1.

COMPARISON BETWEEN NEWTON'S INTERPOLATION AND LAGRANGE METHOD WITH RUNGE KUTT A METHOD IN SOLVING FIRST ORDER DIFFERENTIAL EQUATIONS

BY

MULIGI AMOS

16/A/BSCED/1450/F

A RESEARCH PROJECT SUBMITTED TO THE FACULTY OF EDUCATION IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELORS DEGREE OF SCIENCE WITH EDUCATION OF KABALE UNIVERSITY

AUGUST, 2019

DECLARATION

I **MULIGI AMOS,** declare that this project titled **"Comparison between Newton's interpolation and Lagrange method with Runge Kutta 2" Order method"** has never been submitted to any university or institution of higher learning for any academic qualifications.

Signature: ...~

Date02/08/2019

MULIGI AMOS

(RESEARCHER)

APPROVAL

This research project titled **"Comparison between Newton's interpolation and Lagrange method with Runge Kutta 2" order method"** has been done under my supervision

Signature: ....~ .... Date: 09/08/19 .

DR. MBABAZI DOREEN SSEBULIBA

(UNIVERSITY SUPERVISOR)

4.

DEDICATION

I dedicate this piece of work to my parents Mr. James Wandera and Mrs. Amongin Mary Wandera, brothers and sisters as I do not know what my world would be without you. I cherish you all.

ACKNOWLEDGEMENT

To begin with, I thank the Almighty God for the breath that I have to this minute, the wisdom and the strength he has given me and helped to complete this piece of work. I thank you Lord so much.

I feel so much indebted to the following people without whose moral, academic materials, financial support and positive criticisms this research project would not have come to its completion.

My gratitude goes to **Dr. Mbabazi Doreen Ssebuliba** the research supervisor for the time he gave me for academic and professional guidance and advice in organizing this work into its present state. May the Almighty reward him abundantly; gratitude is also extended to my dear parents for their financial support towards this work.

Thanks go to my family especially my brothers Mr. Taulya Julius, Mr. Jaayo Joel and Kalugana Emmanuel, Sisters, Ms, Akeraut Hellen, Ms, Logose Naume, and Sr. Violet Gimbo, relatives and friends for the support and comfort I received which enabled me to accomplish this project with less difficulty.

I will forever live to remember your efforts.

ABSTRACT

The main aim of the study was to make a comparison between a combination of Newton's interpolation and langrage method with Runge Kutta second order method. The study was guided by objectives like to identify a simpler method of solving first order differential equations given a combination of Newton's interpolation and Lagrange and Runge-Kutta and to identify a method of solving first order equations with increased accuracy given a combination of Newton's interpolation and Lagrange and Runge- Kutta. The method used to solve first order differential equation were Newton's interpolation and Lagrange methods with Runge Kutta and Analytic method which gave the exact values of the solved first order differential equations. The result showed that a combination of Newton's interpolation and Lagrange method gives a small absolute error and therefore efficient method for solving first orders differential equations. Numerical method used (Runge Kutta) is cumbersome and has a bigger percentage error. I therefore recommend the use of combined Newton's interpolation and Lagrange method to solve first Order Differential equations.

TABLE OF CONTENTS

4

CHAPTER ONE

INTRODUCTION

1.0 Introduction

This chapter presents the background of the study, statement of the problem, purpose of the study, objectives of the study and significance of the study.

1.1 Background of the study

Many problems in real life situation can be formulated in the form of ordinary differential equation, hence the need to solve the differential equations. A numerical method is a tool designed to solve numerical problems. A differential equation as for example $u'(x) = \text{Cos}(x)$ for $0<x<3$ is written as an equation involving some derivative of an unknown function u (Weisstein, 2004). There is also a domain of the differential equation (for the example $0 <x< 3$). In reality, a

differential equation is then an infinite number of equations, one for each x in the domain. The analytic or exact solution is the functional expression of u or for the example case $u(x) = \sin(x) + c$ where c is an arbitrary constant, because of this non uniqueness which is inherent in differential equations we typically include some additional equations. For our example case, an appropriate additional equation would be $u(l) = 2$ which would allow us to determine c to be $2- \sin(l)$ and hence recover the unique analytical solution $u(x) = \sin(x)+2 - \sin(l)$.

The differential equation together with the additional equation (s) are denoted a differential equation problem. Note that for our example, if the value of u (1) is changed slightly, for example from 2 to 1.95 then also the values of u are only changing slightly in the entire domain. This is an example of the continuous dependence on data that we shall require: A well-posed differential equation problem consists of at least one differential equation and at least one additional equation such that the system together have one and only one solution (existence and uniqueness) called the analytic or exact solution (Joshn, 1999) to distinguish it from the approximate numerical solutions that we shall consider. Further, this analytic solution must depend continuously on the data in the (vague) sense that if the equations are changed slightly then also the solution does not change too much.

Numerical analysis is the area of mathematics and computer science that creates, analyzes and implements numerical method for solving numerically the problems of continuous mathematics. Such problems originates from real-world applications of algebra, geometry and calculus and they involve variables that vary continuously, such problems occur throughout the natural sciences, social sciences, engineering, medicine and business.

In almost any discussion of interpolation formula is a certain collection, the interpolation formulas are derived which find the interpolated value of a function in terms of certain of its values.

These standard formulas are all expressions for the polynomial, which gives the function at certain values. Polynomial interpolation methods include Newton's divided difference and Lagrange's interpolation formulas. These formulas involve finding a polynomial of order n-1 that passes through the n data points.

Ordinary differential equations in science and engineering: in geometry and mechanics from the first examples onwards (Newton, Leibniz, Euler, Lagrange), in chemical reaction kinetics; molecular dynamics, electronic circuits, population dynamics, and many more application areas. The study in this regard wishes to determine the comparison between Newton's interpolation and Lagrange method.

1.2. Statement of the problem

The need to solve first order differential equations using numerical approaches. Most of the researchers on numerical approach to the solution of ordinary differential equation tend to adopt other methods such as Runge-Kutta method and Euler's method but none of the study has actually combined Newton's interpolation and Lagrange method to solve first order equation. Differential equation is one of the major areas in mathematics with series of method and solutions. The analytic method and numerical methods; analytic method is only applicable to a class of equations, so most of the times numerical methods are used. This study combined Newton's interpolation and Lagrange method to solve the problems of first order differential equation and compared the results with the results obtained when solving the same differential equation with Runge- Kutta second order method.

1.3. Objectives of the study

1.3.1 General objective

The main aim of the study was to make a comparison between a combination of Newton's interpolation and Lagrage method with Runge Kutta second order method.

1.3.2 Specific objectives of the study

i. To identify a simpler method of solving first order differential equations given a combination of Newton's interpolation and Lagrange and Runge-Kutta.

j. To identify a method of solving first order equations with increased accuracy given a combination of Newton's interpolation and Lagrange and Runge- Kutta.

1.4 Research questions

The study came up with research questions so as to ascertain the above stated objectives of the study. The research questions for the study are stated below as follows:

5. What is the simpler method of solving first order differential equations given a combination of Newton's interpolation and Lagrange and Runge Kutta?

ii. What is the method of solving first order equations with increased accuracy?

1.5 Scope of the study

The study was limited to first order differential equation using numerical Newton's interpolation and Lagrange and Runge Kutta second order methods. The study covered numerical solution to first order differential equation using Newton's interpolation and Lagrange and Runge- Kutta

1.6 Significance of the study

The study on comparison between Newton's interpolation and Lagrange method and Runge-Kutta method will be of immense benefit to the entire mathematics departments in Uganda because it will formulate best ways of solving first order differential equations

Finally the study after completion will contribute to the existing literature on numerical solution to first order differential equations using Newton's interpolation and Lagrange methods

1.7 Definition of Key Terms

First order differential equation: A first-order differential equation is an equation

$y' = f(x, y)$ in which $f(6, y)$ is a function of two variables defined on a region in the xy-plane. The equation is of first order because it involves only the first derivative $\frac{2}{}$ and not higher-order dx derivatives).

Interpolation: Is a method of constructing new data points within the range of a discrete set of known data points.

1.8 Limitation of the Study

Financial constraint- Insufficient fund tended to impede the efficiency of the researcher in sourcing for the relevant materials, literature or information and in the process of data collection (internet).

Time constraint- The researcher simultaneously engaged in this study with other academic work. This consequently cut down on the time devoted for the research work.

1
)

CHAPTER TWO

LITERATURE REVIEW

One of the standard topics in numerical analysis courses is the subject of interpolation with particular emphasis on the Lagrange and Newton Interpolating formulas. In both cases, the usual approach is highly computational as one works to construct the polynomial of degree n that passes through $n + 1$ interpolating points. However, as Richard Hamming, one of the giants of modern numerical analysis put it, "The purpose of computing is insight, not numbers". In this article, we will look graphically at the functional components of each of these two interpolating formulas to see the kinds of deeper insights that can be achieved.

The Lagrange Interpolating Formula Suppose that we have the n+ l points(a,y), (O,y) ..., *(O.y,),* where all of the *X,* are different, though not necessarily uniformly spaced. These interpolating points then determine a unique polynomial of degree *n* (or possibly lower, if the points happen to lie on such a curve). One way to express the equation of this polynomial is witll the Lagrange Interpolating Formula:

$$L_n(x) = \sum_{i=0}^{n} y_i \prod_{k=0,k\neq i}^{n} \frac{(x-x_k)}{(x_i-x_k)} = \sum_{i=0}^{n} y_i \frac{(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}.$$

Equivalently, if we write this formula without the summation notation, it becomes

$$L_n(x) = y_0 \frac{(x-x_1)(x-x_2)\cdots(x-x_n)}{(x_0-x_1)(x_0-x_2)\cdots(x_0-x_n)} + y_1 \frac{(x-x_0)(x-x_2)\cdots(x-x_n)}{(x_1-x_0)(x_1-x_2)\cdots(x_1-x_n)} + \cdots + y_n \frac{(x-x_0)(x-x_1)\cdots(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)\cdots(x_n-x_{n-1})}.$$

Notice that this polynomial is composed of n + l distinct polynomial terms, each of degree n (provided *Y;* * 0). Either way, both of these are rather daunting expressions for students and consequently it is not surprising that many tend to miss some of the key underlying concepts (Mineola, 1994)

The Newton Interpolating Formula

Again, suppose that we have the $n+$ lpoints (x_0,y_0), (x_1,y_1), ••. , (xn,Yn), where all of the xi are different. For simplicity, we consider the case where these xi's are uniformly spaced with !).x = xk+l - xk for each k. These interpolating points determine a unique polynomial of degree n (or possibly lower, if the points happen to lie on such a curve). Another way to express the equation of this polynomial is with the Newton Forward Interpolating Formula:

P (x) = y +--o (x-x)+ " Ay Ax 2!Ax)° Ay 0 (x-x)(x-x)+ Yo 3 A (x-x0)(x-x1)(x-x2)+ 3!(A£)

... + A"y 0 n!(Ax)" (x-x)(x-x)··•(x-x) 1 n-1 ,

Where

A = »

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0 = y_2 - 2y_1 + y_0 ,$$

A3 Jo = ~ y- 2 = Y3 - 2 Y2 + 3 Yi - 3 Yo,

A"y 0 = N" 'y-A" 1 Yo

=yn-nYn-1+ n(n-l) 2! Yn-2- n(n-l)(n-2) 3! Yn-3+ .. ·+(-l) n Yo·

According to Gear (1971) noticed that this polynomial P,(a) is also composed of n + l distinct polynomial terms, but each of degree$i$, i= 0, l, 2, ... , n . Term by term, each polynomial is of one degree higher than the previous one. At a quick glance, the above formula is obviously very similar to the formula for the n^{th} degree Taylor polynomial approximation for a function

$y = f(x)$ at $x = x_0$: T(x)=f(x }+f'(x)(x-x)+f''(xo)(x-x)2+/'''(xo)(x-x)3+ n 0 O O 2! 0 3! 0

.. · + f (n)(x) n! 0(x - x0t.

Comparisons between Lagrange and Newton Interpolation

The Lagrange and Newton interpolating formulas provide two different forms for an interpolating polynomial, even though the interpolating polynomial is unique. When we want a **quick symbolic expression of the interpolating polynomial, the Lagrange formula seems to be the** way to go. For this reason, the Lagrange form is most often used for deriving formulas for **approximating derivatives and integrals. For example, many numerical analysis textbooks (for** example, and establish the trapezoidal rule and Simpson's rule by using the Lagrange formula for linear and quadratic interpolating polynomials to approximate the integrand, respectively. However, the Newton formula is much better for computation than the Lagrange formula.

When using the interpolating polynomials for working with functions that are stored in tabular form, we often choose the Newton formula. As we will show below, the forward differences *!)/ Yo* that determine the coefficients of the Newton formula can be easily constructed using a tabular form. More importantly, the Newton formula provides a generally accurate idea of when the degree *n* is sufficiently large by observing the size of the terms with higher-order forward differences. This is a useful technique in deciding what degree polynomial to use.

Suppose we are given five points(2.0, 1.414214),(2.1, 1.449138), (2.2, 1.483240), (2.3, 1.516575), and (2.4, 1.549193), which is based on an example below. We construct the forward difference table for these five points, shown in Table 1.

Table 1 Forward difference table for the five points

l	*X,*	*y*	*Ay,*	*Ny*	A'y,	*sty*
0	2.0	1.414214	0.034924	-- 0.000822	0.000055	- 0.000005
1	2.1	1.449138	0.034102	- 0.000767	0.000050	
2	2.2	1.483240	0.033335	- 0.000717		
',	2.3	1.516575	0.032618			
4	2.4	1.549193				

The last five entries in the first row are used to determine the coefficients of the Newton interpolating polynomial

P,() = 1.414214 + 0.34924(x -2.0).-0.04 l(x - 2.0)(x - 2.1)
+ 0.009167(x-2.0)(x- 2. l)(x-2.2)-0.002083(x-2.0)(x-2.l)(x-2.2)(x-2.3).

By applying the usual optimization approach from Calculus I to the fourth degree polynomial term

-0.002083(x-2.0)(x-2.1)(x-2.2)(x-2.3) on the interval [2.0,2.4], we find that *Max* \\(*x*-

2.0)(x-2. l)(x-2.2)(x-2.3)\\ = 0.0024.
2.0<xr<2.4

Then the largest possible value the last term of $P_4(x)$ that will contribute to the interpolating polynomial at any point in the interval [2.0,2.4] is roughly 0.002083x0.0024:::, 5xl0-^{6}.

This result may be improved on by following a common practice used in approximating functions with the Newton formula. When we want to approximate the function at a point *x* that is inside the first half of the interval, we use the above Newton forward formula. Otherwise we use the Newton backward formula, or equivalently, we apply the Newton forward formula to the same table where the entries are listed in reverse order. If we stay with the original notation for the interpolating points *(xi, Y;)* for *i* = 0, 1, 2,....*n* . We define the backward differences as

follows. Let Vy,=y-y,, Vy, = Vy,)= Vy,-Vy,+=-2y1+y,, and in general. Vy, = VV" 'y,)=V" 'y-V'y,, for *i*>1. Then the Newton Backward Interpolating Formula can be expressed as

,-y,/"2-,_",s-sos-+,_."_,s-se-.a-.s)
!:ix 2!(/:ix) 3!(!:ix)

..., V"y.-x,)a-Mp)·(r-3,).
n!(!:ixt n n I

In contrast, the Lagrange interpolation approach requires far more computation - each time you increase the number of interpolating points by one, you have to recalculate everything (Hairer, 1993). This makes Lagrange interpolation less convenient for seeking the lowest degree interpolating polynomial that fits the data with a given error tolerance.

Runge-Kutta second order process

Given $y \ = f,y)$ with initial conditions $y(xo)yo.$

Runge-Kutta methods are popular for solving ordinary differential equations. The Runge-Kutta methods have advantages over Euler and Taylor series.

Euler method is less efficient in practical problems since it requires h to be small in order to obtain reasonable accuracy.

The Runge-Kutta methods are designed to give greater accuracy and they require only the functional values at selected points in the sub interval (Cuneyt, 2004).

The Runge-Kutta methods of second order are given by:

$$\mathbf{y+1} = y+;\overset{1}{\mathbf{k}} + k)$$

Where: k, = $hf\ a4,3)$

$$k_2 = hf(x_n + h, y_n + k_1)$$

CHAPTER THREE

METHODS

The study compared both Newton's interpolation method and Lagrange method to solve first order differential equation. Since the problem is an initial value problem (IVP), the first value for y is available. The researcher used Newton's interpolation to find the second two terms then use the three values for y to form a quadratic equation using Lagrange method as follows; **Newton's interpolation method**

f(x)=ao +a1 (x-xo)+a2(X-Xo)(x-x1)+ ... +an(X-Xo)(x-x1) .....(X-X-1)

Where

$a_0 = y_0$

$$a_1 = \frac{f(x_1) - f(x_0)}{(x_1 - x_0)}$$

f(Xz)-f(x1) f(x1)-f(xo)

(a-x) (a+-x0)

(Xz - Xo)

Etc

Lagrange method

(X-X1)(X-Xz) + (X-Xo)(X-Xz) + (X-Xo)(X-X1)

y»Ga--«J'o Gr-&-'1 Ge-ca-x,°

y = a

y= a+a,(a- x6)

y= a+a,a-x) + a3a- x)(a- 1,)

Example 1

Solve dy = 1-y for y(0) = 0 taking step h=0.001 dx

Using Newton's interpolation a,=O=

Yo .,-IO-10_[]

(x1 - x0) dx 0,0

$y = 0 + 1(0.01 - 0) = 0.01$

$$a_1 = \frac{\frac{\text{rd»a re»-ra [2]}}{(X_2 - X_1)} - \frac{[2;]}{(x_1 - X_o)} \quad . \quad \frac{}{dx\ 0.01,0.01} \quad \frac{}{dx\ 0,0}}{(0-16) \qquad 0.02-0} = -0.5$$

$y = 0 + 1(0.02 - 0) - 0.50.02 - 0)(0.02 - 0.01) = 0.0199$

Forming quadratic using Lagrange

$$n = \frac{(x-0.01)(x-0.02)}{(0-0.01)(0-0.02)} * 0 + \frac{(x-0)(x-0.02)}{(0.01-0)(0.01-0.02)} * 0.01 + \frac{(x-0)(x-0.01)}{(0.02-0)(0.02-0.01)} * 0.01\ 99$$

$Y_n = -0.Sx^2 + 1.00Sx$

The equation is used to get the values for y at any given value of x

The same differential equation $\frac{dy}{dx} = 1 - y$ $\quad y(0) = 0$ h = 0.001 can be solved usmg

Runge- kutta second order as follows:

From: $Y_{n+1} = Y_n + \frac{1}{2}(k_1 + k_2)$

$k_1 = hf(x_n, Y_n)$

$k = hf(a + h, y + k_1)$

Now $k_1 = hf(x_o, Y_o)$

$k_1 = 0.01f(0,0)$

$k_1 = 0.01(1) = 0.01$

$k_2 = hf(x_o + h, Y_o + k_1)$ $k_2 =$

$0.01f(0.01, 0.01)$

$k = 0.01(0.99) = 0.0099$ 1

$y = y_o + 5k + k)$

$= 0 + \frac{1}{2}(0.01 + 0.0099)$

$= 0.00995$

1
y=y+7+k2)

k, = hf6,,y)

k, = 0.0lf(0.01,0.01)

k_1 = 0.01(0.99) = 0.0099

$$k_2 = hf(x_1 + h, y_1 + k_1)$$

k, = 0.0lf(0.01 + 0.01,0.01 + 0.0099) k_2 =

O.Olf (0.02,0.0199)

1 = 0.01(0.9801) = 0.009801 1

y= 0.00995 + 2 (0.0099 + 0.009801) = 0.0198005

Example 2

Solve the initial value problem y = y - x y(O) = 0.5 Taking step h=0.01

a,= 0.5

Y_0= 0.5

,-*l*OO-*10»* [%] -OS

(a-*x)* .*dx* 0,0.S

Yi = 0.5 + 0.5(0.01 - 0) = 0.505

$$a_2 = \frac{\frac{f(x_2)-f(x_1)}{(x_2-x_1)} - \frac{f(x_1)-f(x_0)}{(x_1-x_0)}}{(x_2 - x_0)} = \frac{\left[\frac{dy}{dx}\right]_{0.01,0.505} - \left[\frac{dy}{dx}\right]_{0,505}}{0.02 - 0} = -0.25$$

y= 0.5 + 0.5(0.02 - 0) - 0.25(0.02 - 0)(0.02 - 0.01) = 0.50995

Forming quadratic using Lagrange

= (x-0.0l)(x-0.02)* 0_5 + (x-0)(x-0.02)+ 0.505 + (x-O)(x-0.01)* 0_50995

)n (0-0.01)(0-0.02) (0.01-0)(0.01-0.02) (0.02-0)(0.02-0.01)

$Y_n = -0.25x^2 + 0.5025x + 0.5$

The equation is used to get the values for *y* at any given value of *x*.

The same differential equation can be solved using Runge- kutta second order as follows:

From:)yn41 $\quad Y_n + \frac{1}{2}(k1 + k2)$

$k, = hf\,(X_n, Y_n)$

$k, = hf(a, + h,3y + k,)$

Now $k, = hf(xo,Yo)$

$k, = 0.0lf\,(0,0.5)$

$k, = 0.01(0.5) = 0.005$ $k, =$

$hfa + h,y + k,)$ $k_2 = 0.0lf$

$(0.01,0.015)$

$k = 0.01(0.005) = 0.00005$ 1

$y = yo + 5 + k)$

$= 0.5 + \frac{1}{2}(0.005 + 0.00005)$

$= 0.502525$

$y = Y1 + \frac{1}{2}(k1 + kz)$

$k, = hf,y)$

$k, = 0.0lf(0.01,0.502525)$

$k, = 0.01(0.492525) = 0.00492525$ $k., =$

$hf(a + h,y + k)$

$k = 0.0lf(0.01 + 0.01,0.502525 + 0.00492525)$ $k_2 =$

$0.0lf(0.02,0.505745025)$

$k, = 0.01(0.48745025) = 0.0048745025$ 13

$$y = 0.502525 + \frac{1}{2}(0.0492525 + 0.0048745025) = 0.529588501$$

CHAPTER FOUR
RESULTS AND DISCUSSION

The two examples were solved analytically as:

Example 1,

$$\frac{dy}{dx} -=1-y$$

$$\frac{dy}{dx} -+y= 1$$

$$\frac{d}{dx}(y + yx) = 1$$

h#vs –fa y + yx = X + C y(0) =

0, c = 0

$$y54+x$$ x

$$y(0.01) = \frac{0.01}{1 + 0.01} = 0.00990099$$

The example2 can be solve analytically as

dy
V-X dx
"+
Of the for - pc)y= 40)

dy
.-y=-x IF/p@)e/de
dx

⇒ e -x dy .-. -- e y = -xe .-x
dx

d
(ey)=-xe
dx

•J#e-[-+eat

$> y = -\int xe^{-x}\, dx,$ $\Rightarrow y = -e^{x}\int -xe^{-x}dx$

L

et $u = x$, $\frac{d}{dx}1$ $\frac{dV}{dx} = e^{-x}$, $\Rightarrow V = -e^{-x}$

$y = -er[-e +]e.1as]$

$$= x - e^{x}\int e^{-x}dx$$

$$= x - e^{x}(-e^{-x}) - ce^{x}$$

$= x + 1 - ce$

Considering initial conditions $y(00) = 0.5$ $0.5 = 0 + 1 - ce$ $> C = 0.5$

$\therefore y = x + 1 - 0.5e$

The equation is used to get the values for *y* at any given value of *x*

The method that has been used gives results very close to the exact value. This is noted by the percentage error that is very minor. The method is very accurate and easy to use after forming the quadratic equation. Thus one can get the value of *y* at any value of ***x*** without necessary getting preceding values of *y.*

80.

Table 2. The table showing results of the equation :: = 1 - *y*

X	Combined Newton's interpolation .an.d La~range	Exact values	Error
0	O	0	Q
0.01	0.0100	0.00990099	0.000099011
0.02	0.0199	0.019607843	[0.000292157\
0.03	0.0297	0.029126213	I0,000573787
0.04	0.0394	0.038461538	I0,000938462
0.05	0.0490	0.047619047	10.004238095
0.06	0.0585	0.056603773	10.0018962211
0.07	0.0679	0.06542056	10.00247944
0.08	0.0772	0.074074074	j0.003125926
0.09	0.0864	0.082568807	10.003831193
0.1	0.0955	0.09090909	[0.00459091

Table 3. The table showing results of the equation $y' = y - x$

x	Combined Newton's interpolation and Lagrange	Exact values	error
0	0.5	0.5	0
0.01	0.50500	0.50~97~9,16	10.0000841
0.02	0.50995	0.509999330	[0.000043/
0.03	0.51485	0.514772733	10.0000772671
0.04	0.51970	0.519594612	[0.0000105388] .
0.05	0.52450	9-5~~~6~,:1?1 ,	19:0091355491
0.06	0.529 25	0.529081726	[0.000168274/
0.07	0.53395	0.533745909	10,0002040911
0.08	0.53860	0.538356466	[0.000243534/
0.09	0.54320	0.542912858	10,0002871421
0.1	0.54775	0.547414541	[0.000335459/

According to the results for instance for ***y(0.01)*** in examplel a combination of Newton's interpolation and Lagrange gives an absolute error of ***0.00009901*** as seen in table 2. Runge Kutta second order gives an absolute error of ***0.00989951.***

For ***y(0.01)*** in example 2 a combination of Newton's interpolation and Lagrange gives an absolute error of ***0.000084*** as compared to ***0.024613585*** absolute error obtained by Runge Kutta

second order

CHAPTER FIVE Conclusion and Recommendation

Numerical method used (Runge Kutta) is cumbersome and has a bigger percentage error. I therefore recommend the use of combined Newton's interpolation and Lagrange method to solve First Order Differential equations

Therefore the combination of Newton's interpolation and Lagrange gives a high accuracy as compared to Runge Kutta second order.

A combination of Newton's interpolation and Lagrange is a simpler method to use since it has less iteration as compared to Runge Kutta second order.

REFERENCES

Davis P.J., *Interpolation and approximation,* Second edition, Dover, New York, NY, 1975 Deuflhard, P. and Bornemann, F. (2002) Scientific computing with ordinary differential equations , Springer, New York.

Dover, Mineola, NY, (1994) *Introduction to numerical analysis,* Second edition,

Dr. CuneytSert (2004) *Numerical Methods Interpolation Mechanical Engineering Department Middle East Technical University Ankara,* Turkey Isaacson E., Keller H.B., Analysis of numerical methods, Second edition,

Faith Chelimo Kosgei, (2018). «solution of first order differential equation using numerical Newton's interpolation and Lagrange method", *International Journal of Development Research, 8, (02),* 18973-18976.

Gear, C.W. (1971*) Numerical initial value problems in ordinary differential equations,* PrenticeHall, Englewood Cliffs, NJ.

Hairer, E., Norset, S.P. and ·wanner, G. (1993) *Solving ordinary differential equations.* I: Nonstiff problems. 2nd revised ed., Springer, Berlin.

International Joumal of Development Research, Vol. 08, Issue, 02, pp.18973-18976, February, (2018)

Springer Verlag. New **York, NY,** (1993), *An introduction to numerical analysis,* Cambridge University press.Cambridge, UK, 2003