

Accuracy Study on Numerical Solutions of Initial Value Problems (IVP) in Ordinary Differential Equations (ODE)

Anthony Anya Okeke¹, Buba M. T. Hambagda², Pius Tumba³

^{1, 2, 3}Department of Mathematics, Faculty of Science, Federal University Gashua,

P. M. B. 1005 Gashua, Yobe State, Nigeria

Corresponding Author: Anthony Anya Okeke

ABSTRACT: In this work, we solve numerically initial value problems (IVP) in Ordinary Differential Equations (ODE) by Euler method. The proposed method is presented from the point of view Taylor's algorithm which invariably simplifies the demanding analysis. The method is quite efficient and practically well suited for solving these problems. Many examples are considered to validate the accuracy and easy application of the proposed method. We compared the approximate solutions with the analytical solution and found out that the approximate solutions converge to the exact solutions monotonically. In addition, to achieve more accuracy in the solutions, the step size needs to be very small. Lastly, we analyzed the error terms for the method for different steps sizes and compared also by appropriate examples to demonstrate the reliability and efficiency.

Keywords: Ordinary Differential Equations (ODE), Initial Value Problems (IVP), Euler Method, Error Analysis

Date of Submission: 16-08-2019

Date Of Acceptance: 13-09-2019

I. INTRODUCTION

We know that mathematics is a science of communication between us and the scientific sciences; in particular it introduces all the rules and problems as formulas, and also searches for solutions. A unit of mathematics that is extensively used in all sciences is the differential equations. According to [1], differential equations are among the most important mathematical tools used in producing models in the engineering, mathematics, physics, aeronautics, elasticity, astronomy, dynamics, biology, chemistry, medicine, environmental sciences, social sciences, banking and many other areas. Many researchers have studied the nature of Differential Equations and many complicated systems that can be described quite precisely with mathematical expressions. Although there are many analytic methods for finding the solution of differential equations, there exist quite a number of differential equations that cannot be solved analytically [2]. This means that the solution cannot be expressed as the sum of a finite number of elementary functions (polynomials, exponentials, trigonometric, and hyperbolic functions). For simple differential equations, it is possible to find closed form solutions [3]. But many differential equations arising in applications are so complicated that it is sometimes impractical to have solution formulas; even when a solution formula is available, it may involve integrals that can be calculated only by using a numerical quadrature formula. In either case, numerical methods provide a powerful alternative tool for solving the differential equations under the prescribed initial condition or conditions [3].

We have many types of practical numerical methods for solving initial value problems for ordinary differential equations. From history, the ancestor of all numerical methods in use today was developed by Leonhard Euler between 1768 and 1770 [4], improved Euler's method and Runge Kutta methods described by Carl Runge and Martin Kutta in 1895 and 1905 respectively [5]. There are excellent and far-reaching books which can be consulted, such as [1-3, 6-15].

From the literature review, we realize that many authors have worked on numerical solutions of initial value problems using the Euler's method and many others have tried to solve initial value problems to get a higher accurate solution by applying numerous methods, such as the Euler method's, the Runge Kutta method, the Adomian Decomposition Method, Hybrid method, Extrapolation method, and also some other methods. See [13-21]. In this paper, Euler's method is applied without any discretization, transformation or restrictive assumption for solving initial value problems in ordinary differential equations.

Euler's method historically is the first numerical technique. It is also called the tangent line method. It is the simplest to understand and geometrically easy to articulate. The method needs to take a smaller value of h , and the numerical results are very encouraging. Finally, we used two examples of different kind of ordinary differential equations to illustrate the proposed formulation. The results obtained from each of the numerical

examples show that the convergence and error analysis which we presented clearly illustrate the efficiency of the methods.

However, the numerical technique has its own advantages and disadvantages to use. The method requires less time consumption, it is simple and single step. Also, in Euler's method $\frac{dy}{dx}$ changes rapidly over an interval, this gives a poor approximation at the beginning of the process in comparison with the average value over the interval. So the calculated value of y in this method occurs much error than the exact value, which reasonably increased in the succeeding intervals, then the final value of y differs on a large scale than the exact value.

Generally, Euler's method needs to take a smaller value of h , and as such, the method is suitable for practical use. If h is not too small enough, this method is inaccurate.

Lastly, this paper is structured as follows: Section 2: problem formulations; Section 3: numerical examples; Section 4: discussion of results; and the last section, the conclusion of the paper.

II. DEFINITIONS AND GENERAL CONCEPTS OF DIFFERENTIAL EQUATIONS

A differential equation is a mathematical equation for an unknown function of one or more variables that relates the values of the function itself and its derivatives of various orders.

2.1 Definition: The general form of a differential equation is as follows:

$$a_0(t) \frac{d^n y}{dt^n} + a_1(t) \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_{n-1}(t) \frac{dy}{dt} + a_n(t)y = f(t) \tag{1}$$

Here $a_i(t)$; $n = 1, 2, 3, \dots$ and $f(t)$ is the function of t and $y = y(t)$ is an unknown function in terms of t .

2.2 Definition: An equation involving a function $y(t)$ of one independent variable (t) and its derivatives $y'(t) \dots y^{(n)}(t)$ is called an ordinary differential equation (ODE) of order n .

2.3 Definition: An implicit ODE of order n depending on $y^{(n)}$ has the form

$$F(t, y, y'(t), y''(t), \dots, y^{(n)}(t)) = 0. \text{ In a special form, } F(t, y, y', y'', \dots, y^{(n-1)}) = y^{(n)} \text{ is called an ODE in explicit form.}$$

2.4 Definition: A partial differential equation for the function $u = u(t_1, t_2, t_3, \dots, t_n)$ is of the form

$$F\left(t_1, t_2, t_3, \dots, t_n, \frac{\partial u}{\partial t_1}, \frac{\partial u}{\partial t_2}, \dots, \frac{\partial u}{\partial t_n}, \frac{\partial^2 u}{\partial t_1 \partial t_2}, \frac{\partial^3 u}{\partial t_2 \partial t_3}, \dots\right) = 0, \text{ where } F \text{ is a linear function of } u \text{ and its derivatives.}$$

2.5 Definition: An initial value problem (IVP) is a differential equation such as $y'(t) = f(t, y(t))$, satisfies the following initial conditions (IC) $y(t_0) = y_0$; $y'(t_0) = y'_0$, where $t_0 \in I$, for some open interval $I \in \mathbb{R}$.

2.6 Definition: A boundary value problem is a differential equation together with a set of additional restraints, called boundary conditions. A solution to a boundary value problem is a solution to the differential equation which also satisfies given boundary conditions. The basic two point boundary value problem is given by $y'(t) = f(t, y(t))$ with $g(y(a), y(b)) = 0$.

2.7 Definition: A set of first order differential equation:

$$\begin{aligned} y &= f(t, y) \\ y_1 &= f_1(t, y_1 \dots y_n) \\ &\dots \\ y_n &= f_n(t, y_1 \dots y_n) \end{aligned}$$

is called a first order system of ordinary differential equations.

2.8 Definition: A set of first order differential equation is called autonomous if all functions f_k on the right-hand side do not depend on t .

$$\begin{aligned} y &= f(y) \text{ or} \\ y_1 &= f_1(y_1 \dots y_n) \\ &\dots \\ y_n &= f_n(y_1 \dots y_n) \end{aligned}$$

2.9 Definition: A solution of a system of ordinary differential equation is the form $y(t) = y_0$ with $y_0 \in \mathbb{R}^2$ is called an equilibrium solution. $y(t) = y(t + T)$ with $T > 0$ is called a period. The parameter T denote period.

2.10 Definition: An equilibrium solution $y(t) = y_0$ of an autonomous system is called stable if any solution in the neighborhood of y_0 will always approach this equilibrium solution as $t \rightarrow \infty$. In this case $y(t) \rightarrow y_0$ for $t \rightarrow \infty$. Otherwise the equilibrium solution is called unstable.

III. PROBLEM FORMULATION

In this section, discussed our proposed method - Euler's method- for finding the approximate solutions of the initial value problem (IVP) of the first order ordinary differential equation of the form

$$y' = f(x, y), \quad x \in (a, b), \quad y(a) = y_0 \tag{2}$$

Where $y' = \frac{dy}{dx}$ and $f(x, y)$ is a given function and $y(x)$ is the solution of the equation (3.1)

Theorem 3.1 [1]

Let $f(x, y)$ be defined and continuous for all points (x, y) in the region D defined by $a \leq x \leq b, -\infty < y < \infty$, a and b finite, and let there exist a constant L such that, for every x, y, y^* , such that (x, y) and (x, y^*) are both in D ,

$$|f(x, y) - f(x, y^*)| \leq L(y - y^*) \tag{3}$$

Then, if y_0 is any given number, there exists a unique solution $y(x)$ of the initial value problem (2), where $y(x)$ is continuous and differentiable for all (x, y) in D .

In this paper, we determine the solution of this equation in the range $a \leq x \leq b$, where a and b are finite, and we assume that f satisfies the Lipchitz conditions stated in Theorem 3.1.

A continuous approximation to the solution $y(x)$ will not be obtained; instead, an approximation to y will be generated at various values, called mesh points, in the interval $a \leq x \leq b$. Numerical techniques for the solution of (1) is to obtain approximations to the values of the solution corresponding to the sequence of points $\{x_n\}$ defined by $x_n = a + nh, n = 0, 1, 2, 3, \dots$. The parameter h is called the step size. The numerical solution of (3.1) is given by a set of points $\{(x_n, y_n): n = 0, 1, 2, 3, \dots, N\}$ and each point (x_n, y_n) is an approximation to the corresponding point $(x_n, y(x_n))$ on the solution curve.

3.1. Euler’s Method

Euler’s method is also called the tangent method and it is the simplest one-step method. It is the most basic example method for numerical integration of ordinary differential equations. The Euler method is named after Leonhard Euler, who treated it in his book *Institutiones Calculi Integralis* published 1768-1870, republished in his collected works (Euler 1913) [2]. The Euler method is subdivided into three namely:

- Forward Euler’s method
- Improved Euler’s method
- Backward Euler’s method

In this paper, we shall only consider the forward Euler’s method

3.2. Derivative of Euler’s Method

Let us consider the initial value problem

$$y' = \frac{dy}{dx} = f(x, y); \quad y(x_0) = y_0 \tag{4}$$

We know that if the function f is continuous in the open interval $a < x < b$ containing $x = x_0$, there exists a unique solution of the equation (4) as

$$y_n = y(x_n); n = 1, 2, 3, \dots \tag{5}$$

The solution is valid for throughout the interval $a < x < b$. We wish to determine the approximate value of y_n of the exact solution $y = y(x)$ in the given interval for the value $x = x_n = x_0 + nh; n = 0, 1, 2, \dots$. Now, the equation of the tangent line through (x_0, y_0) of (3) is

$$y(x) = y_n + f(x_0, y_0)(x - x_0),$$

Setting $x = x_1$, we have

$$y_1 = y_0 + hf(x_0, y_0),$$

Similarly, we get the next approximation as

$$\begin{aligned} y_2 &= y_1 + hf(x_1, y_1), \text{ at } x = x_2 \\ y_3 &= y_2 + hf(x_2, y_2), \text{ at } x = x_3 \end{aligned}$$

In general, the $(n + 1)^{th}$ approximation at $x = x_{n+1}$ is given by

$$y_{n+1} = y_n + hf(x_n, y_n); n = 0, 1, 2, 3, \dots \tag{6}$$

3.3. Truncation Error for Euler’s Method

Numerical stability and errors are well discussed in depth in [10, 12-14]. There are two types of errors in the numerical solution of ODEs: Round-off errors and truncation errors. Round-off error occurs because computers use a fixed number of bits and hence fixed the number of binary digits to represent numbers. In a numerical computation round-off errors are introduced at every stage of computation. Hence though an individual round-off error due to a given number at a given numerical step may be small but the cumulative effect can be significant. When the number of bits required for representing a number is less than the number is usually rounded to fit the available number of bits. This is done either by chopping or by symmetric rounding.

Also, truncation error arises when you use an approximation in place of an exact expression in a mathematical procedure. To estimate the truncation error for the Euler method, we first recall Taylor’s Series approximation of a function.

Essentially, Taylor's Theorem State, that, any smooth function can be approximated by a polynomial. The Taylor Series Expansion of $f(x)$ at a is

$$f(x) = f(a) + f'(a)(x - a) + \frac{f''(a)(x - a)^2}{2!} + \frac{f'''(a)(x - a)^3}{3!} + \dots$$

$$= \sum_{n=0}^{\infty} \frac{f^{(n)}(a)(x - a)^n}{n!} \tag{7}$$

We note that computers are discrete, finite machine. They can't perform infinite calculation like these, so we have to cut the calculation somewhere. This result in truncation error (we truncate the expression). When we truncate the Taylor's Series expression to n terms, there's error left over, and we can include a remainder term R_n to keep the = sign exact.

$$f(x) = f(a) + f'(a)(x - a) + \frac{f''(a)(x-a)^2}{2!} + \frac{f'''(a)(x-a)^3}{3!} + \dots + \frac{f^{(n)}(a)(x-a)^n}{n!} + R_n \tag{8}$$

Where $R_n = \frac{f^{(n+1)}(\zeta)h^{n+1}}{(n+1)!}$, and $x \leq \beta \leq a$.

In (3.7), let $x = x_{n+1}$ and $x = a$, in which

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{1}{2}h^2y''(\beta_n) \tag{9}$$

Since y satisfies the ordinary differential equation (3), which can be written as

$$y'(x_n) = f(x_n, y(x_n)) \tag{10}$$

Hence,

$$y(x_{n+1}) = y(x_n) + hf(x_n, y(x_n)) + \frac{1}{2}h^2y''(\beta_n) \tag{11}$$

By considering (11) to Euler's approximation in (6), it is very clear that Euler's method is obtained by omitting the remainder term $\frac{1}{2}h^2y''(\beta_n)$ in the Taylor's expansion of $y(x_{n+1})$ at the point x_n . Therefore, the truncation T_{n+1} error is given by

$$T_{n+1} = y(x_{n+1}) - y(x_n) = hy'(x_n) + \frac{1}{2}h^2y''(\beta_n) \tag{12}$$

Thus, the truncation error is of $O(h^2)$: $h \rightarrow 0$, i.e. the truncation error is proportional to h^2 . By diminishing the size h , the error can be minimized. If M is positive constant such as $|y''(x)| < \frac{M}{2}$

$$|T_{n+1}| < \frac{Mh^2}{2} \tag{13}$$

Here the right hand size is an upper bound of the truncation error. The absolute value of T_{n+1} is taken for the magnitude of the error only.

3.4. Algorithm of the Euler Method

- (i) Define the function $f(t, y)$, such that $f(t, y) \in [a, b]$
- (ii) Give the initial value for t_0 and y_0 .
- (iii) Specify the step size $h = \frac{b-a}{n}$, where n is number of steps
- (iv) output t_0 and y_0
- (v) for i from 1 to n do
- (vi) Let $k_i = f(t_i, y_i)$, $y_{i+1} = y_i + h * k_i$, and $t_{i+1} = t_i + h$
- (vii) output t_i and y_i
- (viii) End

IV. NUMERICAL EXAMPLES

In this section, we present two numerical examples to verify the accuracy of the proposed method. The numerical results and errors are computed and the findings are represented graphically. The computations were done using MATLAB programming language. The convergence of the IVP is calculated $e_n = |y(x_n) - y_n| < \delta$, where $y(x_n)$ represents the exact solution and δ depends on the problem which varies from 10^{-4} while the absolute error is computed by $|y(x_n) - y_n|$.

Example 1: We consider the initial value problem $y'xy = x$, $y(0) = 5$, on the interval, $0 \leq x \leq 1$. The exact solution of the given problem is given by $y(x) = 1 + 4e^{-\frac{x^2}{2}}$. The results obtained are shown in Tables 1(a) and Table 1(b) and graphically displayed in Figures 1-3.

Table 1. (a) Numerical approximations for different step size.

| n | x_n | Exact Solution y_n | Approximation | | | |
|---|-------|----------------------|-------------------|-------------------|--------------------|--------------------|
| | | | $h = 0.1$ | $h = 0.05$ | $h = 0.0125$ | $h = 0.003125$ |
| 0 | 0.0 | 5.000000000000000 | 5.000000000000000 | 5.000000000000000 | 5.000000000000000 | 5.000000000000000 |
| 1 | 0.1 | 4.980049916770730 | 5.000000000000000 | 4.990000000000000 | 4.9825314154214064 | 4.9818753212959335 |

| | | | | | | |
|----|-----|-------------------|--------------------|--------------------|--------------------|--------------------|
| 2 | 0.2 | 4.920794693227022 | 4.960000000000000 | 4.940274625000007 | 4.9256392572578651 | 4.9244186046076761 |
| 3 | 0.3 | 4.823989927332399 | 4.8807999999999998 | 4.8521109802656257 | 4.8309640466998092 | 4.8292824130117262 |
| 4 | 0.4 | 4.692465385546543 | 4.7643759999999995 | 4.7279285525393124 | 4.7012362998361308 | 4.6992400289496548 |
| 5 | 0.5 | 4.529987610338382 | 4.6138009599999998 | 4.5711691569050350 | 4.5401455856907367 | 4.5325184756621679 |
| 6 | 0.6 | 4.341080845645088 | 4.4331109120000001 | 4.3861379549628925 | 4.3521659890635460 | 4.3500896614836799 |
| 7 | 0.7 | 4.130818152967473 | 4.2271242572799999 | 4.1778058172838000 | 4.1423495142161606 | 4.1405268578679522 |
| 8 | 0.8 | 3.904596148294764 | 4.0012255592703996 | 3.9515857656659099 | 3.9161007529283416 | 3.9147156382819390 |
| 9 | 0.9 | 3.667907243433898 | 3.7611275145287677 | 3.7130976358001044 | 3.6789468258247888 | 3.6781574438768523 |
| 10 | 1.0 | 3.426122638850534 | 3.5126260382211787 | 3.4679353506851176 | 3.4363161765600831 | 3.4362445949120213 |

Table 1(b) Observed absolute errors for example 1.

| n | x_n | Errors | | | |
|----|-------|--------------------|--------------------|--------------------|--------------------|
| | | h = 0.1 | h = 0.05 | h = 0.0125 | h = 0.003125 |
| 0 | 0.0 | 0.000000000000000 | 0.000000000000000 | 0.000000000000000 | 0.000000000000000 |
| 1 | 0.1 | 0.0199500832292703 | 0.0099500832292705 | 0.0024814986506767 | 0.0006008844316403 |
| 2 | 0.2 | 0.0392053067729785 | 0.0194799317729792 | 0.0048445640308437 | 0.0011918051852371 |
| 3 | 0.3 | 0.0568100726676004 | 0.0281210529332263 | 0.0069741193674098 | 0.0017245034260940 |
| 4 | 0.4 | 0.0719106144534569 | 0.0354631669927699 | 0.0087709142895882 | 0.0021742278534251 |
| 5 | 0.5 | 0.0838133496616180 | 0.0411815465666532 | 0.0101579753523549 | 0.0025218045068556 |
| 6 | 0.6 | 0.0920300663549121 | 0.0450571093178045 | 0.0110851434184580 | 0.0027547570356079 |
| 7 | 0.7 | 0.0963061043125268 | 0.0469876643163269 | 0.0115313612486876 | 0.0028678646368165 |
| 8 | 0.8 | 0.0966294109756358 | 0.0469896173711462 | 0.0115046046335778 | 0.0028631332492144 |
| 9 | 0.9 | 0.0932202710948702 | 0.0451903923662069 | 0.0110395823908913 | 0.0027492131886087 |
| 10 | 1.0 | 0.0865033993706450 | 0.0418127118345830 | 0.0101935377095490 | 0.0025403475141204 |

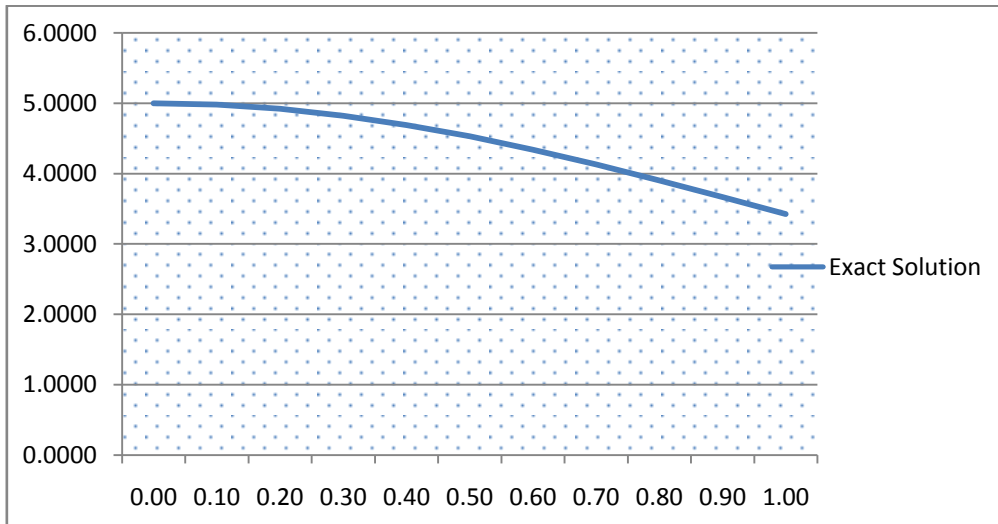


Figure 1: Exact Numerical Solution

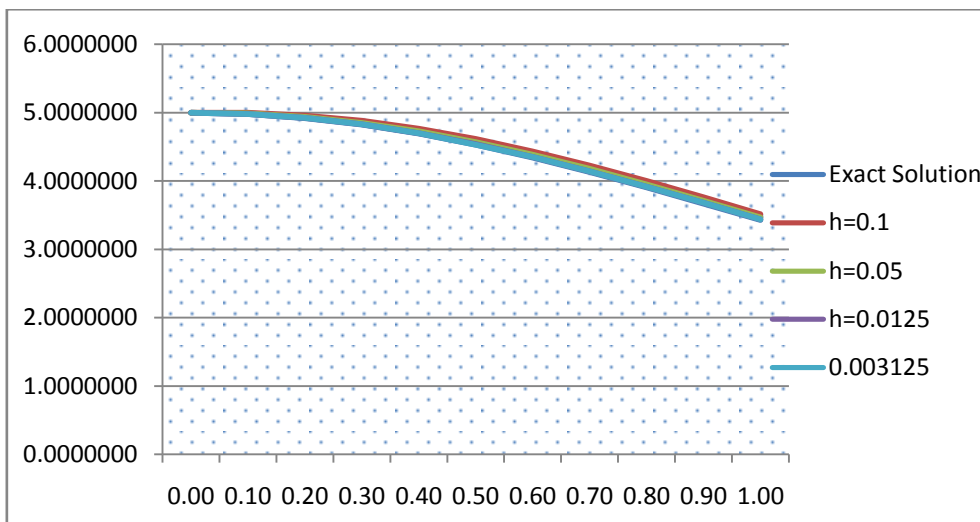


Figure 2: Exact Solution and Numerical Approximation for different step sizes

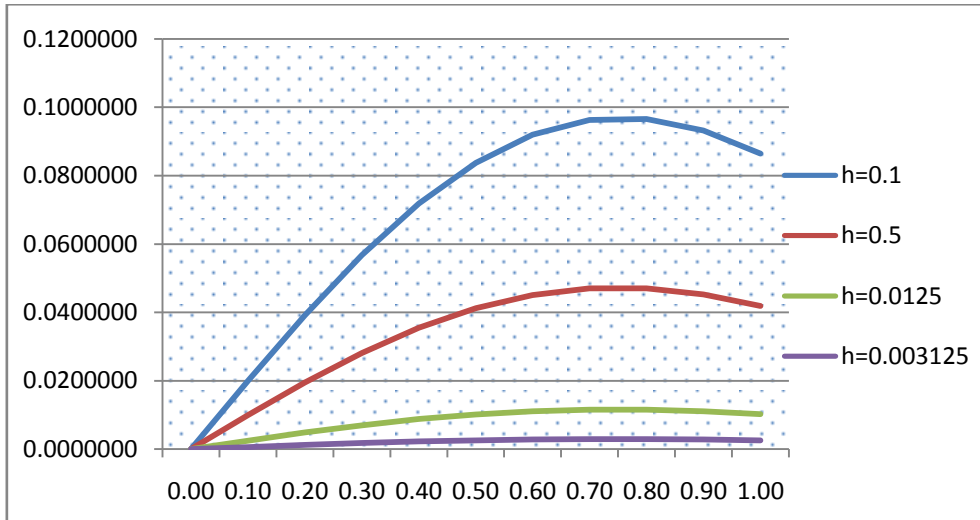


Figure 3: Error for different step sizes

Example 2: We consider the initial value problem $y' + \frac{1}{2}y = \frac{3}{2}$, $y(0) = 4$, on the interval $0 \leq x \leq 1$. The exact solution of the given problem is given by $y(x) = 3 + e^{-\frac{x}{2}}$. The results obtained are shown in Tables 2(a) and Table 2(b) and graphically displayed in Figures 4-6.

Table 2. (a) Numerical approximations for different step size.

| n | x_n | Exact Solution y_n | Approximation | | | |
|----|-------|----------------------|--------------------|--------------------|--------------------|--------------------|
| | | | $h = 0.1$ | $h = 0.05$ | $h = 0.0125$ | $h = 0.003125$ |
| 0 | 0.0 | 4.000000000000000 | 4.000000000000000 | 4.000000000000000 | 4.000000000000000 | 4.000000000000000 |
| 1 | 0.1 | 3.951229424500714 | 3.950000000000002 | 3.950625000000001 | 3.9510801844041317 | 3.9526807928162908 |
| 2 | 0.2 | 3.904837418035959 | 3.902500000000003 | 3.9036878906250001 | 3.9045535171661969 | 3.9061825669182642 |
| 3 | 0.3 | 3.860707976425058 | 3.857375000000002 | 3.8590683010253906 | 3.8603029259098327 | 3.8619538157781688 |
| 4 | 0.4 | 3.818730753077982 | 3.814506250000000 | 3.8166518036622619 | 3.8182170654177376 | 3.8198837713919072 |
| 5 | 0.5 | 3.778800783071405 | 3.7737809374999998 | 3.7763296208564379 | 3.7781900374601092 | 3.7798670720947487 |
| 6 | 0.6 | 3.740818220681718 | 3.7350918906249997 | 3.7379983458266510 | 3.7401211243290184 | 3.7418034986899582 |
| 7 | 0.7 | 3.704688089718713 | 3.6983372960937495 | 3.7015596775014599 | 3.7039145354082361 | 3.7055977234564015 |
| 8 | 0.8 | 3.670320046035640 | 3.6634204312890621 | 3.6669201684248254 | 3.6694791661408139 | 3.6711590714065148 |
| 9 | 0.9 | 3.637628151621773 | 3.6302494097246090 | 3.6339909851088494 | 3.6367283687879297 | 3.6384012931967575 |
| 10 | 1.0 | 3.606530659712633 | 3.5987369392383783 | 3.6026876802191001 | 3.6055797344021654 | 3.6072423491217931 |

Table 2(b) Observed absolute errors for example 1.

| n | x_n | Errors | | | |
|----|-------|--------------------|--------------------|--------------------|--------------------|
| | | $h = 0.1$ | $h = 0.05$ | $h = 0.0125$ | $h = 0.003125$ |
| 0 | 0.0 | 0.000000000000000 | 0.000000000000000 | 0.000000000000000 | 0.000000000000000 |
| 1 | 0.1 | 0.0012294245007141 | 0.0006044245007142 | 0.0001492400965826 | 0.0000360894339484 |
| 2 | 0.2 | 0.0023374180359590 | 0.0011495274109592 | 0.0002839008697624 | 0.0000697646967454 |
| 3 | 0.3 | 0.0033329764250576 | 0.0016396753996673 | 0.0004050505152251 | 0.0001000680764069 |
| 4 | 0.4 | 0.0042245030779817 | 0.0020789494157198 | 0.0005136876602441 | 0.0001272484356862 |
| 5 | 0.5 | 0.0050198455714052 | 0.0024711622149671 | 0.0006107456112958 | 0.0001515383800963 |
| 6 | 0.6 | 0.0057263300567181 | 0.0028198748550667 | 0.0006970963526993 | 0.0001731552518756 |
| 7 | 0.7 | 0.0063507936249638 | 0.0031284122172535 | 0.0007735543104772 | 0.0001923020656545 |
| 8 | 0.8 | 0.0068996147465774 | 0.0033998776108142 | 0.0008408798948256 | 0.0002091683891745 |
| 9 | 0.9 | 0.0073787418971643 | 0.0036371665129238 | 0.0008997828338435 | 0.0002243604456358 |
| 10 | 1.0 | 0.0077937204742550 | 0.0038429794935330 | 0.0009509253104678 | 0.0007116894091599 |

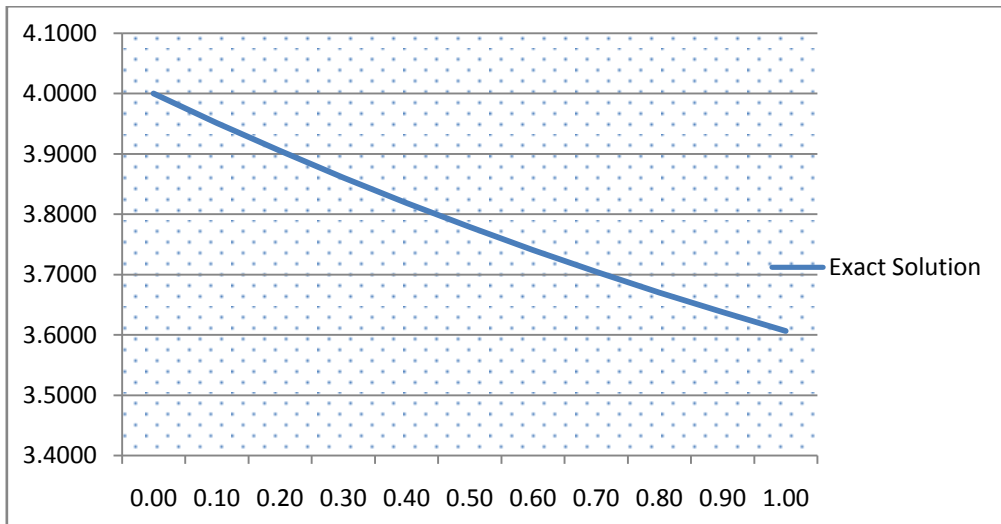


Figure 4: Exact Numerical Solution

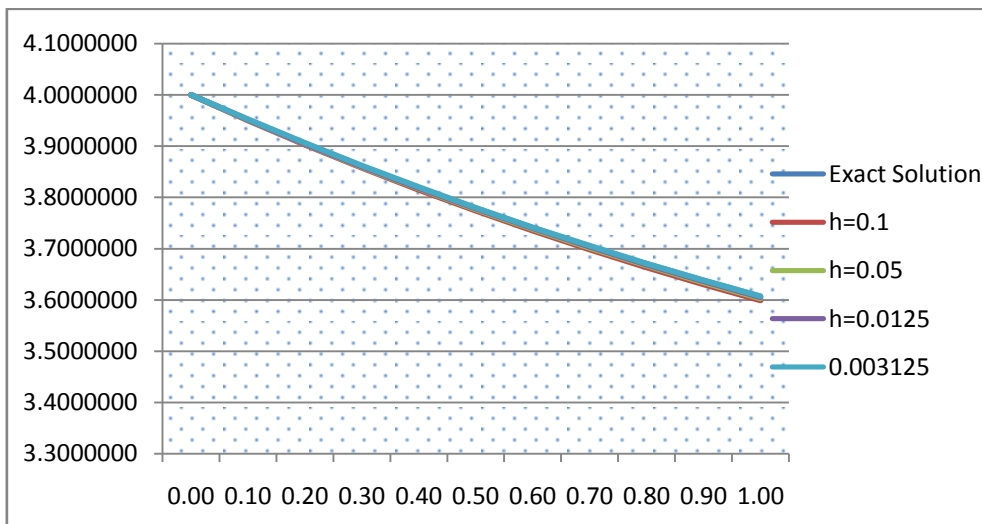


Figure 5: Exact Solution and Numerical Approximation for different step sizes

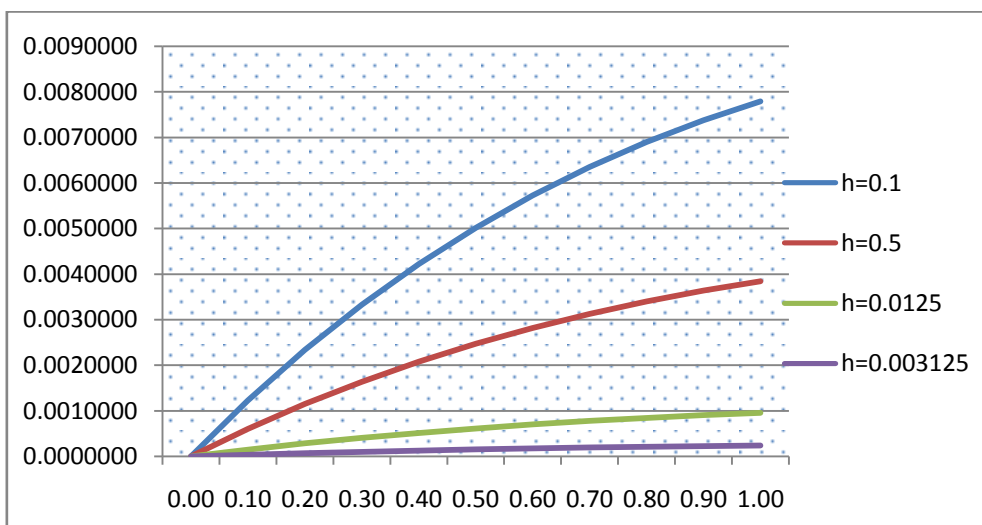


Figure 6: Error for different step sizes

V. DISCUSSION AND RESULTS

The obtained results are displayed in Table 1(a)-(b) and Table 2(a)-(b) and graphically represented in Figures (1-3) and Figures (4-6) respectively. The approximate solutions and absolute errors are calculated using MATLAB programming language with the step sizes 0.1, 0.05, 0.0125, and 0.003125 and also computed with the exact solution. From the tables, we observed that the proposed method give very good results when compared with the exact solutions. We equally observed that the error get large as the value of n increases. Hence, we may say that a numerical solution converges to the exact solution if reducing the step size leads to decreased errors such that in the limit when the step size reduce to zero the errors go to zero.

VI. CONCLUSION

In this paper, the Euler method has been presented for solving first order Ordinary Differential Equations (ODE) with initial conditions. To find more accurate results of the numerical solution, we reduced the step size to very, very small. From our tables and figures, we analyzed that the solution for the proposed method converges to the exact solution for decreasing the step size h. In the same vein, the numerical solutions obtained are in good agreement with the exact solutions and the numerical results of the two problems guarantee consistency, convergence, and stability. Thus, the accuracy increase with decrease step size, we may conclude that this method is applied to solve first-order ODEs with initial conditions to find the anticipated accuracy. In our subsequent research; we shall examine the comparison of the existing Euler methods and other methods like the Adomian decomposition.

REFERENCES

- [1]. D. J. Lambert, Computational Methods in Ordinary Differential Equations. New York: Wiley & Sons, 21-205IEA, 2000.
- [2]. J. C. Butcher, Numerical Methods for Ordinary Differential Equations. West Sussex: John Wiley & Sons Ltd., 2003, 45-95.
- [3]. K. Atkinson, W. Han, D. Stewart, Numerical Solution of Ordinary Differential Equations. New Jersey, John Wiley & Sons, Hoboken, 70-87.
- [4]. L. Euler, Institutiones Calculi Integralis Volumen Primum, Opera Omnia. Vol. XI, B. G. Teubner Lipsiae et Berolini MCMXIII, 1768, 21-228.
- [5]. L. Euler, De integration aequationum differentialium per approximationem, In Opera Omnia, 1st series, Vol II, Institutiones Calculi Integralis, Teubner, Leipzig and Berlin, 424434, 1913.
- [6]. K. E. Brenan, S. L. Campbell S, L. R. Petzold L, Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. New York: Society for Industrial and Applied Mathematics, 1989, 76-127.
- [7]. J. D. Lambert, Numerical Methods for Ordinary Differential Systems: The Initial value Problem. New York: John Wiley & Sons, 1999, 149-205.
- [8]. W. E. Boyce, R. DiPrima, Elementary Differential Equations and Boundary Value Problems. New York: John Wiley & Sons, Inc., 2000, 419-471.
- [9]. B. Carnahan, H. Luther, J. Wikes, Applied Numerical Methods, Florida: Krieger Publishing Company, 1990, 341-386.
- [10]. S. Chapra, R. Canale, Applied Numerical Methods with MATLAB for Engineers and Scientists, 6 Ed., Boston: McGraw Hill, 2006, 707-742.
- [11]. R. Esfandiari, Numerical Methods for Engineers and Scientists using MATLAB. New York: CRC Press (CRC), Taylor & Francis Group, 2013, 329-351.
- [12]. S. Fatunla, Numerical Methods for Initial Value Problems in Ordinary Differential equations. Boston: Academic Press, INC, 1988, 41-62.
- [13]. S. D. Conte, C. D. Boor, Elementary Numerical Analysis: Algorithmic Approach. New York: McGraw-Hill Book Company, 1980, 356-387.
- [14]. E. Kreyszig, Advanced Engineering Mathematics, 10 Eds., Boston: John Wiley & Sons, Inc, 2011, 921-937.
- [15]. A. Iserles, A First Course in the Numerical Analysis of Differential Equations, Cambridge: Cambridge University Press, 1996, 1-50.
- [16]. S. Fadugba, B. Ogunrinde, T. Okunlola, Euler's Method for Solving Initial Value Problems in Ordinary Differential Equations. The Pacific Journal of Science and Technology, Vol. 13 (2), 2012, 152-158.
- [17]. M. A. Islam, Accurate Analysis of Numerical Solutions of Initial Value Problems (IVP) for ordinary differential equations (ODE). IOSR Journal of Mathematics (IOSR-JM), Vol. 11 (3), 2015, 18-23.
- [18]. N. Jamali, Analysis and Comparative Study of Numerical Methods to Solve Ordinary Differential Equation with Initial Value Problem. International Journal of Advanced Research (IJAR). Vol. 7(5), 2019, 117-128: Available from: http://www.journalijar.com/uploads/536_IJAR-27303.pdf.

Anthony Anya Okeke" Accuracy Study on Numerical Solutions of Initial Value Problems (IVP) in Ordinary Differential Equations (ODE)" International Journal of Mathematics and Statistics Invention (IJMSI), vol. 07, no. 02, 2019, pp.40-47