Mathematical Theory and Modeling
ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online)
Vol.3, No.13, 2013

On the Construction and Comparison of an Explicit Iterative Algorithm with Nonstandard Finite Difference Schemes

Sania Qureshi[1*] Zaib-un-Nisa Memon[2] Asif Ali Shaikh[3] Muhammad Saleem Chandio[4]

1. Lecturer in Mathematics, Department of Basic Sciences and Related Studies, Mehran University of Engineering and Technology, Jamshoro.
2. Lecturer in Mathematics, Department of Basic Sciences and Related Studies, Mehran University of Engineering and Technology, Jamshoro.
3. Assistant Professor in Mathematics, Department of Basic Sciences and Related Studies, Mehran University of Engineering and Technology, Jamshoro.
4. Professor of Mathematics, Institute of Mathematics and Computer Science, University of Sindh, Jamshoro.

*Email of the corresponding author: sania.qureshi@faculty.muet.edu.pk

Abstract

An explicit iterative algorithm to solve both linear and nonlinear problems of ordinary differential equations with initial conditions is formulated with main focus given on its comparison with some non-standard finite difference schemes. Two first order linear initial value problems (IVPs) with periodic behavior are used to analyze the performance of the proposed algorithm with respect to maximum absolute error and computational effort where proposed algorithm performs better in both cases. The proposed algorithm efficiently follows the oscillatory behavior of models like Lotka-Volterra predator-prey and mass-spring system (damped case) in comparison to the nonstandard schemes. All necessary computations have been carried out through MATLAB version 8.1 (R2013a) in double precision arithmetic. Numerical results obtained by the proposed algorithm are found to be computationally reliable and practical in comparison with two nonstandard finite difference schemes discussed in literature.

Keywords: Iterative algorithm, nonstandard finite difference scheme, Initial conditions, Maximum absolute error.

1. Introduction

To solve models based on ordinary or partial differential equations is one of the major and challenging tasks being faced by researchers belonging to various fields of science and engineering. It is seldom possible to explicitly solve such models (especially nonlinear) in terms of elementary mathematical functions (Burden and Faires, 2010; Ibijola and Ogunrinde, 2010; Soomro *et al*., 2013) and this is where computer simulation and approximate methods play their dynamic role. Numerical techniques to solve Initial Value Problems (IVPs) modelled by ordinary differential equations (ODEs) help us to analyze and investigate various features of dynamical systems (Yin *et al*., 2013). A few of such dynamical and/or nonlinear systems based on a set of first order ordinary differential equations include Lorenz model (Lorenz, 1963), Lotka – Volterra predator – prey model (Lotka, 1910; Goel, 1971), Lane – Emden equation (Homer, 1870) and Van der Pol's oscillator (Van der Pol, 1927). These systems find number of applications in many scientific fields like, Meteorology, Ecology, Biomathematics, oscillatory chemical reactions, chaos theory, seismology, and astrophysics; just to mention a few, and are often considered to be model problems in order to test accuracy and efficacy of various newly developed algorithms for solving IVPs as commented in (Soomro *et al*., 2013) .

In this paper, an attempt is made to improve an explicit numerical algorithm to solve a single ODE or a system of ODEs subject to initial conditions as shown below:

$$\left.\dot{y} = f\left(t, y\right);\ y\left(t_0\right) = \alpha\right\} \quad (1.1)$$

and

$$\left.\begin{aligned}&\dot{y}_1 = f_1\left(t, y_1, y_2, \cdots, y_n\right)\\&\dot{y}_2 = f_2\left(t, y_1, y_2, \cdots, y_n\right)\\&\vdots\\&\dot{y}_n = f_n\left(t, y_1, y_2, \cdots, y_n\right)\\&\text{subject to}\\&y_1\left(t_0\right) = \alpha_1, y_2\left(t_0\right) = \alpha_2, \cdots, y_n\left(t_0\right) = \alpha_n\end{aligned}\right\} \quad (1.2)$$

These models are said to be nonlinear if nonlinearity occurs in $y(t)$. There is no existence of a unique algorithm to deal with every type of differential equation; therefore, these ODEs have been divided into classes with respect to their order, type, and linearity (Zill, 2011). Many scholars have resorted to either develop new iterative algorithms or improve the efficiency of existing ones in terms of their stability, convergence and number of function evaluations. (Fatunla, 1976; Ibijola, 1997; Wazwaz, 2000; Ramos, 2007; Yang and X, 2008; Sunday and Odekunle, 2012; Nik and Soleymani, 2013) are among many of those who have introduced nonstandard finite difference schemes to solve first order IVPs and (Wambecq, 1976; Chandio and Memon, 2010; Rabiei and Ismail, 2011; Anuar *et al*., 2011; Rabiei *et al*., 2012) are among those who have improved the efficiency of existing standard algorithms.

The purpose of the coming section is to construct an improved iterative algorithm to solve problems of type (1.1) and (1.2). Various existing methods of past could not beat the introduced algorithm for one or the other reason as shown by number of examples chosen for comparison. Limitations of the algorithm are also discussed.

2. Materials and Methods

2.1. Problem Description

Consider a well-posed[1] initial value problem

$$\left.\begin{aligned}&\frac{dy(t)}{dt} = f\left(t, y(t)\right)\\&y(t_0) = y_0,\ t \in [a,b]\end{aligned}\right\} \quad (2.1)$$

Approximation y_i to the theoretical solution $y(t_i)$, in a discrete fashion, will be produced at values called nodes, in a closed interval $[a,b]\cdot$ The nodes are equally spaced throughout the interval $[a,b]\cdot$ Also $t_{i+1} = t_0 + (i+1)\Delta t$ for each $i = 0,1,2,\cdots,n$, where $\Delta t = t_{i+1} - t_i = (b-a)/n$ is called the fixed step size.

2.2 Construction of an Explicit Iterative Algorithm

Consider a continuously differentiable function $y = f(t)$ shown in fig. 1. Suppose the red line PL_1 be the tangent to the curve $y = f(t)$ at $P(t_0, y_0)$ and the green line QL_2 be the line through $Q\left(t_1, y_1^{(1)}\right)$ having the slope

$f\left(t_1, y_1^{(1)}\right)$, where $y_1^{(1)} = y_0 + \Delta t\, f(t_0, y_0)$.

[1] One and only one solution exists to the problem.

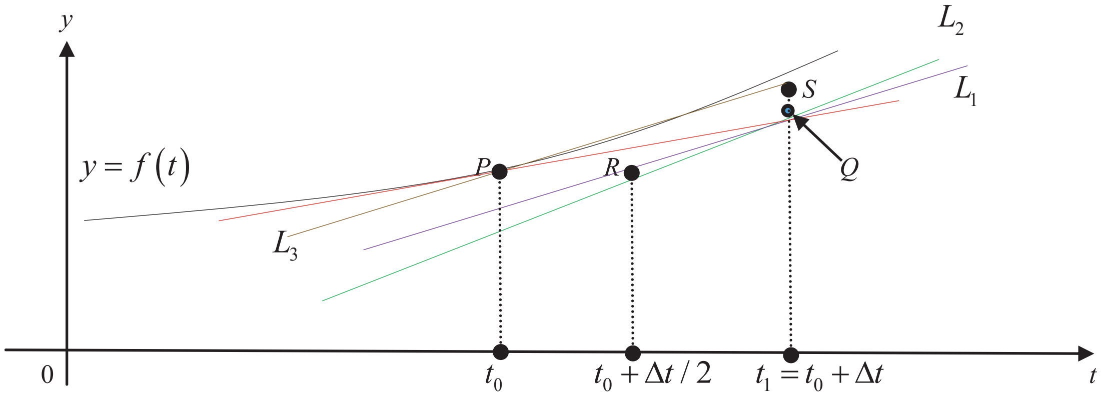

Figure 1. Graphical representation of the proposed iterative algorithm.

Now blue line QL_3 is the line passing through $Q\left(t_1, y_1^{(1)}\right)$ with slope $\left(f\left(t_0, y_0\right)+f\left(t_1, y_1^{(1)}\right)\right)\big/2$ that is average of two slopes $f\left(t_0, y_0\right)$ and $f\left(t_1, y_1^{(1)}\right)$.

Coordinates of the point R with abscissa $t_0+\Delta t/2$ and slope $\left(f\left(t_0, y_0\right)+f\left(t_1, y_1^{(1)}\right)\right)\big/2$ will be

$$R\left(t_0+\frac{\Delta t}{2}, y_0+\frac{\Delta t}{2}\left(\frac{f\left(t_0, y_0\right)+f\left(t_1, y_1^{(1)}\right)}{2}\right)\right)$$

Equation of the line (brown) PS passing through $P\left(t_0, y_0\right)$ with slope at R will be

$$y-y_0=\left(t-t_0\right) f\left(t_0+\frac{\Delta t}{2}, y_0+\frac{\Delta t}{2}\left(\frac{f\left(t_0, y_0\right)+f\left(t_1, y_1^{(1)}\right)}{2}\right)\right)$$

At $t=t_1$, we obtain

$$y_1=y_0+\left(t_1-t_0\right) f\left(t_0+\frac{\Delta t}{2}, y_0+\frac{\Delta t}{2}\left(\frac{f\left(t_0, y_0\right)+f\left(t_1, y_1^{(1)}\right)}{2}\right)\right)$$

$$=y_0+\Delta t\, f\left(t_0+\frac{\Delta t}{2}, y_0+\frac{\Delta t}{2}\left(\frac{f\left(t_0, y_0\right)+f\left(t_1, y_1^{(1)}\right)}{2}\right)\right)$$

Finally, this arrangement results in a point S that will better approximate the exact point lying on the curve $y=f(t)$.

Continuing in the same way, we will get the following proposed iterative algorithm:

$$y_{i+1}=y_i+\Delta t\, f\left(t_i+\frac{\Delta t}{2}, y_i+\frac{\Delta t}{2}\left(\frac{f\left(t_i, y_i\right)+f\left(t_{i+1}, y_i+\Delta t\, f\left(t_i, y_i\right)\right)}{2}\right)\right) \tag{2.2}$$

for $i = 0,1,2,\cdots,n$.

3. Numerical Examples and Discussion

In this section, we proceed to implement the proposed algorithm (2.2) on variety of problems of the type (1.1) and (1.2) taken from the literature. All the computations and graphical displays have been carried out using MATLAB version 8.1 (R2013a) in double precision arithmetic. Numerical results of proposed algorithm have been compared with respect to exact solution, absolute errors, CPU time, step size taken and numerical solutions obtained by existing iterative methods.

Though, linear differential equations can be solved using analytical methods but some of them consume large amount of computer time and memory. For example, the following IVP:

Problem: 01

$$\left.\begin{aligned} \dot{y} &= \sin 5t - 0.4y \\ y(0) &= 5 \end{aligned}\right\}$$

possesses an exact solution in explicit form

$$y(t) = \frac{1}{629}\left[3270e^{-2t/5} - 125\cos 5t + 10\sin 5t\right]$$

But it takes the CPU time of about 7.5432 seconds or if solved by hand involves complicated integration. On the other hand, the CPU time elapsed is approximately 0.001611 seconds when solved using the proposed algorithm showing advantage of implementing numerical algorithms. Further, integration steps (IS) taken by Euler and proposed algorithm are 40 and 20 respectively; even though, the former could not beat the latter as depicted in figure 2.

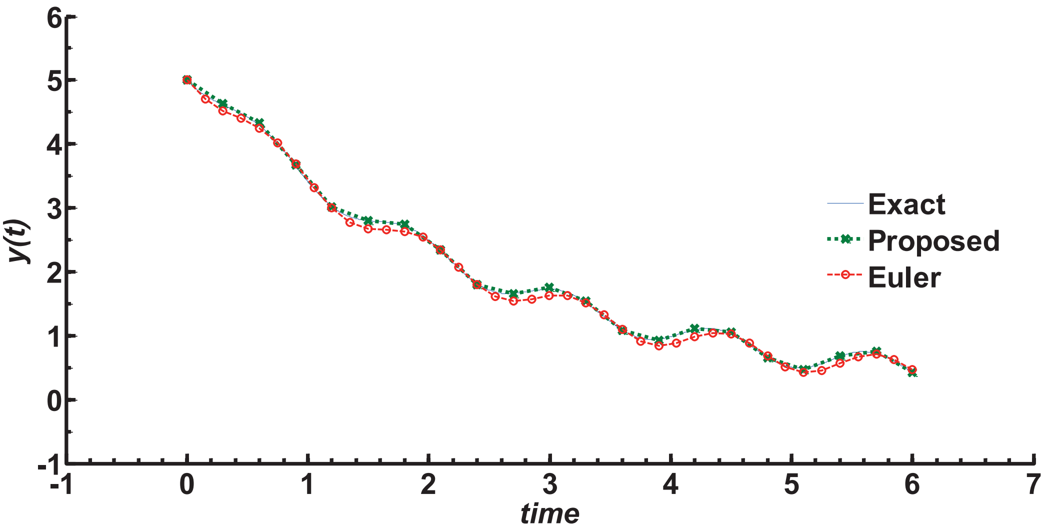

Figure 2. Comparison of Euler (IS = 40) and proposed algorithm (IS = 20) with respect to exact solution on the interval [0 6].

Another initial value problem of considerable attention is the one given below:

Problem: 02

$$\left.\begin{aligned} y' &= y\cos(t) \\ y(0) &= 1 \end{aligned}\right\}$$

Being a linear first order IVP, its analytical solution, $y(t) = e^{\sin(t)}$, is a periodic function with period 2π in t. For the purpose of comparison, a nonstandard *second order* and *A – stable* method by Ramos (2007) is selected, with the iterative algorithm given as:

$$y_{i+1} = y_i + \frac{2\Delta t\left[f\left(t_i, y_i\right)\right]^2}{2f\left(t_i, y_i\right) - \Delta t f'\left(t_i, y_i\right)}$$

where $f'\left(t_i, y_i\right) = f_t + f_y f$ at $\left(t_i, y_i\right)$.

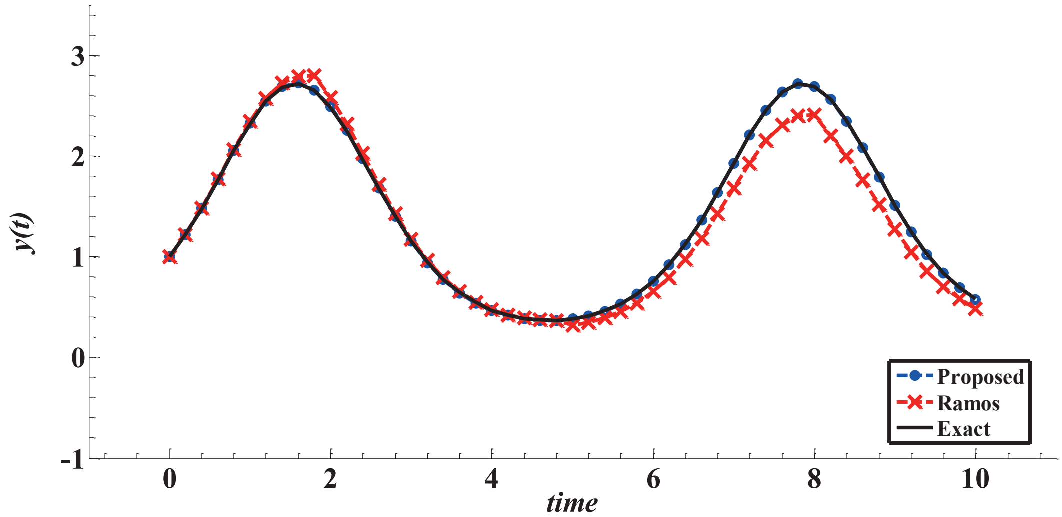

Figure 3. Comparison of proposed method with a method of Ramos with respect to exact solution for step size 0.2.

It is observed from fig. 3 that the method of Ramos shows notable fluctuations at ridges of the solution curve whereas proposed algorithm follows the periodic behavior of the solution curve even at considerably large step size $\left(\Delta t = 0.2\right)$. It should also be mentioned that Ramos method gives second order accuracy and maintains A – *stability* specifically for singular, non-singular and singularly-perturbed problems. It must be admitted at this stage that the methods like Heun's and Ralston's beat the proposed algorithm when it comes to periodicity. Though, the proposed algorithm does not require very small step sizes in order that the error does not propagate dramatically, for certain problems it will be necessary to take small step sizes and this occurs with problems having solutions of oscillatory behavior. In the regions where curvature is considerably larger, the proposed algorithm needs smaller step sizes to follow the solution as shown by figure 4 for the IVP given in problem 02. Nevertheless, there are better options to solve such problems. For further details, see (Fang, 2009) and (Yang & X., 2008).

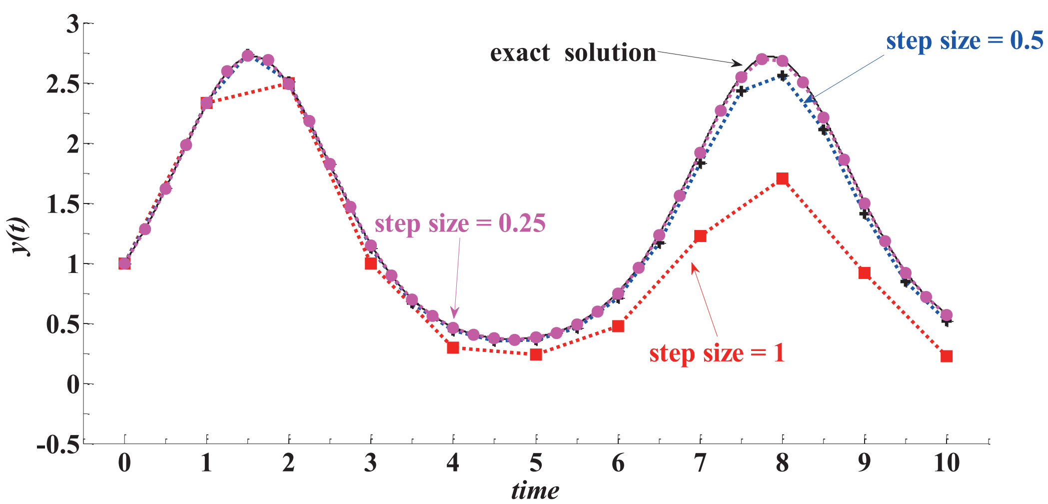

Figure 4. Proposed method follows the oscillatory behavior with reducing step size.

Computation of errors and CPU time of different algorithms with various integration steps are given in table 1. It can be observed that maximum absolute error produced by Ramos method is considerably higher than the one produced by proposed algorithm with the same integration steps and so is the case with the CPU time noted for both the methods.

Table 1. Comparison of Proposed Algorithm with Ramo's Method for $y' = y\cos(t);\ y(0) = 1$ with respect to errors and Computer time.

Method	Integration Steps	Error at $(t = 10)$	Maximum Absolute Error	CPU Time (seconds)
Ramos	25	0.141939963560321	0.872205397973849	0.002321
	50	0.096591768133376	0.366086264684939	0.002841
	100	0.045508916402834	0.239923661433473	0.004234
	200	0.003532000426836	0.015513069156405	0.007780
	400	0.020619199968314	0.100016509934113	0.018664
Proposed Algorithm	25	0.035530962934161	0.059822450713906	0.000709
	50	0.005305961565138	0.008011144164723	0.001571
	100	0.000849565979959	0.002314692012949	0.005447
	200	0.000152269593188	6.192312521497989e-04	0.010940
	400	0.000030552080064	1.599858792880049e-04	0.012100

Another feature depicted by figure 5 is continuous oscillations in error plot of Ramos method with increasing integration steps whereas error plot of proposed algorithm does not exhibit such behavior.

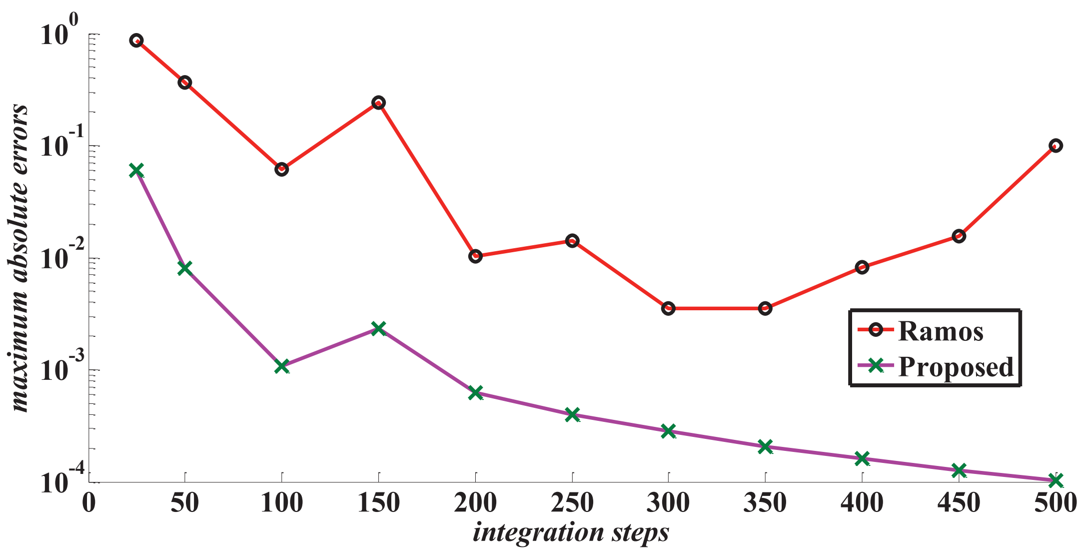

Figure 5. Error plots obtained by Ramos and Proposed Algorithm

The dynamical equations of a rationalized biological model of two challenging populations (shown by figure 6); called Lotka – Volterra predator – prey system, are given by two coupled nonlinear first order ordinary differential equations:

Problem: 03

$$\left.\begin{aligned}\dot{y}_1 &= \alpha y_1(t) - \beta y_1(t) y_2(t) \\ \dot{y}_2 &= -\gamma y_2(t) + \delta y_1(t) y_2(t)\end{aligned}\right\}$$

where $y_1(t)$ and $y_2(t)$ are number of prey (for example, rabbit) and predators (for example, fox) respectively at any given time *t*, $\dot{y}_1$ and $\dot{y}_2$ represent growth rates of the two species respectively over time, and α, β, γ, and δ are factors concerning the interaction of the two species.

Figure 6. Predator – Prey interaction, *source: www.personal.psu.ed*

The system cannot exactly be solved except for two equilibria at $(0,0)$ and $(\gamma/\delta, \alpha/\beta)$ as detailed in Zill (2009). Nonetheless, the system is capable of being analyzed through numerical algorithms. Figure 7 displays numerical solution and phase portrait of the system obtained by Euler's, Wambecq's[2] and proposed algorithm using step size of 0.1 with initial conditions $y_1(0)=2$ and $y_2(0)=1$ taking the parameters' values $\alpha=1.2,\ \beta=0.6,\ \gamma=0.8,\ \delta=0.3$ from Chapra (2012).

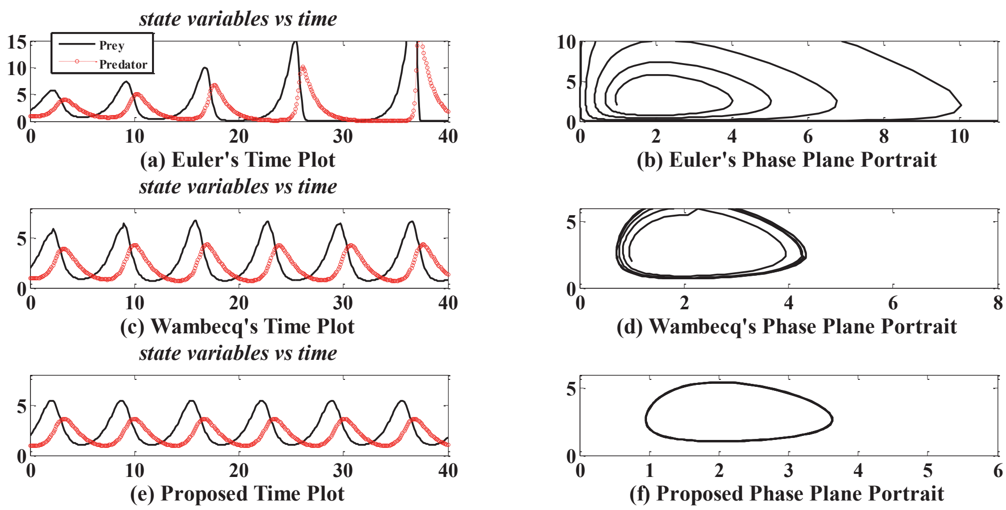

Figure 7. Numerical solution of predator – prey system using Euler's, Wambecq's and Proposed algorithm with 400 integration steps.

Although, same step size has been used for all three algorithms but fig. 7a shows oscillations with increasing amplitudes and the same is apparent by the phase plane portrait (fig. 7b). This results in using much smaller step sizes required by the Euler's method. Likewise, Wambecq's method have sharp corners at peaks of the oscillations shown in fig. 7c and the same is confirmed by the phase portrait in fig. 7d. On the other hand,

[2] $y_{i+1} = y_i + \Delta t\left[k_1^2/(2k_1 - k_2)\right]$; where $k_1 = f(t_i, y_i)$, $k_2 = f(t_i + 0.5\Delta t, y_i + 0.5\Delta t k_1)$

proposed algorithm produced better results with the same step size. Over time, a cyclical behavior develops in fig. 7e; as expected. Also, the phase portrait (fig. 7f) reveals repetition of the process by a closed trajectory.

Finally, second order linear (autonomous) ordinary differential equation called harmonic oscillator (mass-spring system) has been investigated through proposed algorithm and Wambecq's method against its solution obtained by ode45 solver offered by MATLAB keeping relative error tolerance as small as 1e-06. For the sake of simplicity, we consider the case of damped harmonic oscillator $(b>0)$ with mass of the object equals 1 (Blanchard *et al.*, 2012):

Problem: 04

$$\left.\begin{array}{l}\ddot{y}(t)+b\dot{y}(t)+ky(t)=0\\ \text{subject to}\\ y(0)=\alpha,\dot{y}(0)=\beta\end{array}\right\}$$

where $y(t)$ and $\dot{y}(t)$ are position and velocity of the spring respectively, $(k>0)$ is called spring constant and b is known as damping constant.

The system has been solved numerically through three iterative algorithms with same step size as shown in figure 8 choosing damping constant relatively small $(b=0.17)$ and spring constant $k=1$ with initial conditions: $y(0)=3.14, \dot{y}(0)=0$. Solution to the system, as expected, goes up and down with decreasing amplitude and slowly comes to rest as time goes on; and proposed algorithm favorably agrees with that of ode45 but Wambecq's shows amplitudes less than those obtained by rest of the two iterative algorithms; however, it also attains resting position with time. This type of behavior is confirmed by the phase plane portrait shown in figure 9 where solutions are spiraling in to the point where mass is at rest and there is no velocity, called equilibrium point. It can be observed from figure 9 that Wambecq came across various disruptions while reaching to equilibrium point.

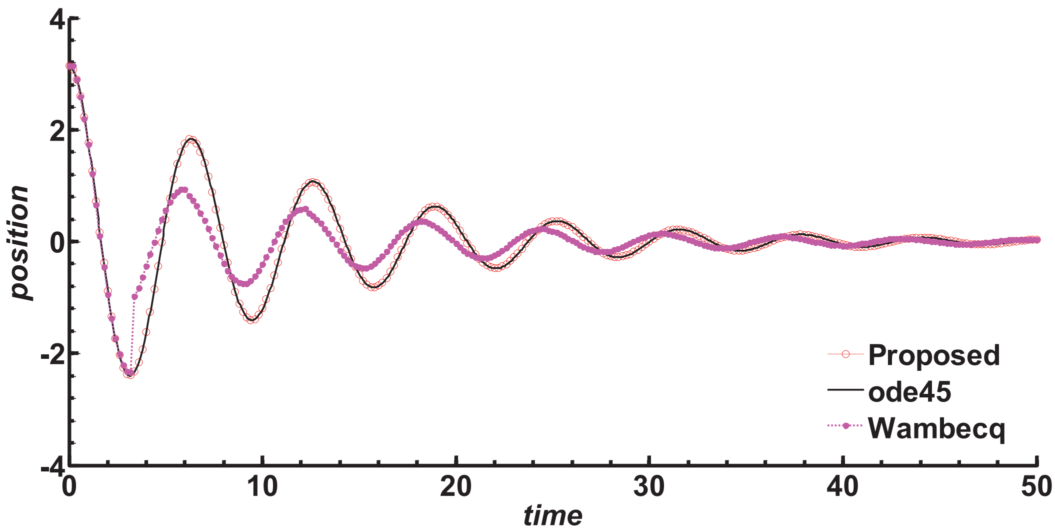

Figure 8. Behavior of solution of damped harmonic oscillator investigated by three different iterative algorithms.

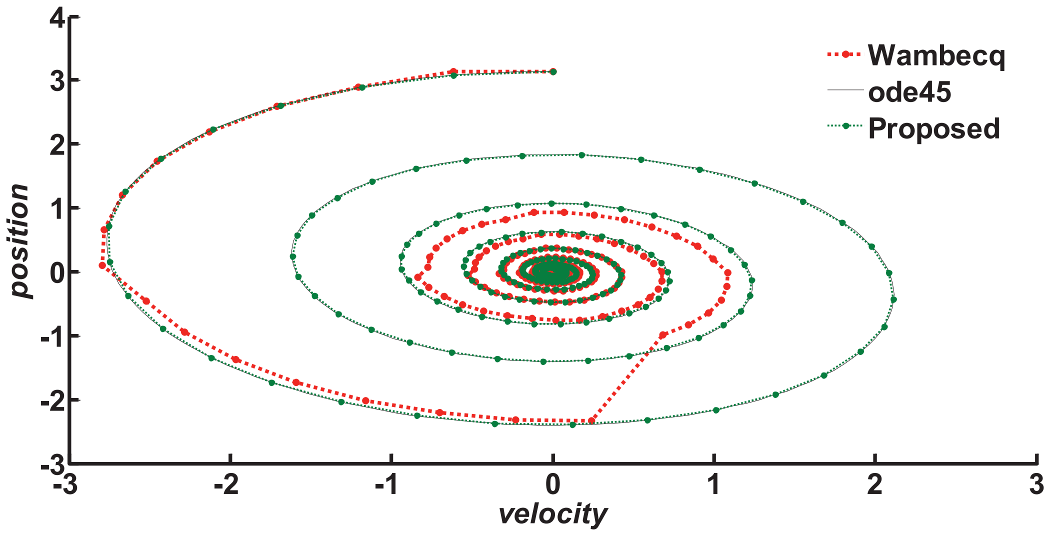

Figure 9. Phase plane portrait of damped harmonic oscillator generated by three different iterative algorithms.

4. Conclusion

In this work, formulation and a comparative study of the proposed algorithm is carried out with two nonstandard finite difference schemes. Numerical results obtained support the efficiency of the proposed algorithm in comparison to nonstandard schemes even at larger step sizes. The algorithm is also found to have lower computational cost with greater accuracy in results both for linear and nonlinear problems considered.

5. Future Work

In future, much of work would be based upon standard error analysis of the algorithm proposed. Convergence of the proposed algorithm would be discussed in detail. Its stability region would also be drawn and a comparison with standard and few more non-standard methods would be under consideration.

6. Acknowledgement

The authors would like to express their sincere thanks to anonymous referees of the journal for their thoughtful reading of the work presented.

References:

Anuar, K. H., Othman, K. I., Ishak, F., Ibrahim, Z. B., & Majid, Z. (2011). Development Partitioning Intervalwise Block Method for Solving Ordinary Differential Equations. *World Academy of Science, Engineering and Technology, 58*, 243-245.

Blanchard, P., Devaney, R. L., & Hall, G. R. (2012). *Differential Equations* (4th ed.). Boston: Richard Stratton. Retrieved November 5, 2012

Burden, R. L., & Faires, J. D. (2010). *Numerical Analysis: Theoery and Applications* (9 ed.). India, India: Cengage Learning. Retrieved August 8, 2013, from www.cengage.co.in

Chandio, M. S., & Memon, A. G. (2010). Improving the Efficiency of Heun's Method. *Sindh University Research Journal (Science Series), 42*(2), 85-88.

Chapra, S. C. (2012). *Applied Numerical Methods with Matlab for Engineeris and Scientists* (3rd ed.). India: McGraw Hill.

Fang, Y. S. (2009). A robust trigonometrically fitted embedded pair for perturbed oscillator. *Journal of Computational and Applied Mathematics, 225*(2), 347-355.

Fatunla, S. O. (1976). A New Algorithm for Numerical Solution of Ordinary Differential Equations. *Computations and Mathematics with Applications, 2*, 247-253.

Fu-Kang Yin, W.-Y. H.-Q.-Q. (2013). Legendre Wavelets-Picard Iteration Method for Solution of Nonlinear Initial Value Problems. *International Jouurnal of Applied Physics and Mathematics, 3*(2), 127-131. Retrieved October 22, 2013

Goel, N. S. (1971). *On the Volterra and other Nonlinear Models of Interacting Populations.* Academic Press Inc.

Homer, L. J. (1870). On the Theoretical temperature of the Sun under the Hypothesis of a Gaseous Mass Maintaining its Volume by its Internal Heat and Depending on the Laws of Gases Known to Terrestrial Experiment. *The American Journal of Science and Arts, 50*(2), 57-74.

Ibijola, E. A. (1997). New Scheme for Numerical Integration of Special Initial Value Problems in Ordinary Differential Equations. *Ph.D. Thesis Submitted to the Department of Mathematics*. Nigeria: University of Benin.

Ibijola, E. A., & Ogunrinde, R. B. (2010). On a New Numerical Scheme for the Solution of IVPs in ODEs. *Australian Journal of Basic and Applied Sciences, 4*(10), 5277-5282.

Lorenz, E. N. (1963). Deterministic non-periodic flow. *Journal of the Atmospheric Sciences, 20*(2), 130-141.

Lotka, J. A. (1910). Contribution to the Theory of Periodic Reaction. *Journal of Physics and Chemistry, 14*(3), 271-274.

Nik, H. S., & Soleymani, F. (2013). A Taylor-type method for solving nonlinear ordinary differential equations. *Alexandria Engineering Journal, 52*(3), 543-550.

Rabiei, F., & Ismail, F. (2011). New Improved Runge-Kutta method with reducing number of functions evaluations. *ASME Press*. doi:10.1115/1.859797

Rabiei, F., Ismail, F., Norazak, S., & Seddighi, S. (2012). Accelerated Runge-Kutta Nystrom for Solving Autonomous Second-Order Ordinary Differential Equations. *World Applied Sciences Journal, 17*(12), 1070-1549.

Ramos, H. (2007). A non-standard explicit integration scheme for initial-value problems. *Applied Mathematics and Computation, 189*(1), 710-718.

Soomro, A. S., Tularam, G. A., & Shaikh, M. M. (2013). A Comparison of Numerical Methods for Solving the Unforced Van Der Pol's Equation. *Mathematical Theory and Modeling, 3*(2), 66-78.

Sunday, J., & Odekunle, M. R. (2012). A New Numerical Integrator for the Solution of Initial Value Problems in Ordinary Differential Equations. *The Pacific Journal of Science and Technology, 13*(1), 221-227.

Van der Pol, B. (1927). On relaxations-oscillations. *The London, Edinburgh and Dublin Phil. Mag. & J. of Sci., 2*(7), 978-992.

Wambecq, A. (1976). Nonlinear methods in solving ordinary differential equations. *Journal of Computational and Applied Mathematics, 2*(1), 27-33. Retrieved December 10, 2012

Wazwaz, A. M. (2000). A New Algorithm for Calculating Adomian Polynomials for Nonlinear Operationn. *Applied Mathematics and Computation, 3*, 1375-1395.

Yang, H., & X., W. (2008). Trigonometrically-fitted AKRN methods for perturbed oscillators. *Applied Numerical Mathematics, 58*(9), 1375-1395.

Zill, D. G. (2009). *A First Course in Differential Equations with Modeling Applications* (9th ed.). Cengage Learning.