Equations (2009),1-13.
[29] Ralph E. W and Venkat R. S Computational Methods in Chemical Engineering with
Maple, 2010 Springer-Verlag Berlin Heidelberg pp85-86, DOI 10.1007/978-3-642-04311-6
[30] Abolarin O.E, Kuboye J.O, Adeyefa E.O, Ogunware B.G, New efficient numerical model for solving second, third and fourth order ordinary differential equations directly, Journal of Science. GU J Sci 33(4) (2020), 830-831.

[31] Adoghe, L. O., Ogunware, B. G, Omole, E. O. A family of symmetric implicit higher order methods for the solution of third order initial value problems in ordinary differential equations, Theoretical Mathematics & Applications, 6(3), (2016), 67-84.

[32] Hussain H, Ismai F, N. Senu I, Solving directly special fourth-order ordinary differential equations using Runge-Kutta type method, Journal of Computational and Applied Mathematics (2016), http://dx.doi.org/10.1016/j.cam.2016.04.002.

[33] Roseline B. O, Ololade F.F, Taiwo S.F. Optimization of one-step hybrid method for direct solution of fifth order ordinary differential equations of initial value problems, World Journal of Advanced Research and Reviews, 2021, 09(01), 239–249.

[34] Jena, S. R , Mohanty M. Numerical treatment for ODE (Fifth order). International Journal on emerging Technologies. 2019; 10(4): 191–196

Article DOI: 10.52589/AJMSS-52OUBAHN
DOI URL: https://doi.org/10.52589/AJMSS-52OUBAHN