EULER'S MODIFIED AND RUNGE-KUTTA METHODS TO SOLVING SYSTEMS OF FIRST ORDER DIFFERENTIAL EQUATIONS

BY

JOSEPH, EBENEZER DANIEL

MATRICULATION NUMBER: 20183036

A PROJECT SUBMITTED TO THE

DEPARTMENT OF MATHEMATICS,

COLLEGE OF PHYSICAL SCIENCES,

FEDERAL UNIVERSITY OF AGRICULTURE, ABEOKUTA.

IN PARTIAL FULFILMENT FOR THE AWARD OF BACHELOR OF SCIENCE DEGREE IN MATHEMATICS.

NOVEMBER, 2023.

DECLARATION

I **JOSEPH, EBENEZER DANIEL** hereby declare that the final year project titled: "**Euler's Modified and Runge-Kutta Methods to Solving Systems of First Order Differential Equations**" submitted by me, in partial fulfillment of the requirements for the award of Bachelor of Science Degree in Mathematics, Federal University of Agriculture, Abeokuta. I declare that all external sources and references used in this project have been properly acknowledged and cited.

JOSEPH, EBENEZER DANIEL

Date:..................................

CERTIFICATION

This is to certify that the work titled: " **Euler's Modified and Runge-Kutta Methods to Solving Systems of First Order Differential Equations** " was carried out by **JOSEPH, EBENEZER DANIEL** with **Matriculation Number: 20183036,** a student of the Department of Mathematics, College of Physical Sciences, Federal University of Agriculture, Abeokuta, Ogun State, Nigeria.

....

PROF. M. O. OMEIKE ***DATE***

SUPERVISOR

....

DR. E. O. ADELEKE ***DATE***

Ag. **HEAD OF DEPARTMENT**

DEDICATION

I dedicate this project to Almighty God, the alpha and omega, the creator of the universe and all mankind, who made the completion of this work a reality. I dedicate this work to my amazing grandparents, Late Pa Kasimawo Joseph and Mrs Ruth Joseph, down to my beloved parents (Mr Ngam Gift and Mrs Ngam Foluke Joseph) as well as my ever-supportive siblings and to everyone that has been supportive and helpful in my education life.

ACKNOWLEDGMENT

I adore and glorify the Merciful God, who has made the success of the research work and the completion of my BSc. programme at large a reality.

I am grateful to my supervisor, Professor M. O. OMEIKE for his utmost support and encouragement throughout the entire process of completing this project.

I am grateful to the Acting Head of Department, Dr. E. O. Adeleke, immediate past Head of department Prof. B.I. Olajuwon and all lecturers of the Department of Mathematics.

My sincere appreciation also goes to my parent, Mr. Ngam Gift and Mrs Ngam Foluke Joseph, for their full support, advice, prayer, love and care placed on me throughout this project period and my stay on campus at large.

I also appreciate my wonderful siblings: Joseph Ezekiel, Ebunola Shittu, Raphael Emmanuel, Raphael Victor, and Ngam Princess, for their contribution physically, spiritually, financially towards the success of my programme.

My active Head of Class, Lawal Ibrahim, must not be left behind. Thank you for being an amazing representative.

Finally, my sincere gratitude also goes to all those who have contributed to my success in FUNAAB: my friends (Olaiya Taiwo, Ajibade Tunde, Opadijo Taiwo, Sowemimo Oluwatomisin, and Adeniran Tayo), departmental mates (Adebisi Adewunmi, Oluwaloseyifunmi, Lawal Ibrahim, Bello Sekinah and Olalere Babatunde) and many others that i couldn`t mention their names. Thank you all and God bless you. (AMEN).

ABSTRACT

Solving first-order ordinary differential equations (ODEs) is a fundamental challenge in computational mathematics, with applications in various scientific and engineering domains. This project explores the efficacy and applicability of two numerical methods, namely the Euler Modified (Improved Euler or Heun's) method and the Runge-Kutta methods, in the context of first-order ODE solutions. The Euler Modified method, an improvement over the basic Euler method, demonstrates commendable accuracy and computational efficiency, making it a practical choice for problems with smooth solutions. However, it faces limitations in tackling stiff ODEs characterized by rapid transitions. In contrast, the family of Runge-Kutta methods, offers exceptional accuracy and stability across a broad spectrum of ODE types, including stiff systems. This work enhances our understanding of the Euler Modified and Runge-Kutta methods, offering practical guidance for numerical ODE solving.

Table of Contents

Contents | **Page**

Contents	Page
Title page	i
Declaration	ii
Certification	iii
Dedication	iv
Acknowledgment	v
Abstract	vi
Table of Contents	vii
CHAPTER ONE: Introduction	1
1.1 Background of the study	1
1.2 Motivation	2
1.3 Objectives	3
1.4 Preliminaries and Definitions of Terms	3
CHAPTER TWO: Literature Review	5
CHAPTER THREE: Methodology	8
3.1 Euler's Modified (Heun's) Method	8

3.2 Runge-Kutta Method 10

CHAPTER FOUR: Applications 13

4.1 Illustrative Examples 13

4.1.1 Example on Euler's Modified Method 13

4.1.2 Additional Example on Euler's Modified Method 18

4.1.3 Example on Runge-Kutta Method 21

4.1.4 Additional Example on Runge-Kutta Method 26

CHAPTER FIVE: Conclusion and Recommendations 29

5.1 Conclusion 29

5.2 Recommendations 30

REFERENCES 31

CHAPTER ONE

1.0 INTRODUCTION

1.1 Background to the Study

Ordinary Differential Equations (ODEs) are mathematical tools that describe how a function changes with respect to its independent variable. They find extensive applications in numerous scientific and engineering disciplines, ranging from physics and chemistry to biology and economics. One fundamental task in ODE analysis is to find numerical solutions, especially for complex systems where exact analytical solutions seem impossible.

Picture a world where predicting future events and understanding dynamic processes is essential. Ordinary Differential Equations (ODEs) serve as a bridge between the past and the future, offering a mathematical language to describe how things change over time. From modeling population dynamics to simulating chemical reactions, ODEs are omnipresent in science and engineering.

The world is full of circumstances governed by the laws of change, where the future state depends on the current conditions. These dynamic systems can often be described using Ordinary Differential Equations (ODEs), making ODEs a cornerstone of mathematical modeling in various scientific domains.

In this project, we delve into the world of numerical methods, focusing on the Euler Modified and Runge-Kutta techniques, which provide effective means of approximating solutions to first-order ODEs. These methods are essential tools for approximating the solutions of first-order ODEs, allowing us to gain valuable insights into dynamic processes and make predictions with precision.

1.2 Motivation

The motivation behind this research stems from the critical role that numerical methods play in addressing complex real-world problems across various scientific and engineering disciplines. Ordinary Differential Equations (ODEs) serve as fundamental models for dynamic systems, allowing us to describe the behavior of physical, biological, and engineering systems over time. Consequently, numerical methods become indispensable tools for approximating solutions, making informed predictions, and gaining insights into dynamic systems.

In addition, the motivation for this research lies in the pivotal role of numerical methods in ODE solving, the challenges they pose, and the need to better understand and utilize Euler Modified and Runge-Kutta methods.

1.3 Objectives

- ✓ To Investigate the Accuracy of Euler Modified and Runge-Kutta Methods
- ✓ To Examine the Stability of Euler Modified and Runge-Kutta Methods
- ✓ To Enhance Understanding of Numerical ODE Solving

1.4 Preliminaries and Definitions of Terms

- ✓ **Ordinary Differential Equations** (ODEs): Ordinary Differential Equations are mathematical equations that involve derivatives of an unknown function with respect to a single independent variable. They are used to model how a quantity changes concerning time or another independent variable.
- ✓ **Initial Value Problem** (IVP): An Initial Value Problem is a specific type of ODE where you are given an equation along with an initial condition. The goal is to find the solution that satisfies both the equation and the initial condition.
- ✓ **Numerical Methods**: Numerical methods are techniques used to approximate solutions to mathematical problems when exact analytical solutions are either difficult or impossible to obtain. In the context of ODEs, numerical methods are used to approximate the solutions of differential equations.
- ✓ **Step Size**: In numerical methods for solving ODEs, the step size (or time step) is the interval at which the solution is approximated. Smaller step sizes generally lead to more accurate results but can increase computational cost.

- ✓ **Euler**'s **Modified Method**: The Euler Modified method, also known as the Improved Euler method or Heun's method, is a numerical technique for solving ODEs. It is an enhancement of the basic Euler method and provides more accurate approximations by considering both the slope at the current point and the slope at an intermediate point within a given step size.
- ✓ **Runge-Kutta Method**: The Runge-Kutta methods are a family of numerical techniques used for solving ODEs. They are based on a weighted average of slopes at various points within a step and are known for their accuracy and stability. The most common types are the second-order (RK2) and fourth-order (RK4) Runge-Kutta methods.
- ✓ **First-Order ODE**: A first-order ordinary differential equation is an ODE where the highest derivative of the unknown function is the first derivative. Mathematically, it is represented as $\frac{dy}{dx} = f(x, y)$, where y is the unknown function, x is the independent variable, and f(x, y) is a given function.

CHAPTER TWO

2.0 LITERATURE REVIEW

Introduction

Ordinary Differential Equations (ODEs) are foundational tools in various scientific and engineering disciplines, enabling the modeling of dynamic systems and predicting their behavior over time. When exact analytical solutions are difficult, numerical methods come to the forefront as indispensable tools for approximating solutions to ODEs. This literature review aims to explore the historical development, applications, and comparative analysis of two widely used numerical techniques: the Euler's Modified method and Runge-Kutta methods for solving first-order ODEs.

Historical Development

The history of numerical methods for solving ODEs traces back to the pioneering work of Leonhard Euler in the 18th century. Euler's original method, often referred to as the Euler Forward method, was a simple yet groundbreaking approach that approximated the solution by advancing along the tangent line at each point. However, Euler's method suffered from stability issues, leading to the development of improved variants.

The Euler's Modified method is an improvement over the original Euler's method, designed to enhance accuracy. By using a midpoint correction, it addresses the inherent errors of the Euler technique and provides more accurate approximations. Butcher J. C. (2008) in his work "Numerical Methods for Ordinary Differential Equations" have highlighted its significance in accurately solving ODEs while maintaining simplicity.

The Runge-Kutta methods, first introduced by Carl Runge and Martin Kutta in the late 19th century, represent a substantial advancement in numerical ODE solving. These methods, particularly the second-order (RK2) and fourth-order (RK4) variants, have gained widespread acceptance due to their accuracy, stability, and adaptability to various ODE types. Over the years, Runge-Kutta methods have become a cornerstone in numerical analysis, underpinning many scientific simulations and engineering applications.

In "Numerical Methods for Engineers and Scientists," Gilat and Subramaniam (2013) explored different numerical methods, including the Runge-Kutta method, particularly the fourth-order Runge-Kutta. They emphasized its higher accuracy and stability compared to the Euler method, making it a preferred choice for solving ODEs requiring precision.

Applications

Numerical methods for solving ODEs find application in diverse fields. In physics, for instance, the Euler Modified and Runge-Kutta methods are integral in modeling the motion of celestial bodies, the behavior of fluids, and electrical circuit dynamics. In biology, these methods aid in modeling population growth, enzyme kinetics, and ecological systems. Additionally, engineers rely on these methods to simulate the behavior of mechanical systems, control systems, and chemical reactions.

Comparative Analysis

Comparative studies between Euler Modified and Runge-Kutta methods have shown that while the Euler method is computationally straightforward, it may lack accuracy, especially with smaller step sizes. On the other hand, the RK4 method, with its multi-stage calculations, yields more precise results, albeit with a slightly higher computational load.

Advancements and Current Research

Current research in the field focuses on hybrid methods that combine the advantages of both Euler Modified and RK4 techniques. Such hybrid methods aim to achieve higher accuracy while minimizing computational costs. Recent studies (Adams et Johnson, 2017) have proposed innovative modifications to these classical methods, offering improved efficiency and accuracy in solving ODEs.

Conclusion

The Euler Modified and Runge-Kutta methods are pivotal in solving first-order ODEs numerically. While each method has its strengths and weaknesses, ongoing research continues to refine these techniques, leading to improved accuracy and computational efficiency in solving differential equations.

CHAPTER THREE

3.0 METHODOLOGY

The First Order Ordinary Differential Equations can be solved using different methods, such as the Euler Method, Heun's Method, Runge-kutta Method, Finite Difference Method, etc. Here, we will discuss the two most important techniques called the Heun's Method and Runge-Kutta Method in detail.

3.1 Euler's Modified (Heun's) Method

The Euler Modified method, also known as the Improved Euler method or Heun's method, is a straightforward yet effective numerical technique for solving first-order ordinary differential equations (ODEs). This method is an improvement over the basic Euler method, addressing some of its limitations.

Characteristics

- Two-Step Process: The Euler Modified method employs a two-step process for approximating the solution at each step. It calculates a preliminary estimate using the slope at the current point and another estimate at a midpoint within the step size. These two estimates are then averaged to obtain the final approximation.
- Accuracy: Compared to the basic Euler method, the Euler Modified method offers improved accuracy. By considering an intermediate point within the step, it reduces truncation errors and provides more accurate results for smoother functions and less steep slopes.

- Simplicity: One of its key advantages is its simplicity. It is easy to implement and computationally efficient, making it a suitable choice for relatively simple ODEs or initial exploration of more complex problems.

Strength

- Improved Accuracy: The primary strength of the Euler Modified method is its enhanced accuracy compared to the basic Euler method. It is particularly useful for ODEs where the slope changes significantly within the step size.
- Ease of Implementation: Due to its simplicity, the Euler Modified method is accessible to those new to numerical ODE solving. It serves as a good starting point for understanding numerical methods.

Limitations

- Stability Issues: The Euler Modified method may exhibit stability issues when applied to stiff ODEs, where rapid changes in the solution occur within the step size. This can lead to numerical instability and inaccurate results.
- Limited Precision: While it offers improved accuracy over the basic Euler method, the Euler Modified method may still suffer from limited precision when dealing with highly nonlinear or oscillatory ODEs.

3.2 Runge-Kutta Method

The Runge-Kutta method is a numerical technique used for solving ordinary differential equations (ODEs) and is particularly effective for solving initial value problems. It's a family of numerical integration methods that are widely used because of their accuracy and ease of implementation. The method was developed by German mathematicians Carl Runge and Martin Kutta in the late 19th and early 20th centuries.

Here's an overview of the Runge-Kutta method

Background: The Runge-Kutta method is used to approximate the solution of a first-order ordinary differential equation of the form:

$$\frac{dy}{dt} = f(t, y).$$

where :

t is the independent variable (usually time),

y is the dependent variable, and

$f(t, y)$ is a known function that describes the rate of change of y with respect to t.

K4 can be expressed as follows for a single time step:

$K_1 = \Delta t \cdot f(t_n, y_n)$.

$K_2 = \Delta t \cdot f\,(t_n + \frac{1}{2}\Delta t\,,\, y_n + \frac{1}{2}K_1)$.

$K_3 = \Delta t \cdot f\,(t_n + \frac{1}{2}\Delta t\,,\, y_n + \frac{1}{2}K_2)$.

$K_4 = \Delta t \cdot f\,(t_n + \Delta t,\ y_n + K_3)$.

$$y_{n+1} = y_n + \frac{1}{6}(K_1 + 2\,K_2 + 2\,K_3 + K_2)\,.$$

where:

y_n is the approximate value of y at time t_n

y_{n+1} is the estimated value of y at time t_{n+1}

K_1, K_2, K_3, and K_4 are intermediate values representing the rate of change of y at different stages within the time step.

General Idea: The method works by breaking down the time interval into discrete steps and approximating the change in y over each step. It then updates the value of y at each step to iteratively compute the solution.

Accuracy: RK4 is a fourth-order method, which means that its error decreases with step size to the fourth power. This makes it more accurate than simpler methods like the Euler method for the same step size.

Advantages:

- RK4 is relatively easy to implement and is suitable for a wide range of differential equations.
- It provides good accuracy, making it a popular choice for numerical simulations.
- The method is stable for many types of problems.

Limitations:

- RK4 can be computationally expensive for very small step sizes, especially in high-dimensional systems.

- It may not be suitable for stiff differential equations, where the solution changes rapidly.

In summary, the Runge-Kutta method, particularly the fourth-order RK4 variant, is a versatile and widely used technique for numerically solving ordinary differential equations. It offers a good balance between accuracy and computational efficiency, making it a valuable tool in various scientific and engineering applications.

CHAPTER FOUR

4.0 APPLICATIONS

4.1 Illustrative Examples

4.1.1 Example on Euler`s Modified Method

We'll solve the following ODE:

$$\frac{dy}{dx} = -2x + 4, \; y(0) = 1 .$$

Step 1: Initialization

We start with the initial conditions:

x_0 = 0 (initial x-value) .

y_0 = 1 (initial y-value) .

h = 0.2 (step size) .

x_{end} = 1 (end of the interval) .

Step 2: Iterations

We'll perform a series of iterations to approximate the solution. At each iteration, we'll calculate the slope at the current point, an intermediate slope, and use them to update the value of y for the next iteration.

Step 3: Calculations

Now, let's calculate the values at each iteration:

- ✓ **Initial Iteration (n = 0):**

$x_0 = 0$.

$y_0 = 1$.

Calculate the slope at x_0 :

$$Slope = -2x_0 + 4 = -2(0) + 4 = 4 .$$

Calculate an intermediate slope at $x_0 + \frac{h}{2}$:

$$Intermediate\ Slope = -2\left(0 + \frac{0.2}{2}\right) + 4 = 3.8 .$$

Update y_1 using the average of the slopes:

$$y_1 = y_0 + \frac{h}{2}\ (Slope + Intermediate\ Slope) = 1 + \frac{0.2}{2} \cdot (4 + 3.8) = 1.38 .$$

✓ **Iteration 1 (n = 1):**

$$x_1 = x_0 + h = 0 + 0.2 = 0.2 .$$

We need to calculate the slope at x_1 :

$$Slope = -2\,x_1 + 4 = -2\,(0.2) + 4 = 3.6 .$$

Calculate an intermediate slope at $x_1 + \frac{h}{2}$:

$$Intermediate\ Slope = -2\left(0.2 + \frac{0.2}{2}\right) + 4 = 3.4 .$$

Update y_1 using the average of the slopes:

$$y_1 = y_0 + \frac{h}{2}\ (Slope + Intermediate\ Slope) = 1 + \frac{0.2}{2}\ (3.6 + 3.4) = 1.36 .$$

✓ **Iteration 2 (n=2):**

$$x_2 = x_1 + h = 0.2 + 0.2 = 0.4 .$$

Calculate the slope at x_2 :

$$Slope = -2\,x_2 + 4 = -2\,(0.4) + 4 = 3.2 .$$

Calculate an intermediate slope at $_2 + \frac{h}{2}$:

$$Intermediate\ Slope = -2\left(0.4 + \frac{0.2}{2}\right) + 4 = 3.0 .$$

Update y_2 using the average of the slopes:

$$y_2 = y_1 + \frac{h}{2}\,(Slope + Intermediate\ Slope) = 1.36 + \frac{0.2}{2}\,(3.2 + 3.0) = 1.72 .$$

✓ **Iteration 3 (n=3):**

$$x_3 = x_2 + h = 0.4 + 0.2 = 0.6 .$$

Calculate the slope at x_3 :

$$Slope = -2x_3 + 4 = -2\,(0.6) + 4 = 2.8 .$$

Calculate an intermediate slope at x3 + $\frac{h}{2}$:

$$Intermediate\ Slope = -2\left(0.6 + \frac{0.2}{2}\right) + 4 = 2.6 .$$

Update y_3 using the average of the slopes:

$$y_3 = y_2 + \frac{h}{2}\,(Slope + Intermediate\ Slope) = 1.72 + \frac{0.2}{2}\,(2.8 + 2.6) = 2.08 .$$

✓ **Iteration 4 (n=4):**

$$x_4 = x_3 + h = 0.6 + 0.2 = 0.8 .$$

Calculate the slope at x_4 :

$$Slope = -2\,x_4 + 4 = -2\,(0.8) + 4 = 2.4\,.$$

Calculate an intermediate slope at $x_4 + \frac{h}{2}$:

$$Intermediate\ Slope = -2\left(0.8 + \frac{0.2}{2}\right) + 4 = 2.2 \quad .$$

Update y_4 using the average of the slopes:

$$y_4 = x_3 + \frac{h}{2}\,(Slope + Intermediate\ Slope) = 2.08 + \frac{0.2}{2} \cdot (2.4 + 2.2) = 2.44 \quad .$$

✓ **Iteration 5 (n=5):**

$$x_5 = x_4 + h = 0.8 + 0.2 = 1.0\ .$$

Calculate the slope at x_5 :

$$Slope = -2\,x_5 + 4 = -2\,(1.0) + 4 = 2.0\ .$$

Calculate an intermediate slope at $x_5 + \frac{h}{2}$:

$$Intermediate\ Slope = -2\left(1.0 + \frac{0.2}{2}\right) + 4 = 1.8\ .$$

Update y5 using the average of the slopes:

$$y_5 = y_4 + \frac{h}{2}\,(Slope + Intermediate\ Slope) = 2.44 + \frac{0.2}{2} \cdot (2.0 + 1.8) = 2.68\,.$$

After completing all five iterations, we have calculated the values for x and y. Here are the results for iterations 1 to 5:

Iteration (n)	x_n	y_n
0(Initial)	0.0	1.000
1	0.2	1.360
2	0.4	1.720
3	0.6	2.080
4	0.8	2.440
5	1.0	2.680

These results represent the approximate solution to the given first-order ODE over the interval [0,1] using the Euler`s Modified method with a step size of h =0.2 .

4.1.2 Additional Example on Euler Modified Method

$$\frac{dy}{dx} = x - y,$$

where $y(0) = 1$.

Let's apply the Euler`s Modified method with a step size $h = 0.1$ over the interval $[0, 0.5]$.

Step 1: Initialization

$x_0 = 0$ (initial x-value) .

$y_0 = 1$ (initial y-value) .

h = 0.1 (step size) .

$x_{end} = 0.5$ (end of the interval) .

h = 0.1 (step size) .

Step 2: Iterations

We'll compute the successive values of x and y using the Euler Modified method.

Iteration 1 ($n = 1$):

- Calculate the slope at $x_1 = x_0 + h = 0 + 0.1 = 0.1$:

  Slope = $X_1 - Y_0 = 0.1 - 1 = 0.9$.

- Calculate an intermediate slope at $x_1 + \frac{h}{2}$:

  Intermediate Slope = $x1 + \frac{h}{2} - \left(y0 + \frac{h}{2} \times Slope\right)$

$$= 0.1 + \frac{0.1}{2} - \left(1 + \frac{0.1}{2} \times (-0.9)\right) = -0.844 \ .$$

- Update y_1 using the average of the slopes:

$$y1 = y0 + h \times Intermediate\ Slope = 1 + 0.1 \times (-0.844) = 0.916\,.$$

Iteration 2 ($n = 2$):

- Calculate the slope at $x2 = x1 + h = 0.1 + 0.1 = 0.2$:

  Slope $= x2 - y1 = 0.2 - 0.916 = -0.716$.

- Calculate an intermediate slope at $x_2 + \frac{h}{2}$:

  Intermediate Slope $= x2 + \frac{h}{2} - (y1 + \frac{h}{2} \times Slope)$

$$= 0.2 + \frac{0.1}{2} - \left(0.916 + \frac{0.1}{2} \times (-0.716)\right) = -0.663 \ .$$

- Update y_2 using the average of the slopes:

$$y2 = y1 + h \times Intermediate\ Slope = 0.916 + 0.1 \times (-0.663) = 0.85 \,.$$

Iteration 3 ($n = 3$):

- Calculate the slope at $x3 = x2 + h = 0.2 + 0.1 = 0.3$:

  Slope $= x3 - y2 = 0.3 - 0.85 = -0.55$.

- Calculate an intermediate slope at $x_3 + \frac{h}{2}$:

  Intermediate Slope $= x3 + \frac{h}{2} - (y2 + \frac{h}{2} \times Slope)$

$$= 0.3 + \frac{0.1}{2} - \left(0.85 + \frac{0.1}{2} \times (-0.55)\right) = -0.496 \ .$$

- Update y_3 using the average of the slopes:

$$y3 = y2 + h \times Intermediate\ Slope = 0.85 + 0.1 \times (-0.496) = 0.8 \,.$$

Iteration 4 ($n = 4$):

- Calculate the slope at $x4 = x3 + h = 0.3 + 0.1 = 0.4$:

  Slope $= x4 - y3 = 0.4 - 0.8 = -0.4$.

- Calculate an intermediate slope at $x_4 + \frac{h}{2}$:

  Intermediate Slope $= x4 + \frac{h}{2} - (y3 + \frac{h}{2} \times Slope)$

$$= 0.4 + \frac{0.1}{2} - \left(0.8 + \frac{0.1}{2} \times (-0.4)\right) = -0.346 .$$

- Update y_4 using the average of the slopes:

$$Y4 = y3 + h \times Intermediate\,Slope = 0.8 + 0.1 \times (-0.346) = 0.765 .$$

Iteration 5 ($n = 5$):

- Calculate the slope at $x5 = x4 + h = 0.4 + 0.1 = 0.5$:

  Slope $= x5 - y4 = 0.5 - 0.765 = -0.265$.

- Calculate an intermediate slope at $x_5 + \frac{h}{2}$:

  Intermediate Slope $= x5 + \frac{h}{2} - (y4 + \frac{h}{2} \times Slope)$

  $= 0.5 + \frac{0.1}{2} - (0.765 + \frac{0.1}{2} \times (-0.265)) = -0.213$.

- Update y_5 using the average of the slopes:

$$y5 = y4 + h \times Intermediate\,Slope = 0.765 + 0.1 \times (-0.213) = 0.744 .$$

Upon completing the iterations:

At $x = 0.5$, the approximated value of y using the Euler modified method is approximately 0.744 .

4.1.3 Example on Runge-Kutta Method

<u>Modeling The Cooling Of A Hot Cup Of Coffee</u>

Problem Statement:

Suppose we have a cup of coffee initially at a temperature of 80°C, and it's placed in a room with a constant temperature of 25°C. The rate at which the coffee cools down follows the first-order ODE:

$$\frac{dT}{dt} = -k \cdot (T - Troom) .$$

where:

T is the temperature of the coffee at time t.

Troom is the room temperature (25°C).

k is the cooling rate constant.

RK4 Implementation:

Initialization:

T (0) = 80 °C (initial temperature)

Troom = 25 °C (room temperature)

K = 0.1 (cooling rate constant)

Δt = 0.5 (time step size)

Iteration 1 (t = 0.5 seconds):

At t = 0.5 seconds, we estimate T(0.5) using the RK4 method:

$K_1 = \Delta t \cdot (-k \cdot (T(0) - Troom))$

$= 0.5 \ ((-0.1)(80 - 25)) = -2.75 .$

$K_2 = \Delta t \cdot (-k \cdot (T(0) + 0.5 \ (K_1) - Troom))$

$= 0.5 \cdot \Big((-0.1)\big(80 + 0.5\,(-2.75\,) - 25\,\big)\Big) = -2.68125\,.$

$K_3 = \Delta t \cdot (-k \cdot (T(0) + 0.5\ (K_2) - Troom\ )\ )$

$= 0.5 \cdot \Big((-0.1)\ \big(80 + 0.5\,(-2.68125) - 25\,\big)\Big) = -2.68297\,.$

$K_4 = \Delta t \cdot (-k \cdot (T(0) - K_3 - Troom\ )\ )$

$= 0.5\ \Big((-0.1)\big(80 - (-2.68297) - 25\big)\Big) = -2.88415\,.$

Update T (0.5) using these values:

$T\ (0.5) = T(0) + \frac{1}{6}\,(K_1 + 2\ K_2 + 2\ K_3 + K_4) = 77.27290\ .$

At t = 0.5 seconds, the estimated coffee temperature is approximately 77.30 °C.

Iteration 2 (t = 1.0 seconds):

At t = 1.0 seconds, we estimate T (1.0) using the RK4 method:

$K_1 = \Delta t \cdot (-k \cdot (T\ (0.5) - Troom\ )\ )$

$= 0.5 \cdot \Big((-0.1)(77.27290 - 25\,)\Big) = -2.61365\,.$

$K_2 = \Delta t \cdot (-k \cdot (T\ (0.5) + 0.5\ (K_1\ ) - Troom\ )\ )$

$= 0.5 \cdot \Big((-0.1)\big(77.27290 + 0.5\ (-2.61365) - 25\,\big)\Big) = -2.54830\,.$

$K_3 = \Delta t \cdot (-k \cdot (T\ (0.5) + 0.5\ (K_2) - Troom\ )\ )$

$= 0.5 \cdot \Big((-0.1)\big(77.27290 + 0.5\ (-2.54830) - 25\,\big)\Big) = -2.54994\ .$

$K_4 = \Delta t \cdot (-k \cdot (T\ (0.5) - K_3 - Troom\ )\ )$

$= 0.5 \cdot \Big((-0.1)\big(77.27290 - (-2.54994) - 25\big)\Big) = -2.74114\,.$

Update T(1.0) using these values:

$T\ (1.0) = T(0.5) + \frac{1}{6}\,(K_1 + 2\ K_2 + 2\ K_3 + K_4) = 74.68102\,.$

At t = 1.0 seconds, the estimated coffee temperature is approximately 74.68 °C.

Iteration 3 (t = 1.5 seconds):

At t = 1.5 seconds, we estimate T(1.5) using the RK4 method:

$K_1 = \Delta t \cdot (-k \cdot (T(1.0) - Troom))$

$= 0.5 \ \left((-0.1) \left(74.68102 - 25 \right)\right) = -2.48405 .$

$K_2 = \Delta t \cdot (-k \cdot (T(1.0) + 0.5 (K_1) - Troom))$

$= 0.5 \ \left((-0.1)\left(74.68102 + 0.5 (-2.48405) - 25 \right)\right) = -2.42195 .$

$K_3 = \Delta t \cdot (-k \cdot (T(1.0) + 0.5 (K_2) - Troom))$

$= 0.5 \ \left((-0.1)\left(74.68102 + 0.5 \cdot (-2.42195) - 25 \right)\right) = -2.42350 .$

$K_4 = \Delta t \cdot (-k \cdot (T(1.0) - K_3 - Troom))$

$= 0.5 \ \left((-0.1)\left(74.68102 - (-2.42350) - 25 \right)\right) = -2.60523 .$

Update T(1.5) using these values:

$T(1.5) = T(1.0) + \frac{1}{6} (K_1 + 2 K_2 + 2 K_3 + K_4) = 72.21766 .$

At t = 1.5 seconds, the estimated coffee temperature is approximately 72.22 °C.

Iteration 4 (t = 2.0 seconds):

At t = 2.0 seconds, we estimate T (2.0) using the RK4 method:

$K_1 = \Delta t \cdot (-k \cdot (T(1.5) - Troom))$

$= 0.5 \cdot \left((-0.1)\left(72.21766 - 25 \right)\right) = -2.36088 .$

$K_2 = \Delta t \cdot (-k \cdot (T(1.5) + 0.5 (K_1) - Troom))$

$$= 0.5 \cdot \Big((-0.1)\big(72.21766 + 0.5\,(-2.36088) - 25\,\big)\Big) = -2.92686\,.$$

$K_3 = \Delta t \cdot (-k \cdot (T\,(1.5) + 0.5\,(K_2) - \text{Troom}))$

$$= 0.5\,\Big((-0.1)\big(72.21766 + 0.5\,(-2.92686) - 25\,\big)\Big) = -2.28771\,.$$

$K_4 = \Delta t \cdot (-k \cdot (T\,(1.5) - K_3 - \text{Troom}))$

$$= 0.5\,\big((-0.1)(72.21766 - (-2.28771) - 25\,)\big) = -2.47527\,.$$

Update T (2.0) using these values:

$T(2.0) = T(1.5) + \frac{1}{6}(K_1 + 2\,K_2 + 2\,K_3 + K_4) = 69.67345$.

At t = 2.0 seconds, the estimated coffee temperature is approximately 69.67 °C.

Iteration 5 (t = 2.5 seconds):

At t = 2.5 seconds, we estimate T (2.5) using the RK4 method:

$K_1 = \Delta t \cdot (-k \cdot (T\,(2.0) - \text{Troom}))$

$$= 0.5 \cdot \Big((-0.1) \cdot \big(69.67345 - 25\,\big)\Big) = -2.23367\,.$$

$K_2 = \Delta t \cdot (-k \cdot (T\,(2.0) + 0.5\,(K_1) - \text{Troom}))$

$$= 0.5 \cdot \Big((-0.1) \cdot \big(69.67345 + 0.5 \cdot (-2.23367) - 25\,\big)\Big) = -2.17783\,.$$

$K_3 = \Delta t \cdot (-k \cdot (T\,(2.0) + 0.5\,(K_2) - \text{Troom}))$

$$= 0.5 \cdot \Big((-0.1) \cdot \big(69.67345 + 0.5 \cdot (-2.17783) - 25\,\big)\Big) = -2.80423\,.$$

$K_4 = \Delta t \cdot (-k \cdot (T\,(2.0) - k3 - \text{Troom}))$

$$= 0.5 \cdot \Big(-0.1 \cdot \big(69.67345 - (-2.80423) - 25\,\big)\Big) = -2.37388\,.$$

Update T(2.5) using these values:

$T(2.5) = T(2.0) + \frac{1}{6}(K_1 + 2\,K_2 + 2\,K_3 + K_4) = 67.24484$.

At t = 2.5 seconds, the estimated coffee temperature is approximately 67.24 °C.

Iteration 6 (t = 3.0 seconds):

Finally, at t=3.0 seconds, we estimate T (3.0) using the RK4 method:

$K_1 = \Delta t \cdot (-k \cdot (T(2.5) - Troom))$

$= 0.5 \cdot \left((-0.1)(67.24484 - 25) \right) = -2.11224$.

$K_2 = \Delta t \cdot (-k \cdot (T(2.5) + 0.5 (K_1) - Troom))$

$= 0.5 \cdot \left((-0.1) \cdot (67.24484 + 0.5 (-2.11224) - 25) \right) = -2.05944$.

$K_3 = \Delta t \cdot (-k \cdot (T(2.5) + 0.5 (K_2) - Troom))$

$= 0.5 \cdot \left((-0.1) \cdot (67.24484 + 0.5 \cdot (-2.05944) - 25) \right) = -2.06076$.

$K_4 = \Delta t \cdot (-k \cdot (T(2.5) - K_3 - Troom))$

$= 0.5 \cdot \left((-0.1)(67.24484 - (-2.06076) - 25) \right) = -2.21528$.

Update T(3.0) using these values:

$T(3.0) = T(2.5) + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) = 65.15019$.

At t = 3.0 seconds, the estimated coffee temperature is approximately 65.15 °C.

These calculations provide a detailed understanding of how the coffee's temperature decreases over time due to its cooling rate.

:

4.1.4 Additional Example on Runge-Kutta Method

Problem

Consider the differential equation

$$\frac{dy}{dx} = x^2 + 1 .$$

with the initial condition $y(0) = 0$ over the interval [0,1] using a step size $h = 0.25$.

Solution

Given:

$y_0 = 0$ (initial value of y) .

$x_0 = 0$ (initial value of x) .

$h = 0.25$ (step size) .

Iteration 1 ($x = 0$):

We'll calculate k_1, k_2, k_3, and k_4

$$k_1 = h \times (x_0^2 + 1) = 0.25 \times (0^2 + 1) = 0.25 \ .$$

$$k_2 = h \times \left(x_0^2 + 1 + \frac{k_1}{2}\right) = 0.25 \times (0^2 + 1 + 0.125) = 0.28125 \ .$$

$$k_3 = h \times \left(x_0^2 + 1 + \frac{k_2}{2}\right) = 0.25 \times (0^2 + 1 + \frac{0.28125}{2}) = 0.28516 \ .$$

$$k_4 = h \times (x_0^2 + 1 + k_3) = 0.25 \times (0^2 + 1 + 0.28516\ ) = 0.32129 \ .$$

$$y_1 = y_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$= 0 + \tfrac{1}{6}(0.25 + 2(0.28125) + 2(0.28516) + 0.32129) = 0.28402.$$

$$x_1 = x_0 + h = 0 + 0.25 = 0.25.$$

Iteration 2 ($x = 0.25$):

We'll calculate k_1, k_2, k_3, and k_4

$$k_1 = h \times (x_1^2 + 1) = 0.25 \times (0.25^2 + 1) = 0.26563.$$

$$k_2 = h \times \left(x_1^2 + 1 + \frac{k_1}{2}\right) = 0.25 \times (0.25^2 + 1 + \frac{0.26563}{2}) = 0.29883.$$

$$k_3 = h \times \left(x_1^2 + 1 + \frac{k_2}{2}\right) = 0.25 \times (0.25^2 + 1 + \frac{0.29883}{2}) = 0.30298.$$

$$k_4 = h \times (x_1^2 + 1 + k_3) = 0.25 \times (0.25^2 + 1 + 0.30298\ ) = 0.34137.$$

$$y_2 = y_1 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) =$$

$$0.28402 + \frac{1}{6}(0.26563 + 2(0.29883) + 2(0.30298) + 0.34137) = 0.58579.$$

$$x_2 = x_1 + h = 0.25 + 0.25 = 0.5.$$

Iteration 3 ($x = 0.5$):

We'll calculate k_1, k_2, k_3, and k_4

$$k_1 = h \times (x_2^2 + 1) = 0.25 \times (0.5^2 + 1) = 0.3125.$$

$$k_2 = h \times \left(x_2^2 + 1 + \frac{k_1}{2}\right) = 0.25 \times (0.5^2 + 1 + \frac{0.3125}{2}) = 0.3516.$$

$$k_3 = h \times \left(x_2^2 + 1 + \frac{k_2}{2}\right) = 0.25 \times (0.5^2 + 1 + \frac{0.3516}{2}) = 0.3565\ .$$

$$k_4 = h \times (x_2^2 + 1 + k_3) = 0.25 \times (0.5^2 + 1 + 0.3565\ ) = 0.4016.$$

$$y_3 = y_2 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) =$$

$$0.58579 + \frac{1}{6}(0.3125 + 2(0.3516) + 2(0.3565) + 0.4016) = 0.94084.$$

$$x_3 = x_2 + h = 0.5 + 0.25 = 0.75\ .$$

Iteration 4 ($\boldsymbol{x = 0.75}$):

We'll calculate k_1, k_2, k_3, and k_4

$k_1 = h \times (x_3^2 + 1) = 0.25 \times (0.75^2 + 1) = 0.39063.$

$$k_2 = h \times \left(x_3^2 + 1 + \frac{k_1}{2}\right) = 0.25 \times (0.75^2 + 1 + \frac{0.39063}{2}) = 0.43945\ .$$

$$k_3 = h \times \left(x_3^2 + 1 + \frac{k_2}{2}\right) = 0.25 \times (0.75^2 + 1 + \frac{0.43945}{2}) = 0.44556.$$

$$k_4 = h \times (x_3^2 + 1 + k_3) = 0.25 \times (0.75^2 + 1 + \ 0.44556.) = 0.50202.$$

$$y_4 = \ y_3 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) =$$

$$= 0.94084 + \ \frac{1}{6}(0.39063 + 2(0.43945) + 2(0.44556)) + 0.50202) = 1.38462$$

$$x_4 = x_3 + h = 0.75 + 0.25 = 1.$$

At $\boldsymbol{x = 1}$, the estimated solution is approximately $y_4 = 1.38462$.

CHAPTER FIVE

5.0 CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In the realm of numerical methods for solving first-order ordinary differential equations (ODEs), the Euler Modified and Runge-Kutta methods have proven to be invaluable tools for approximating solutions across various scientific and engineering domains. The Euler Modified method, as an improvement over the basic Euler method, offers enhanced accuracy, making it suitable for ODEs with relatively smooth solutions. Euler Modified and Runge-Kutta methods have been shown to be applicable in various scientific and engineering disciplines, including physics, biology, engineering, and economics. The selection of the appropriate method should align with the specific problem characteristics and requirements of the application domain.

5.2 Recommendations

I recommend that when dealing with ODEs that exhibit rapid changes or oscillations, prioritize the use of Runge-Kutta methods, especially RK4, due to their superior accuracy and stability in such scenarios. For relatively simple problems with smooth solutions, Euler Modified seems to be efficient choice. Also, I encourage researchers to stay updated with advancements in numerical ODE solving techniques and consider incorporating newer methods into their toolkit as they emerge.

REFERENCES

[1] Butcher, J. C. (2008). Numerical Methods for Ordinary Differential Equations, John Wiley and Sons pp. 285-295.

[2] Hairer, E., and Wanner, G. (1996). Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, Springer-Verlag pp. 108-114.

[3] Hairer, E., Nørsett, S. P., and Wanner, G. (1993). Solving Ordinary Differential Equations I: Nonstiff Problems, Springer pp. 87-95.

[4] Lambert, J. D. (1973). Computational Methods in Ordinary Differential Equations, John Wiley and Sons pp. 65-72.

[5] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). Numerical Recipes: The Art of Scientific Computing, Cambridge University Press pp. 85-95.

[6] Quarteroni, A., Sacco, R., and Saleri, F. (2007). Numerical Mathematics (2nd ed.), Springer-Verlag pp. 125-141.

[7] Shampine, L. F., and Reichelt, M. W. (1997). The MATLAB ODE Suite, Society for Industrial and Applied Mathematics (SIAM) Journal on Scientific Computing, 18(1), 1-22.

[8] Stoer, J., and Bulirsch, R. (2002). Introduction to Numerical Analysis, Springer-Verlag pp. 52-63.