Enhanced Knowledge in Sciences and Technology Vol. 3 No. 2 (2023) 148-156
© Universiti Tun Hussein Onn Malaysia Publisher's Office

EKST

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/ekst
e-ISSN : 2773-6385

Solving Lotka-Volterra Prey-Predator Model Using Implicit Multistep Method of Order Four

Haizad Hakimi[1], Syahirbanun Isa[2]*

[1]Merchantrade Asia Sdn Bhd (Celcopon Sdn Bhd),
Suite 1316, Level 13, Lobby 3, Block A,
Damansara Intan, 47400 Petaling Jaya, Selangor, MALAYSIA

[2]Department of Mathematics and Statistics,
Faculty of Applied Sciences and Technology,
Universiti Tun Hussein Onn Malaysia (Pagoh Campus),
84600 Pagoh, Muar, Johor, MALAYSIA

*Corresponding Author Designation

DOI: https://doi.org/10.30880/ekst.2023.03.02.017
Received 15 January 2023; Accepted 04 February 2023; Available online 30 November 2023

Abstract: This paper study the applicability, and the efficiency of the implicit multistep method of order four, resulting in the system of first order nonlinear ordinary differential equations (ODEs). A system of first order nonlinear ODEs have been considered with initial conditions which are Lotka-Volterra prey-predator model. The model has been solved by implicit multistep method of order four using different step size, *h* with the help of MATLAB. The absolute error analyzed are also carried out explicitly in the framework. The numerical results of the model have been calculated and compared with Runge-Kutta method of order four and MATLAB ODE solver (ode45). Moreover, for the Lotka-Volterra prey-predator model, a comparison of the computation is compared and is worked out to illustrate the general advantage of proposed method.

Keywords: Lotka-Volterra Prey-Predator Model, Implicit Multistep Method, Ordinary Differential Equation

1. Introduction

A system of differential equations is a finite set of differential equations such a system can be either linear or non-linear. Also, such a system can be either a system of ordinary differential equations (ODEs) or a system of partial differential equations (PDEs). The prey-predator equation is another name for the set of first order nonlinear differential equations known as the Lotka-Volterra equations. To describe the dynamics of ecological systems where one species acts as a predator and the other as a prey, the Lotka-Volterra model is frequently utilized.

The Lotka–Volterra model is numerically solved using Runge-Kutta-Fehlberg method (RKF) and Laplace Adomian decomposition method (LADM) in [1]. It is found out that RKF have high rate of

**Corresponding author: syahir@uthm.edu.my*

accuracy in comparison to LADM. Authors in [2] demonstrated that, over time the Lotka-Volterra model, either the competitive exclusion maintains, resulting in the extinction of one species, or the two species create a coexistence equilibrium. [3] outlined two accurate, model-based techniques for solving the fractional-order Lotka-Volterra population model system based on the Morgan-Voyce (MV) functions.

The following general Lotka–Volterra model is used for *n* species, and the equations might describe prey-predator or competition cases.

$$\frac{dN_i}{dt} = N_i(\alpha_i - \sum_{j=1}^{n} \beta_{ij} N_j), \ i = 1, 2, \dots, n; \ i \neq j. \qquad \text{Eq. 1}$$

where, $\frac{dN_i}{dt}$ indicates the two populations observed growth rates over time, t and α, β, γ and δ are parameters describing the interaction of the two species.

Based on [4], system of ODEs can be solved analytically and numerically. Due to some limitations of analytical method especially for solving difficult and complicated problems, numerical methods have been chosen to overcome this limit. Numerical methods consist of one-step and multistep methods. The accurate one-step methods for example Runge-Kutta method of order four required more function evaluation. Hence, it is less efficient. The multistep methods with the same order can give the accurate solution with less function evaluation. An implicit method will be used because it is more accurate. [5] concentrated their research on implicit multistep numerical technique for real-time modelling of stiff systems which resulting that using an implicit multistep approach cuts down on the amount of time it takes to calculate. First-order delay differential equations were solved using hybrid block Adams Moulton techniques (BHAMM) for step number, along with two and three off-grid locations, without applying the interpolation condition in [6].

A multistep, *m*-step approach to solving the initial-value problem

$$y' = f(t, y), \ \alpha \le t \le b, \ y(a) = \alpha, \qquad \text{Eq. 2}$$

a difference equation that can be used to get an approximate value y_{i+1} at the mesh point t_{i+1} represented by the following equation, where *m* is an integer greater than 1:

$$\begin{aligned} y_{i+1} &= a_{n-1} y_i + a_{n-2} y_{i-1} + \cdots + a_0 y_{i+1-n} \\ &\quad + h[b_n f(t_{i+1}, y_{i+1}) + b_{n-1} f(t_i, y_i) + \cdots + b_0 f(t_{i+1-n}, y_{i+1-n})], \end{aligned} \qquad \text{Eq. 3}$$

for $i = n-1, n, \dots, N-1$, where $h = \frac{h(b-a)}{N}$, the $a_0, a_1, \dots, a_{n-1}$ and $b_0, b_1, \dots, b_n$ are constants, and the starting values

$$y_0 = \alpha, y_1 = \alpha_1, y_2 = \alpha_2, \dots, y_{n-1} = \alpha_{n-1} \qquad \text{Eq. 4}$$

are specified. When $b_m = 0$, the method of Eq. 3 is called explicit and when $b_m \neq 0$ the method is called implicit. An implicit multistep method of order four method will be used in this project.

The objectives of this study were to examine Lotka-Volterra prey predator model that appropriate for numerical analysis. Next, to solve Lotka-Volterra prey-predator model using implicit multistep method of order four and compare the solution of Lotka-Volterra prey-predator model using implicit multistep method of order four with Runge-Kutta method of order four and MATLAB ODE (ode45).

2. Materials and Methods

2.1 Lotka-Volterra prey-predator model

Referring to general Lotka-Volterra model in Eq. 1, the case of two species that will be using for this study, which is defined as follows (Source [7])

$$\frac{dN_1}{dt} = N_1(\alpha - \beta N_2)$$
$$\frac{dN_2}{dt} = N_2(\delta N_1 - \gamma) \qquad \text{Eq. 5}$$

with the initial conditions $N_1(0) > 0$ and $N_2(0) > 0$. Prey species have a population size of N_1, whereas predator species have a population size of N_2. The per capita decrease in prey per predator is shown in α, while the per capita increase in predator per prey is shown in γ. Prey and predator species have death rates of β and δ, respectively. The parameters $\alpha, \beta, \delta, \gamma$ are all positive. $\frac{dN_1}{dt}$ and represent $\frac{dN_2}{dt}$ the growth rates of the prey and predator species over time, *t*, respectively.

2.2 Implicit Multistep Method of Order Four

Multistep approaches used the estimation at several previous mesh points to ascertain the estimation at the following point. By the Eq. 3, when $m = 2$, the starting values Eq. 4 are

$$w_0 = \alpha,\ w_1 = \alpha_1,\ w_2 = \alpha_2, \qquad \text{Eq. 6}$$

with

$$w_{i+1} = w_i + h\left[\frac{9}{24} f(t_{i+1}, w_{i+1}) + \frac{19}{24} f(t_i, w_i) - \frac{5}{24} f(t_{i-1}, w_{i-1}) + \frac{1}{24} f(t_{i-2}, w_{i-2})\right]. \qquad \text{Eq. 7}$$

For each $i = 2, 3, \ldots, N-1$, is the implicit multistep method of order four, which is an implicit three-step procedure. In the above equation, the initial values must be defined, usually by assuming $w_0 = \alpha$ and calculating the starting values using the any one-step methods. The values in function $f(t, w)$ of the above equation are calculated based on the model in Eq. 5.

3. Results and Discussion

This section demonstrates the solutions of the system of first order nonlinear ODEs using the proposed method. The Lotka-Volterra prey predator model includes numerous cases, each one comes with a unique case where the value for a specific parameter is different.

3.1 Test Problem

This study considered only one case of two species Lotka-Volterra prey-predator model. By referring to the model in Eq. 5 with the initial conditions of $N_1(0) = 4$, $N_2(0) = 9$ and the time interval, $0 \le t \le 15$. The parameters in this case are $\alpha = 0.1$, $\beta = 0.0014$, $\delta = 0.08$, $\gamma = 0.0012$ [8].

The problem is tested with two different step sizes, h to determine its accuracy. The step size that we will be considering are when $h = 1$ due to the suitability with time interval, t and will be compared with a smaller step size, $h = 0.1$. The numerical solutions then compared with Runge-Kutta method of order four and MATLAB ODE solver (ode45).

3.2 Numerical Result

This section discussed the results for the Lotka-Volterra prey-predator. The model is solved numerically by using implicit multistep method of order four in which will be compared with Runge-Kutta method of order four and MATLAB ODE solver (ode45). All the calculation are performed using MATLAB software.

Table 1 and Table 2 show the numerical results and Table 3 shows the absolute error of the Lotka-Volterra prey-predator model using the proposed method which compared to Runge-Kutta method of order four and MATLAB ODE solver (ode45) for step size, $h = 0.1$.

Table 1: Numerical results of Lotka-Volterra prey predator model using MATLAB ODE solver (ode45) at $h = 0.1$

t	Matlab ODE solver (ode45)	
	N1	N2
0	4.000000000000	9.000000000000
1	4.355229552958	12.555900623896
2	4.712661174290	18.024508997048
3	5.050392184670	26.610324316665
4	5.330192556619	40.280871902782
5	5.489826045430	62.089349336158
6	5.440087091765	96.190336033097
7	5.082754369499	146.697496885459
8	4.370549062258	214.351254270641
9	3.389576928911	292.358203405037
10	2.357849082424	367.302605267776
11	1.490888425561	427.249404224656
12	0.878177689395	468.371766373612
13	0.494118837149	493.566885596673
14	0.270673309217	507.856073405226
15	0.146063225774	515.517256259041

Table 2: Numerical results of Lotka-Volterra prey predator model using implicit multistep method of order four and Runge-Kutta method of order four at $h = 0.1$

t	Implicit multistep method of order 4		Runge-Kutta of order 4	
	N1	N2	N1	N2
0	4.000000000000	9.000000000000	4.000000000000	9.000000000000
1	4.355235051243	12.555649553729	4.355235059104	12.555649195405
2	4.712662386733	18.024454386442	4.712662415852	18.024453092060
3	5.050399790528	26.609972649007	5.050399858094	26.609969734022
4	5.330202571060	40.280605331115	5.330202693103	40.280600324379
5	5.489867261262	62.087258166895	5.489867421240	62.087252386325
6	5.440183812449	96.185554570723	5.440183901724	96.185553640119
7	5.082884447443	146.690626300172	5.082884297429	146.690637063475
8	4.370829517966	214.337793620024	4.370829251947	214.337806532510
9	3.389811432429	292.349224883110	3.389811583313	292.349214956297

10	2.358040465916	367.298256897160	2.358041018923	367.298232586068
11	1.491056795316	427.246353362950	1.491057139557	427.246345732528
12	0.878483254299	468.359643264537	0.878483250659	468.359653563711
13	0.494371679431	493.559445834347	0.494371565065	493.559459331503
14	0.270778418006	507.860459954810	0.270778342264	507.860469535929
15	0.146108808574	515.525647669339	0.146108788920	515.525653522268

Table 3: Absolute error at $h = 0.1$

t	Implicit multistep method of order 4		Runge-Kutta of order 4	
	N1	N2	N1	N2
0	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
1	5.498285e-06	2.510702e-04	5.506146e-06	2.514285e-04
2	1.212443e-06	5.461061e-05	1.241562e-06	5.590499e-05
3	7.605858e-06	3.516677e-04	7.673424e-06	3.545826e-04
4	1.001444e-05	2.665717e-04	1.013648e-05	2.715784e-04
5	4.121583e-05	2.091169e-03	4.137581e-05	2.096950e-03
6	9.672068e-05	4.781462e-03	9.680996e-05	4.782393e-03
7	1.300779e-04	6.870585e-03	1.299279e-04	6.859822e-03
8	2.804557e-04	1.346065e-02	2.801897e-04	1.344774e-02
9	2.345035e-04	8.978522e-03	2.346544e-04	8.988449e-03
10	1.913835e-04	4.348371e-03	1.919365e-04	4.372682e-03
11	1.683698e-04	3.050862e-03	1.687140e-04	3.058492e-03
12	3.055649e-04	1.212311e-02	3.055613e-04	1.211281e-02
13	2.528423e-04	7.439762e-03	2.527279e-04	7.426265e-03
14	1.051088e-04	4.386550e-03	1.050330e-04	4.396131e-03
15	2.804557e-04	8.391410e-03	2.801897e-04	8.397263e-03

Figure 1 and Figure 2 represent the graph of numerical results and the absolute error of Lotka-Volterra prey-predator model using the proposed method which compared to Runge-Kutta method of order four and MATLAB ODE solver (ode45) for step size, $h = 0.1$.

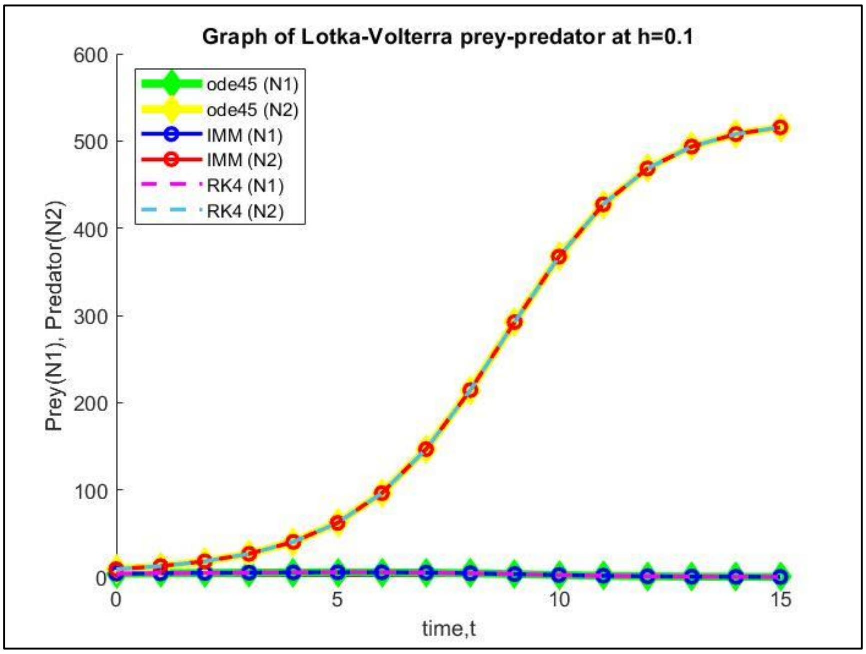

Figure 1: Graph of Lotka-Volterra prey-predator model for step size $h = 0.1$

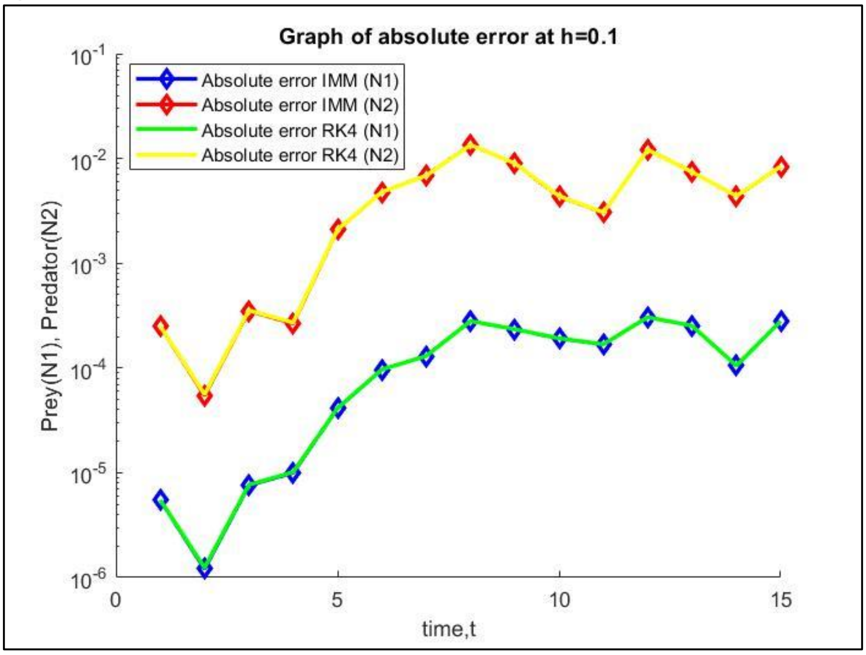

Figure 2: Graph of Absolute error for step size $h = 0.1$

Table 3 and Table 4 show the numerical results and Table 5 shows the absolute error of Lotka-Volterra prey-predator model using the proposed method which compared to Runge-Kutta method of order four and MATLAB ODE solver (ode45) for step size, $h = 1$.

Table 4: Numerical results of Lotka-Volterra prey predator model using Matlab ODE solver (ode45) at $h = 1$

t	Matlab ODE solver (ode45)	
	N1	N2
0	4.000000000000	9.000000000000
1	4.355229776948	12.555890294999
2	4.712661400917	18.024498777693
3	5.050391525657	26.610350762125
4	5.330193163149	40.280848556941
5	5.489824896778	62.089419233143
6	5.440094602381	96.189916340317
7	5.082748760910	146.697800693637
8	4.370549762065	214.351166683359
9	3.389564460698	292.358907382199
10	2.357846041633	367.302847903709
11	1.490888700074	427.249413503344
12	0.878176707665	468.371939231285
13	0.494132124755	493.565920506169
14	0.270664116881	507.856711649955
15	0.146063105902	515.517252498891

Table 5: Numerical results of Lotka-Volterra prey predator model using implicit multistep method of order four and Runge-Kutta method of order four at $h = 1$

t	Implicit multistep method of order 4		Runge-Kutta of order 4	
	N1	N2	N1	N2
0	4.000000000000	9.000000000000	4.000000000000	9.000000000000
1	4.355242939374	12.555348766033	4.355242939374	12.555348766033
2	4.712689203880	18.023402381916	4.712689203880	18.023402381916
3	5.050468951661	26.607178383083	5.050468951661	26.607178383083
4	5.330445168287	40.267674588109	5.330360358623	40.274050499578
5	5.490483632560	62.051753806123	5.490194038425	62.073355962023
6	5.441460953561	96.115192875002	5.440787853570	96.159409596116
7	5.085197157515	146.602749534306	5.083842982486	146.648930445045
8	4.374401890399	214.339150115103	4.372073544093	214.285095570400
9	3.392424202012	292.615124714805	3.391075946038	292.301522319712
10	2.352351789403	367.779695476227	2.359085311592	367.269869256001
11	1.475289148385	427.439219761252	1.491943960934	427.231877538224
12	0.866360124844	467.857144298690	0.879367552862	468.347597847525
13	0.495743801020	492.677911402956	0.495239030340	493.549704543580
14	0.278654846788	507.146251034196	0.271531268562	507.858786989459
15	0.151786283782	515.110150067667	0.146690827534	515.535426202571

Table 6: Absolute error at $h = 1$

t	Implicit multistep method of order 4		Runge-Kutta of order 4	
	N1	N2	N1	N2
0	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
1	1.316243e-05	5.415290e-04	1.316243e-05	5.415290e-04
2	2.780296e-05	1.096396e-03	2.780296e-05	1.096396e-03
3	7.742600e-05	3.172379e-03	7.742600e-05	3.172379e-03
4	2.520051e-04	1.317397e-02	1.671955e-04	6.798057e-03
5	6.587358e-04	3.766543e-02	3.691416e-04	1.606327e-02
6	1.366351e-03	7.472347e-02	6.932512e-04	3.050674e-02
7	2.448397e-03	9.505116e-02	1.094222e-03	4.887025e-02
8	3.852128e-03	1.201657e-02	1.523782e-03	6.607111e-02
9	2.859741e-03	2.562173e-01	1.511485e-03	5.738506e-02
10	5.494252e-03	4.768476e-01	1.239270e-03	3.297865e-02
11	1.559955e-02	1.898063e-01	1.055261e-03	1.753597e-02
12	1.181658e-02	5.147949e-01	1.190845e-03	2.434138e-02
13	1.611676e-03	8.880091e-01	1.106906e-03	1.621596e-02
14	7.990730e-03	7.104606e-01	8.671517e-04	2.075340e-03
15	5.723178e-03	4.071024e-01	6.277216e-04	1.817370e-02

Figure 3 and Figure 4 represent the graph of numerical results and the absolute error of Lotka-Volterra prey-predator model using the proposed method which compared to Runge-Kutta method of order four and MATLAB ODE solver (ode45) for step size, $h = 1$.

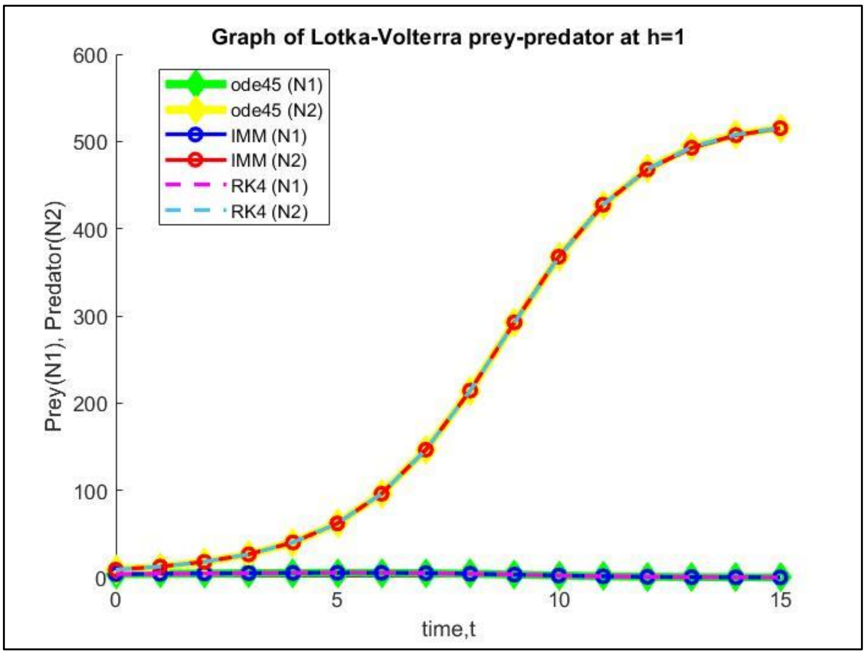

Figure 3: Graph of Lotka-Volterra prey-predator model for step size $h=1$

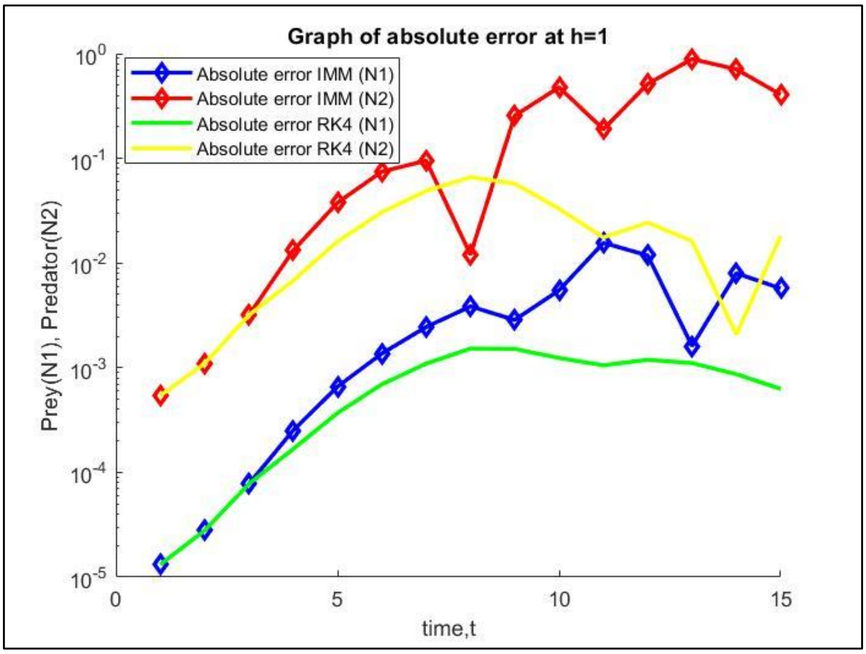

Figure 4: Graph of Absolute error for step size $h=1$

Table 1, Table 2, Table 4 and Table 5 shown that the implicit multistep method of order four is reliable to solve Lotka-Volterra prey-predator model. It can be proved by the graph in Figure 1 and Figure 3 where the solutions of implicit multistep method of order four are coincide with MATLAB ODE solver (ode45) also with Runge-Kutta method of order four. The absolute in Table 3 and Table 6 errors have calculated based on MATLAB ODE solver (ode45) results and it was found out as in Figure 2 and Figure 4 that implicit multistep method of order four and Runge-Kutta method of order four are comparable. However, implicit multistep method of order four is more efficient since Runge-Kutta method of order four need more function evaluation to obtain the result for the same step of solution.

4. Conclusion

This study emphasized on solving Lotka-Volterra prey-predator model using the implicit multistep method of order four. The results are compared with Runge-Kutta method of order four and MATLAB ODE solver (ode45). MATLAB software is used to calculate the solutions using various step sizes. The results showed that decreasing the step size improved the numerical output. It can be concluded that the implicit multistep method of order four can be used to solve nonlinear ODEs and is more efficient than Runge-Kutta method of order four.

Acknowledgement

The authors would also like to thank the Faculty of Applied Sciences and Technology, Universiti Tun Hussein Onn Malaysia for its support.

References

[1] S. Paul, S. P. Mondal, & P. Bhattacharya, "Numerical solution of Lotka Volterra prey predator model by using Runge–Kutta–Fehlberg Method and Laplace Adomian decomposition method," Alexandria Engineering Journal, *55*(1), 613–617, 2016, doi: 10.1016/j.aej.2015.12.026.

[2] S. Chen, J. Shi, Z. Shuai & Y. Wu, "Global Dynamics of a Lotka-Volterra competition patch model," Nonlinearity, 35(2), 817, 2021. Available: https://arxiv.org/pdf/2101.02265.pdf

[3] M. Izadi, Ş. Yüzbaşı, W. Adel, "Accurate and efficient matrix techniques for solving the fractional Lotka–Volterra population model," Physica A: Statistical Mechanics and its Applications, 600, 127558, 2022, doi: 10.1016/j.physa.2022.127558.

[4] R. L. Burden, J. D. Faires, & A. M. Burden, Numerical Analysis. Cengage learning, 2016

[5] Z. Lei, L. Jie, W. Menglu, & L. Mengya, "An implicit multistep numerical method for real-time simulation of stiff systems," Simulation, 97(12), 835-848, 2021, doi: 10.1177/00375497211021653.

[6] C. Chibuisi, et al., "Implementation of Two-step Hybrid Block Adams Moulton Solution Methods for First Order Delay Differential Equations," in Journal of Physics: Conference Series (Vol. 2199, No. 1, p. 012017), IOP Publishing, 2022, Available: https://iopscience.iop.org/article/10.1088/1742-6596/2199/1/012017/pdf.

[7] S. Paul, S. P. Mondal, & P. Bhattacharya, "Numerical solution of Lotka Volterra prey predator model by using Runge–Kutta–Fehlberg method and Laplace Adomian decomposition method," Alexandria Engineering Journal, 55(1), 613-617, 2016, doi: 10.1016/j.aej.2015.12.026

[8] M. Umar, Z. Sabir, & M. A. Z. Raja, "Intelligent computing for numerical treatment of nonlinear prey–predator models," Applied Soft Computing, 80, 506-524, 2019, doi: 10.1016/j.asoc.2019.04.022