International Journal on Robotics, Automation and Sciences

Numerical Integration Approach for Nonlinear Differential Equation in Growth Modelling

Hui Shan Tai and Srimazzura Basri*

***Abstract* – The nonlinear ordinary differential equation (ODE) is a common mathematical model for real-world problems. However, its analytical solution is hard to find and may not exist due to the nonlinear and complex structures. Thus, an approximate method is usually employed in mathematical modelling to obtain its solution. This study applies numerical integration techniques, namely Gaussian quadrature and Simpson's rule methods, to solve nonlinear ODE, which is a hyperbolic growth model. We first discuss the ODE model and then substitute its exact solution model into the ODE model to obtain the model's numerical solution using numerical integration approaches. Next, we aim to predict the solution of the nonlinear growth model by proposing two linear models and integrating them iteratively. We introduce a least square optimization problem and derive a set of first-order necessary conditions for estimating the model parameter optimally. A gradient descent method is employed to iterate and update the solution of the linear model. The numerical integration techniques are efficient, while the proposed method has proved to be an alternative approach to handling nonlinear ODEs, especially for a nonlinear growth model, since the optimal linear model solution satisfactorily approximates the growth model solution with a small mean square error value.**

Keywords—Nonlinear Ordinary Differential Equations, Approximate Solution, Numerical Integration Approach, Growth Modelling, Euler, Runge-Kutta.

I. Introduction

Nonlinear ordinary differential equations (ODEs) are a common mathematical model for representing real-world problems, as an ODE model represents a rate of change [1]. They are an extension of the derivatives in calculus, and the anti-derivative is called an integral [2]. ODEs have been well-studied in a wide range of applications, from engineering sciences to financial business, either as initial value problems, boundary value problems, or dynamical systems.

In a nonlinear ODE, the dependent variable and its derivatives appear nonlinearly. Hence, nonlinear ODEs have a complex structure, and closed-form solutions are generally unavailable [3]. An initial value problem, which consists of an ODE and an initial condition [4] is the common mathematical model used to represent real-world problems. Al-Mazmumy [5] has proposed a modified Adomian decomposition method to solve nonlinear initial value problems for ODEs. Koroche [6] used the fourth-order Adams predictor-corrector method to approximate the exact solution of the proposed first-order initial value problem.

There are many existing solution methods, such as Euler and Runge-Kutta fourth order (RK4), for solving nonlinear ODEs. Sharma [7] proposed using the finite difference method and Runge-Kutta (4,5) to solve the nonlinear problem. Capuano [8] developed the

*Corresponding Author email: mazzura@uthm.edu.my, ORCID: 0000-0001-6732-9749
Hui Shan Tai is with Department of Mathematics and Statistics, Universiti Tun Hussein Onn Malaysia, Pagoh Campus, Muar, Johor, Malaysia (e-mail: aw210214@student.uthm.edu.my)
Srimazzura Basri is with Department of Mathematics and Statistics, Universiti Tun Hussein Onn Malaysia, Pagoh Campus, Muar, Johor, Malaysia (e-mail: mazzura@uthm.edu.my)

International Journal on Robotics, Automation and Sciences (2025) 7, 2:29-35
https://doi.org/10.33093/ijoras.2025.7.2.4
Manuscript received:3 Apr 2025 | Revised: 15 May 2025 | Accepted: 22 May 2025 | Published: 30 Jul 2025

Published by MMU PRESS. URL: http://journals.mmupress.com/ijoras

Backward Euler method with the Newton-Raphson method to replace the Runge-Kutta methods in solving nonlinear ordinary differential equations. The numerical methods embed the definition of derivative into the nonlinear ODE and convert it to a discrete form to simplify the calculation procedure in a digital computer to give an approximate solution of a nonlinear ODE [9].

In recent years, machine learning approaches have been employed in solving differential equations. Televnoy [10] proposed embedding a modified Runge-Kutta method into the neural ODE to minimize the fading gradients issue, and the results are compared with standard machine learning methods. Viana [11] proposed to implement specific numerical integration methods, such as the Euler and Runge-Kutta methods, on prescribed recurrent neural network cells to reduce the gap between predictions and observations of systems described by ODEs. Jaradat [12] developed an adaptive time-stepping control algorithm based on the control theory approach for solving the PROTEUS-NODAL code, which is an ODE initial value problem. Similarly, Ratchagit [13] demonstrated the applications of machine learning techniques, such as artificial neural network (ANN), long short-term memory network (LSTM), and convolutional neural network (CNN) to predict the concentration of PM2.5 particles in Chiang Mai. This study shares a common goal of minimizing prediction error through optimization-based iterative methods, even though the research field differs.

However, direct integration methods based on quadrature rules are generally less popular in solving nonlinear ODEs due to the difficulty and time-consuming in integrating complex nonlinear functions explicitly, even when they are well studied and applied in specific classes of problems, such as collocation-based solvers. Furthermore, using the Riemann sum theorem supports the Euler approximation method. Besides, the analytical solution of nonlinear ODE is also not easy to find since the structure of the equation is nonlinear and complex. The direct integration technique cannot be applied to obtain the exact solution of a nonlinear ODE since many functions in nonlinear ODE have unknown derivatives. In addition, some non-autonomous and nonlinear ODEs, which have an explicit function of time, will burden the calculation procedure and be costly.

Therefore, three objectives of the study are established. First, to approximate the solution of nonlinear ordinary differential equations using a numerical integration approach of a linear model. Second, to predict the solution of growth models with the numerical integration approach of a linear model. Third, to verify the efficiency of the numerical integration approach of a linear model in solving nonlinear ordinary differential equations. For illustration, a nonlinear growth model is studied.

II. METHODOLOGY

Consider a general nonlinear ordinary differential equation (ODE) model in growth modelling given

$$\frac{dx}{dt} = f(t), \tag{1}$$

where x is a state vector, f is the dynamic function and dx/dt is the rate of change of the state with respect to time t [14]. By integrating both sides of (1), the solution is given by

$$x(t) = x(t_0) + \int_{t_0}^{t} f(\tau)\, d\tau, \tag{2}$$

with $x(t_0)$ is the initial value and τ is a dummy variable for integration. The ODE model (1) has a nonlinear and complex structure, which makes it difficult and expensive to solve the integral (2) directly using standard mathematical methods. However, the following numerical integration approaches can be used for solving the integral in (2).

(a) The Simpson's Rule

$$\int_{t_0}^{t} f(\tau)\, d\tau \approx \frac{t-t_0}{6}\left[f(t_0) + 4f\left(\frac{t_0+t}{2}\right) + f(t)\right]. \tag{3}$$

(b) The Gaussian Quadrature Rule

$$\int_{t_0}^{t} f(\tau)\, d\tau \approx \left(\frac{t-t_0}{2}\right) f\left(\frac{t-t_0}{2}\left(\frac{-1}{\sqrt{3}}\right) + \frac{t-t_0}{2}\right) + \left(\frac{t-t_0}{2}\right) f\left(\frac{t-t_0}{2}\left(\frac{1}{\sqrt{3}}\right) + \frac{t-t_0}{2}\right). \tag{4}$$

Notice that (3) is an adaptive quadrature, while (4) is a non-adaptive quadrature. Gauss quadrature and adaptive Simpson's rule are highly accurate for smooth and nonlinear integrands and have low computational cost. Thus, they are suitable for handling the smooth behavior of the nonlinear ODE.

Now, we assume that the solution from (2) is measurable by the output equation,

$$y(t) = x(t) \tag{5}$$

without knowing the ODE model (1). Also, consider a general linear model,

$$\frac{dx}{dt} = a \cdot t, \tag{6}$$

where a is the model parameter known as the growth rate. The solution of the linear model is given by

$$x(t) = x(t_0) + \int_{t_0}^{t} a \cdot \tau d\tau. \tag{7}$$

After handling the integral part, the solution of (7) is expressed by

$$x(t) = x(t_0) + \frac{1}{2} \cdot a \cdot (t^2 - t_0^2). \tag{8}$$

The transformation substitutes the solution model into the nonlinear ODE so that the rate of change depends only on time. This enables a direct and unique relationship between the model parameters and the output data by expressing the solution through a linear parameterized structure. In this way, parameter identifiability is preserved under the transformation.

Notice that the solution of the linear model in (6) will not give the solution to the nonlinear ODE in (2) due to the different structures of the models (1) and (6). Thus, the solution of the linear model in (6) is unable to predict the solution of nonlinear ODE in (2).

Nevertheless, we propose a least squares optimization problem,

$$\text{Minimize } J = \frac{1}{2}\int_{t_0}^{t} (y(\tau) - x(\tau))^{\mathrm{T}} (y(\tau) - x(\tau))\, d\tau \text{ , (9)}$$

where J is the objective function representing the differences between the nonlinear ODE model and the linear model and $^{\mathrm{T}}$ is the transpose operator. The objective function (9) is convex, as it has a quadratic form with respect to the model parameter a, and its Hessian matrix is positive definite. With this problem, we aim to estimate the model parameter a through the observation of the output $y(t)$ such that the objective function J is minimized, and the solution of the linear model is updated as closely as possible to the output $y(t)$. On this basis, the solution of the nonlinear ODE can be predicted using the linear model. This problem is regarded as a prediction of a nonlinear ODE through a numerical integral method of a linear model.

From the objective function (9), the gradient is derived by

$$\frac{\partial J}{\partial a} = (y(t) - x(t))^{\mathrm{T}} \cdot \frac{1}{2} \cdot (t^2 - t_0^2) \text{ , } \quad (10)$$

and the model parameter a is updated by the recursion equation,

$$a^{(i+1)} = a^{(i)} - \alpha \cdot \left(\frac{\partial J}{\partial a} \right)^{(i)} \text{ , } \quad (11)$$

where α is the step size and i is the iteration number. The recursion equation in (11) is known as the steepest gradient descent method [15] [16]. During the iterative procedure, the step size α and the initial value of the model parameter $a^{(0)}$ are required to start the iteration. Due to the convexity and smoothness of the objective function, the gradient descent method (11) is guaranteed to converge to a global minimum under appropriate step size selection. A small step size is chosen for the gradient descent method to prevent overshooting during the iterative process. The first-order necessary condition [17]

$$\frac{\partial J}{\partial a} = 0 \quad (12)$$

is satisfied, and the model parameter

$$a^{(i+1)} \approx a^{(i)} \quad (13)$$

is expressed when the linear model converges toward the nonlinear solution. This implies that two model parameters have an almost equivalent value within an accepted tolerance ε, which is mathematically written by

$$\left\| a^{(i+1)} - a^{(i)} \right\| < \varepsilon \text{ . } \quad (14)$$

Hence, we denote the model parameter obtained from (13) as $\hat{a}$, and the predictive model is given by

$$\frac{d\hat{x}}{dt} = \hat{a} \cdot t \quad (15)$$

while the prediction solution in the discrete time is represented by

$$\hat{x}_{k+1} = x_k + \frac{1}{2} \cdot \hat{a} \cdot (t_{k+1}^2 - t_k^2) \text{ , } \quad (16)$$

where

$$x_k = \hat{x}_k + \beta(y_k - \hat{x}_k) \quad (17)$$

is a line search equation that satisfies the first-order necessary condition $\partial J / \partial x = 0$, and β is the step size for the line search equation, for the time step $k = 0, 1, ..., n$. Here, the term y_k is the discrete time of the output $y(t)$. As a result, the solution (16) will approximate the solution of the nonlinear ODE in (1).

III. Results

In this section, we consider the hyperbolic growth model, as expressed by

$$\frac{dx}{dt} = r\frac{x^2}{x_0} \text{ , } \quad (18)$$

with the initial population $x(t_0) = x_0$ at time $t = t_0$, where r is the growth rate and x is the population at time t. While dx / dt is the rate of change of population growth x over time t. By using the separable variable method and integrating both sides, the solution of (18) is presented as follows.

$$x(t) = \frac{x_0}{1 - r(t - t_0)} \text{ . } \quad (19)$$

Then, we consider a hyperbolic growth model [18],

$$\frac{dx}{dt} = \frac{0.0113}{5.0402} x^2 \text{ , } \quad (20)$$

with the initial condition $x_0 = 5.0402$ for $0 < t < 200$ months. The exact solution of (20) is given by

$$x(t) = \frac{5.0402}{1 - 0.0113t} \text{ , } \quad (21)$$

where $t_0 = 0$. Figure 1 shows the exact solution curve of the model (20), and it is then further used as the observed data. The points in the figure are generated from (21), and they show a hyperbolic growth curve.

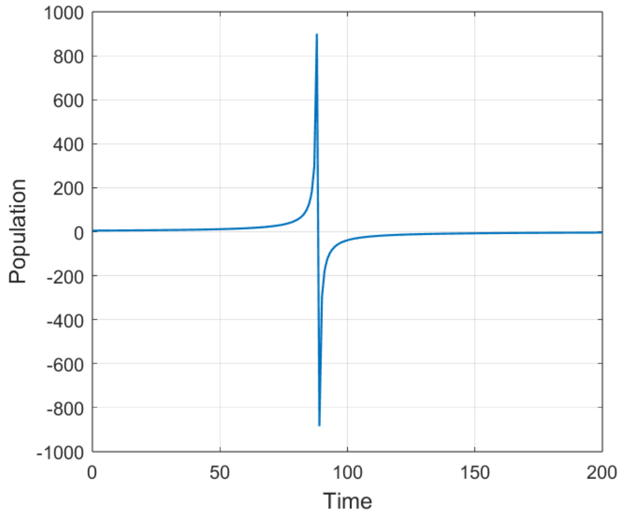

FIGURE 1. Exact solution curve of hyperbolic growth model

Substituting (21) into (20), the nonlinear hyperbolic growth model is expressed by

$$\frac{dx}{dt} = \frac{(0.0113)(5.0402)}{(1-0.0113t)^2}, \quad (22)$$

and we use the Gauss quadrature in (4) and Simpson's rule in (3) to approximate the solutions of (22). Notice that the approximate solution to (22) is similar to the exact solution of (21), as shown in Figure 2 and 3.

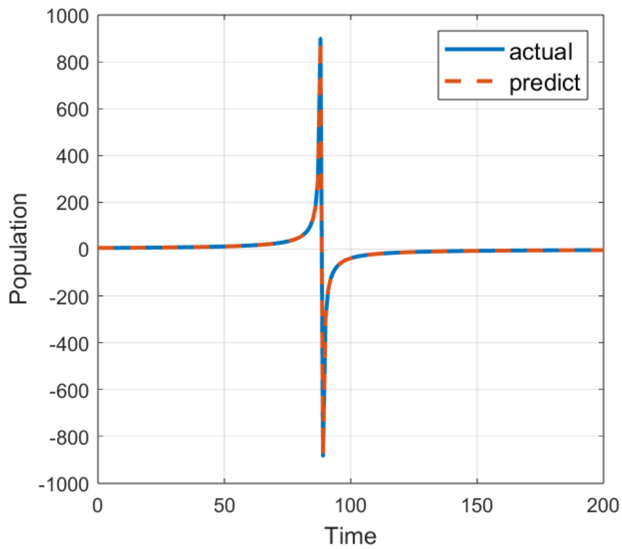

FIGURE 2. Solution curve of hyperbolic growth model by Gauss quadrature

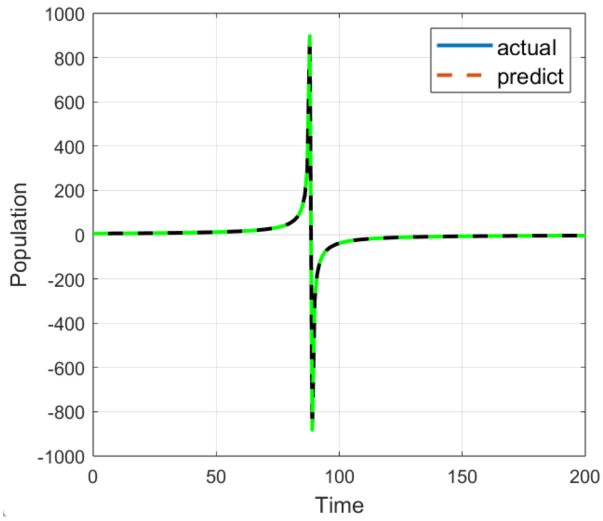

FIGURE 3. Solution curve of hyperbolic growth model by Simpson's rule

The numerical solution curves seem to be very close to the exact solution curve. Their accuracy is supported by numerical errors that are shown in Figures 4 and 5. Both figures are identical, as the mean square error values obtained from using the integration methods are the same.

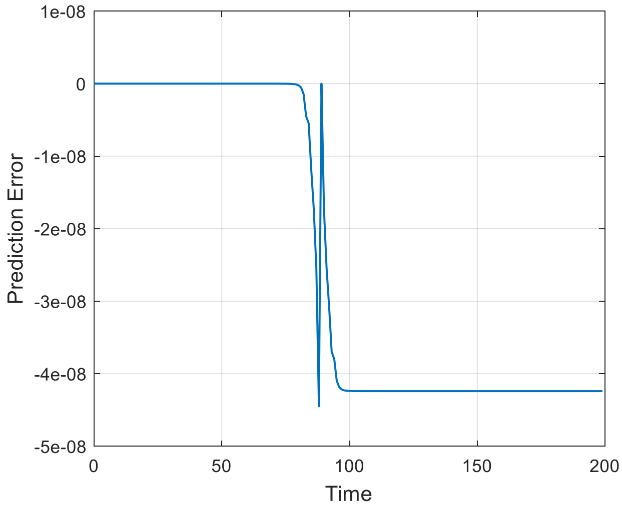

FIGURE 4. Numerical error of hyperbolic growth model by Gauss quadrature

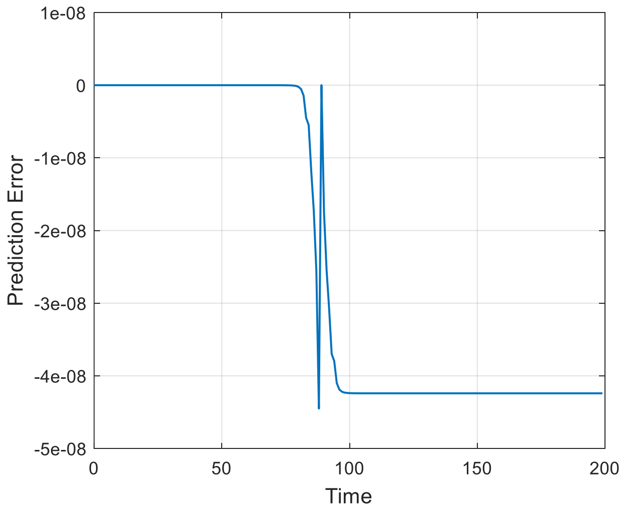

FIGURE 5. Numerical error of hyperbolic growth model by Simpson's rule

Now, referring to (6), we consider two linear models, given as follows,

$$\frac{dx}{dt} = 0.2 \cdot t \quad (23)$$

$$\frac{dx}{dt} = -0.2 \cdot t \quad (24)$$

with the initial condition $x_0 = 5.0402$ for $0 < t < 200$ months. These models with the respective model parameters $a = 0.2$ and $a = -0.2$ are chosen and known as the initial predictive models to begin the iterative procedure. After handling the integral part, the discrete-time models of (23) and (24) are given by

$$x_{k+1} = x_k + 0.1(t_{k+1}^2 - t_k^2)\,, \quad (25)$$

$$x_{k+1} = x_k - 0.1(t_{k+1}^2 - t_k^2)\,. \quad (26)$$

Figures 6 and 7 show the solution curves for the initial models (23) and (24), respectively, where the first solution curve is an exponential growth, and the second solution curve is a decay growth.

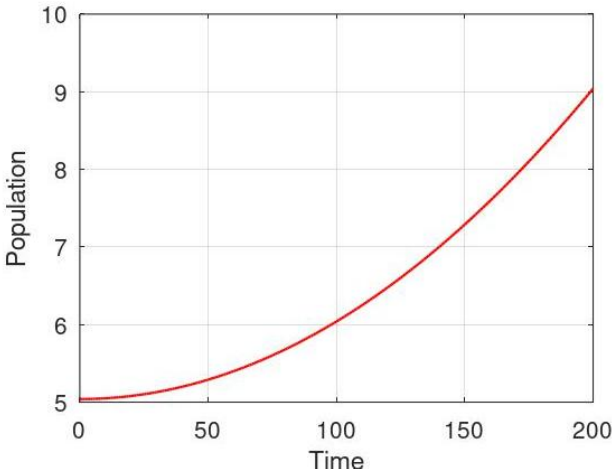

FIGURE 6. Exponential solution growth curve for linear model

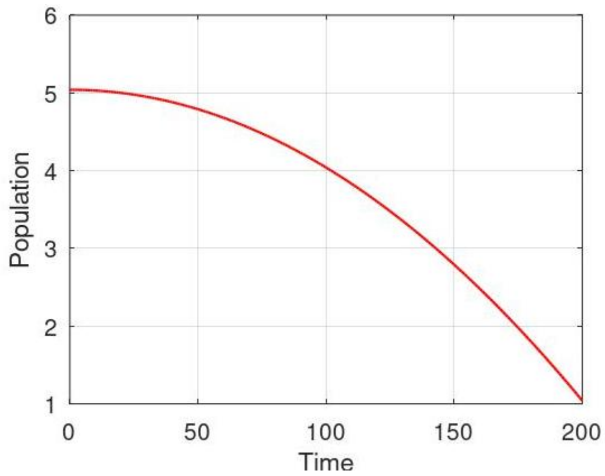

FIGURE 7. Decay solution growth curve for linear model

By using the algorithm proposed in the previous section, the predicted solution to the hyperbolic growth model in (22) is obtained and shown in Figure 8.

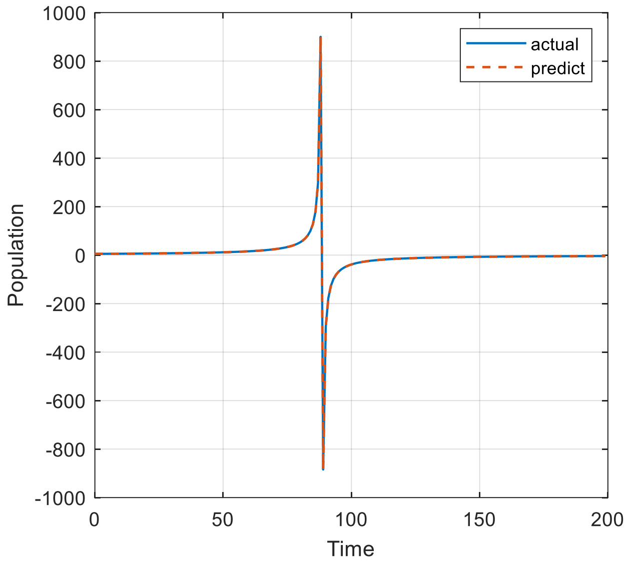

FIGURE 8. Final solution curve for hyperbolic growth model

The solution curve closely approximates the solution curve of the hyperbolic growth in (22), and the prediction error, which is very small error values, is shown in Figure 9.

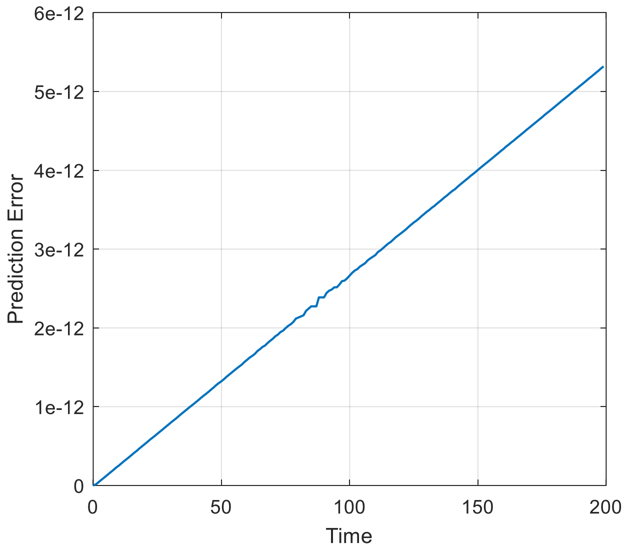

FIGURE 9. Prediction error for hyperbolic growth model

The prediction error curve shows a steady linear increase over time because the linear model cannot keep up with the steep, accelerating growth of the nonlinear hyperbolic model. This mismatch occurs because the hyperbolic model grows increasingly faster as it approaches the singularity, while the linear model assumes a constant or slowly changing rate of growth. As a result, the small difference between the two models at earlier time steps accumulates and is amplified at later time steps, leading to the observed increase in prediction error. Despite the prediction error showing a slight increase, the predictive model method is proven to be more efficient with a small magnitude of error.

Figure 10 shows the parameter estimation for the hyperbolic growth model.

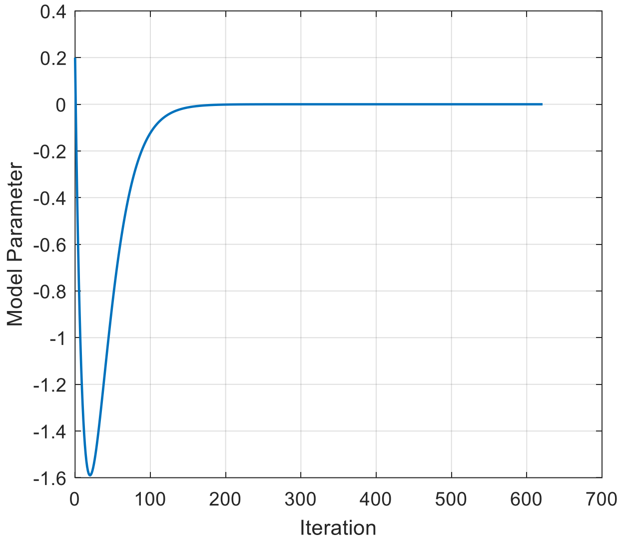

FIGURE 10. Parameter estimation for hyperbolic growth model

It shows an inverted bell-shaped curve, where the parameter initially moves further negative at the beginning of the iterations due to the steepness of the nonlinear hyperbolic curve requiring a highly negative adjustment. As iterations proceed, the predictive modelling method adjusts the parameter toward a stable value until the local minimum value is obtained, when the error is unable to be further reduced or equal to 0.

For using the predictive modelling method, the tolerance is set to 10^{-22}, the maximum iteration number is 10000, while the step size for the gradient descent method, $\alpha = 0.001$ and the step size for the line search equation, $\beta = 0.1$. Therefore, the final predictive models of the hyperbolic growth model, which use the initial models (23) and (24), are provided below.

$$\frac{d\hat{x}}{dt} = -1.4475 \times 10^{-12} t \qquad (27)$$

$$\frac{d\hat{x}}{dt} = -1.4458 \times 10^{-12} t \qquad (28)$$

with the line search equation

$$x = \hat{x} + 0.1(y - \hat{x}) \qquad (29)$$

where y represents the actual solution of the hyperbolic growth model.

Table 1 shows the simulation results of the prediction solution of the hyperbolic growth model (22).

TABLE 1. Simulation results for hyperbolic growth model.

Model	Iteration number	Mean square error	Elapsed time (sec)
Proposed method[a]	621	4.7131×10^{-24}	0.048868
Proposed method[b]	615	4.7019×10^{-24}	0.039363
Gauss quadrature	-	9.8231×10^{-16}	0.217478
Simpson's rule	-	9.8231×10^{-16}	0.217478

a. using the growth model as the initial model
b. using the decay model as the initial model

In Table 1, the mean square error values for the proposed method, Gauss quadrature, and Simpson's rule are compared. The efficiency of the proposed method is highly demonstrated since the mean square error values are smaller compared to the standard integration solvers. The proposed method updates the model parameter during the iteration process to minimize the difference between the prediction and actual data, while the standard integration solvers only approximate the integral using the given function without adjusting the model to fit the actual data.

In summary, a linear model in terms of time was proposed and integrated since the integrand might not often be straightforward to reform in its independent variable. The outcome of integrating the linear model provided us with a quadratic term. Then, a least square optimization problem was introduced to predict the nonlinear growth model, and the optimization problem was solved using the steepest descent method. The model parameter was estimated by observing the nonlinear growth solution curves, while the prediction solutions were obtained by integrating the linear model.

A mean square error presented the performance of using a numerical integration approach and integrating a linear model to solve the nonlinear ODEs of the growth models. The numerical integration approaches showed a very small mean square error value, while integrating a linear model indicated a satisfactory mean square error value. The solution curves of the integrating linear model, which best fit the growth models' solution, expressed a useful prediction model for managing the nonlinear ODE of the growth models.

IV. Discussion

In this study, the major strength of the proposed method is the ability to obtain the numerical solution of a nonlinear ODE without knowing its complexity and structure. The complexity of solving the nonlinear ODE is significantly reduced by introducing a simple linear model. The model parameter in the linear model is then optimized so that the predicted solution best fits the observed data. This approach bypasses the difficulties related to nonlinear model formulation, analytical solution derivation, and direct numerical integration of complex dynamics.

While the method significantly simplifies solving nonlinear ODEs, the accuracy of the approximation depends heavily on the appropriateness of the linear model chosen to fit the observed data. A simple linear model may not be sufficient to capture the complex dynamics of highly nonlinear or chaotic systems. Since the method assumes that the system's solution is smooth and continuous, the method may also perform poorly for dynamics that exhibit sharp transitions or nonsmooth behavior.

The formulation of the method based on least squares optimization and the gradient descent method may allow for a straightforward extension to higher-dimensional problems, such as nonlinear partial differential equations (PDEs). By appropriately redefining the model and optimization objective, the technique may offer a flexible and powerful tool to approximate complex dynamic systems in PDE contexts.

V. Conclusion

This study considered a general nonlinear ODE, and its solution was expressed in an integral form. Reforming the integrand in the independent variable, that is, time, would allow us to find the numerical solution of the integral using a numerical integration method. For illustration, a hyperbolic growth model is studied. The exact solution of the growth model was substituted into the respective ODE model, and numerical integration methods, which are Gauss quadrature and Simpson's rule, were applied to give the solution of the ODE by obtaining the solution of the integral. In conclusion, this study verified and compared the efficiency of these approaches in handling an integral in the nonlinear ODE. Integrating an implicit linear ODE to capture the solution of an integral in an implicit nonlinear ODE and using more advanced optimization, such as conjugate gradient methods, for solving more complex nonlinear ODEs with higher dimensions is recommended to be studied.

Acknowledgement

We thank the anonymous reviewers for the careful review of our manuscript.

Funding Statement

There are no funding agencies supporting the research work.

Author contributions

Hui Shan Tai: Conceptualization, Methodology, Validation, Writing – Original Draft Preparation;

Srimazzura Basri: Project Administration, Supervision, Writing – Review & Editing.

Conflict of interests

No conflicts of interest were disclosed.

Ethics statements

This research did not involve human participants, animal subjects, or sensitive personal data, and therefore did not require ethical approval.

References

[1] T. Amir, "2.5 Applications of First-Order ODE", *Differential Equations*, Canada: eCampus Ontario, 2024. URL:https://ecampusontario.pressbooks.pub/diffeq/chapter/2-5-applications-of-first-order-ode/ (Accessed 6 June, 2025)

[2] P.J. Olver, "Nonlinear Ordinary Differential Equations," *United States: University of Minnesota*, 2022. URL: https://www-users.cse.umn.edu/~olver/ln_/odq.pdf

[3] G. Sarma and Vezhopalu, "Analytical and Numerical Investigation of Nonlinear Differential Equations: A Comprehensive Study", *Journal of Computational Analysis and Applications (JoCAAA),* vol. 34, no.1, pp. 213 – 222, 2025. URL:https://eudoxuspress.com/index.php/pub/article/view/1742

[4] F.S. Guzmán, "Ordinary Differential Equations", *Numerical Methods for Initial Value Problems in Physics*, New York: Springer Cham, pp. 7-97, 2023. DOI: https://doi.org/10.1007/978-3-031-33556-3

[5] M. Al-Mazmumy, A.A. Alsulami, H.O. Bakodah and N. Alzaid, "Modified Adomian Method Through Efficient Inverse Operators to Solve Nonlinear Initial-Value Problems for Ordinary Differential Equations," *Axioms 2022*, vol. 11, no.12, pp. 698-1 – 698-16, 2022. DOI: https://doi.org/10.3390/axioms11120698

[6] K.A. Koroche and G.K. Mayu, "Adam-Bash forth-Multan Predictor Corrector Method for Solving First Order Initial Value Problem and its Error Analysis," *Indian Journal of Advanced Mathematics (IJAM),* vol. 1, no. 2, pp. 30-40, 2021. DOI: https://doi.org/10.54105/ijam.B1112.101221

[7] N. Sharma, S. Singh and D. Kumar, "Numerical Solution of Nonlinear Dual-Phase-Lag Model for Analyzing Heat Transfer in Tissue During Thermal Therapy," *Computational and Mathematical Methods*, vol. 3, no. 6, pp. 1-15, 2021. DOI: https://doi.org/10.1002/cmm4.1183

[8] R. Capuano, N. Vaiana, D. Pellecchia and L. Rosati, "A Solution Algorithm for a Modified Bouc-Wen Model Capable of Simulating Cyclic Softening and Pinching Phenomena," *Internal Federation of Automatic Control,* vol. 55, no. 20, pp. 319-324, 2022. DOI: https://doi.org/10.1016/j.ifacol.2022.09.115

[9] V. Cornells, F.J. Vermolen, M.B.V. Gijzen and V. Thea, *Numerical Methods for Ordinary Differential Equations,* TU Delft OPEN Publishing, 2023. DOI: https://doi.org/10.5074/t.2023.001

[10] A. Televnoy, S. Ivanov, T. Zudilova and T. Voitiuk, "Neural ODE Machine Learning Method with Embedded Numerical Method," *Conference of Open Innovation Association, FRUCT 2021,* pp. 451 – 457, 2021. DOI: https://doi.org/10.23919/FRUCT50888.2021.9347610

[11] F.A.C. Viana, R.G. Nascimento, A. Dourado and Y.A. Yucesan, "Estimating Model Inadequacy in Ordinary Differential Equations with Physics-Informed Neutral Networks," *Computers & Structures,* vol. 245, pp. 1 – 12, 2021. DOI: https://doi.org/10.1016/j.compstruc.2020.106458

[12] M.K. Jaradat and W.S. Yang, "An Adaptive Time-Stepping Control Algorithm for Molten Salt Reactor Transient Analyses," *Annals of Nuclear Energy,* pp. 1 – 22, vol. 190, 2023. DOI: https://doi.org/10.1016/j.anucene.2023.109880

[13] M. Ratchagit, "Forecasting PM2.5 Concentrations in Chiang Mai using Machine Learning Models," *International Journal on Robotics, Automation and Sciences*, vol. 6, no. 2, pp. 37 – 41, 2024. DOI: https://doi.org/10.33093/ijoras.2024.6.2.6

[14] S.L. Kek, W.J. Leong and C.M.L. Kon, "Curve Tracking of Nonlinear Dynamic System Using Linear State-Space Model," *11th International Conference of Control Systems, and Robotics (CDSR 2024)*, pp 141-1 – 141-6, 2024. DOI: https://doi.org/10.11159/cdsr24.141

[15] J. Nocedal and S.J. Wright, "Numerical Optimization," *New York: Springer Science+Business Media*, 2nd edition, 2006. URL:https://www.math.uci.edu/~qnie/Publications/NumericalOptimization.pdf (Accessed 2 June, 2025)

[16] E.K.P. Chong, W.-S. Lu and S.H. Zak, "An Introduction to Optimization: With Applications to Machine Learning," *John Wiley & Sons*, 5th edition, pp. 109-112, 2024. URL: https://www.engr.colostate.edu/~echong/book5/ (Accessed 2 June, 2025)

[17] P. Pedregal, "Introduction to Optimization," *New York: Springer-Verlag*, 2004. URL:https://doc.lagout.org/science/0_Computer%20Science/3_Theory/Non-Linear%20Systems%2C%20Modeling%2C%20Optimization/Introduction%20to%20Optimization.pdf (Accessed 6 June, 2025)

[18] D. Hathout, "Modelling Population Growth: Exponential and Hyperbolic Modelling," *Applied Mathematics,* vol. 4, pp. 299-304, 2013. DOI: https://doi.org/10.4236/am.2013.42045