VARIABLE STEP VARIABLE ORDER BLOCK BACKWARD DIFFERENTIATION METHOD FOR SOLVING DIRECTLY HIGHER ORDER STIFF ORDINARY DIFFERENTIAL EQUATIONS

ASMA IZZATI BINTI ASNOR

UNIVERSITI SAINS MALAYSIA

2023

VARIABLE STEP VARIABLE ORDER BLOCK BACKWARD DIFFERENTIATION METHOD FOR SOLVING DIRECTLY HIGHER ORDER STIFF ORDINARY DIFFERENTIAL EQUATIONS

by

ASMA IZZATI BINTI ASNOR

Thesis submitted in fulfilment of the requirements
for the degree of
Doctor of Philosophy

September 2023

ACKNOWLEDGEMENT

Alhamdulillah, with profound gratitude, I embrace the culmination of this significant journey, a testament to determination and resilience. First and foremost, I extend my heartfelt gratitude to my dedicated supervisor, Ts. Dr. Siti Ainor Binti Mohd Yatim. Your unwavering support, insightful guidance, and mentorship have been instrumental in my research journey. Your patience, expertise, and steadfast commitment to excellence have not only shaped this thesis but have also greatly contributed to my intellectual growth. I am profoundly grateful for the invaluable opportunities to learn under your guidance. I dedicate this thesis to the unwavering love and support of my family, especially my parents, as their sacrifices and encouragement have been the bedrock of my academic journey. Their belief in my abilities and their constant motivation has propelled me forward, even during the most challenging moments. To my cherished friends and loved ones, their boundless love and unwavering support, coupled with the camaraderie of my dear friends, have sweetened this challenging journey. This thesis is a heartfelt tribute to their enduring faith in my abilities. With heartfelt gratitude, I extend my thanks for being my source of inspiration, my refuge, and my motivation. Your presence in my life has made this academic endeavour more vibrant and meaningful. Each of you has played an indispensable role in shaping the person I have become, and for that, I am profoundly thankful. This thesis not only represents my academic accomplishments but is a reflection of the collective strength and support that have propelled me forward. As I stand at this milestone, I carry with me the lessons, values, and cherished memories from each one of you. Together, we have not only achieved academic milestones but have also created enduring bonds that I deeply cherish..

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF SYMBOLS

b	Number of blocks
c	Safety factor
h	Step size
i	Number of points in the current block
k	Number of backvalues
m	Order of the proposed method
M	Order of the ODE problem
n	Number of equations
q	Step size ratio of second previous block
r	Step size ratio of first previous block
s	Number of steps
h_{new}	Current step size
h_{old}	Previous step size
$h_{\max}$	Maximum step size
$L_m(r)$	Stability polynomial of m^{th}-order method
$L_{(2)}(r)$	Stability polynomial of 2nd-order method
$L_{(3)}(r)$	Stability polynomial of 3rd-order method
$L_{(4)}(r)$	Stability polynomial of 4th-order method
$\mathrm{LTE}_{(m)}$	Local Truncation Error for m^{th}-order method
$\mathrm{LTE}_{(2)}$	Local Truncation Error for 2nd-order method

$\mathrm{LTE}_{(3)}$	Local Truncation Error for 3rd-order method
$\mathrm{LTE}_{(4)}$	Local Truncation Error for 4th-order method
y_{n+1}	First current value
y_{n+2}	Second current value
$y_{n+1}^{(P)}$	First predictor value
$y_{n+2}^{(P)}$	Second predictor value
$y_{n+1}^{(C)}$	First corrector value
$y_{n+2}^{(C)}$	Second corrector value
$y_{n-4},\dots,y_n$	Previous values
$y_{n+2}^{(m)}$	Corrector formula of y_{n+2} of order m
$y_{n+2}^{(m-1)}$	Corrector formula of y_{n+2} of order $m-1$
α	Coefficient of corrector values
$\hat{\alpha}$	Coefficient of predictor values
β	Coefficient of first derivative for the corrector values
$\hat{\beta}$	Coefficient of first derivative for the predictor values
γ	Coefficient of second derivative for the corrector values
$\hat{\gamma}$	Coefficient of second derivative for the predictor values
μ	Coefficient of the function
$\rho(r)$	First characteristic polynomial
$\sigma(r)$	Second characteristic polynomial

LIST OF ABBREVIATIONS

2VS-BBDF(3)	2^{nd}-order Variable Step Block Backward Differentiation Formulae for solving third-order stiff ODEs
2BBDF	Direct Two Point Block Backward Differentiation Formula for solving second-order stiff ODEs
2BBDF(3)	3^{rd} Order Direct Block Backward Differentiation Formula for solving second-order stiff ODEs
2BBDF(4)	4^{th} Order Direct Block Backward Differentiation Formula for solving second-order stiff ODEs
2BBDF(5)	5^{th} Order Direct Block Backward Differentiation Formula for solving second-order stiff ODEs
2DBBDF	Two-Point Diagonally Implicit Block Backward Differentiation Formula for solving second-order stiff ODEs
2DBBDF(2)VS	Variable Step Diagonal Block Backward Differentiation Formulae for solving second-order stiff ODEs
2VS-BBDF(4)	2^{nd}-order Variable Step Block Backward Differentiation Formulae for solving fourth-order stiff ODEs
3BBDF	Direct Block Backward Differentiation Formula for solving third-order stiff ODEs
3-BBDF	Variable Step Size 3-Point Block Backward Differentiation Formulas for solving first-order stiff ODEs
3VS-BBDF(3)	3^{rd}-order Variable Step Block Backward Differentiation Formulae for solving third-order stiff ODEs
3VS-BBDF(4)	3^{rd}-order Variable Step Block Backward Differentiation Formulae for solving fourth-order stiff ODEs
4VS-BBDF(3)	4^{th}-order Variable Step Block Backward Differentiation Formulae for solving third-order stiff ODEs
5-DBBDF	Direct Block Backward Differentiation Formula for solving second-order stiff ODEs
5-GEAR	Fifth Order of Gear’s method for solving first-order ODEs

AEBDF	Adaptive Extended Backward Differentiation Formula for solving first-order stiff ODEs
AVE	Average error
BBDF	Block Backward Differentiation Formula
BBDF(5)	Fifth Order Two-Point Block Backward Differentiation Formulas for solving first-order stiff ODEs
BBDF2	Variable Step Block Backward Differentiation Formulas for solving second-order stiff ODEs
BBDF2-α	Block Backward Differentiation Alpha-Formula for solving second-order stiff ODEs
BBDFO(6)	Block Backward Differentiation Formula with Off-Step Points of Order 6 for solving first-order stiff ODEs
BDF	Backward Differentiation Formula
BBDF(6)	Block Backward Differentiation Formula of order six
BDFVS	Variable Step Variable Order Backward Differentiation Formula for solving first-order ODEs
BEBDF	Block Extended Backward Differentiation Formula for solving first-order stiff ODEs
CBDF	Continuous Block Backward Differentiation Formula for solving first-order stiff ODEs
DBBDF	Third Order Direct Block Backward Differentiation Formulas for solving second-order stiff ODEs
DE	Differential Equation
DI	Direct Integration
EBDF	Extended Backward Differentiation Formula for solving first-order stiff ODEs
FS	Failure steps
HBDF	Hybrid Backward Differentiation Formula for solving first-order stiff ODEs
IVMs	Initial Value Methods
IVPs	Initial Value Problems
LMM	Linear Multistep Method

LTE	Local Truncation Error
MABDF	Mixed BBDF and Block of Adams Type Formula for solving second-order stiff ODEs
MEBDF	Modified Extended Backward Differentiation Formulae for solving first-order stiff ODEs
*m*VS-BBDF(3)	m^{th}-order Variable Step Block Backward Differentiation Formulae for solving third-order stiff ODEs
*m*VS-BBDF(4)	m^{th}-order Variable Step Block Backward Differentiation Formulae for solving fourth-order stiff ODEs
*m*VS-BBDF(*M*)	m^{th}-order Variable Step Block Backward Differentiation Formulae for solving M^{th}-order stiff ODEs
MXE	Maximum error
NTS	Number of total steps
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PECE	Predict-Evaluate-Correct-Evaluate
*p*3	Third-Order Variable Step Block Backward Differentiation Formula for solving first-order stiff ODEs
*p*4	Fourth-Order Variable Step Block Backward Differentiation Formula for solving first-order stiff ODEs
*p*5	Fifth-Order Variable Step Block Backward Differentiation Formula for solving first-order stiff ODEs
RKTF4	Three-stage of Runge-Kutta of order four
RKTF5	Four-stage of Runge-Kutta of order five
STDRKT	Two-derivative Runge-Kutta type
STDRKT2(5)	Two-stages of Runge-Kutta of order five
STDRKT3(6)	Three-stages of Runge-Kutta of order six
SS	Successful steps
TLV	Tolerance limit value
USM	Universiti Sains Malaysia

VOBBDF	Variable Order 2-Point Block Backward Differentiation Formulas for solving second-order stiff ODEs
VS-BBDF(2)	Variable Step Block Backward Differentiation Formula for solving second-order stiff ODEs
VS-BBDF(3)	Variable Step Block Backward Differentiation Formulae for solving third-order stiff ODEs
VS-BBDF(M)	Variable Step Block Backward Differentiation Formulae for solving M^{th}-order stiff ODEs
VSVO-BBDF	Variable Step Size Variable Order Block Backward Differentiation Formula for solving first-order stiff ODEs
VSVO-BBDF(2)	Variable Step Size Variable Order Block Backward Differentiation Formula for solving second-order stiff ODEs
VSVO-BBDM	Variable Step Variable Order Block Backward Differentiation Method
VDP	Van der Pol

LIST OF APPENDICES

KAEDAH PEMBEZAAN BLOK KE BELAKANG LANGKAH BERUBAH PERINGKAT BERUBAH BAGI MENYELESAIKAN PERSAMAAN PEMBEZAAN BIASA KAKU PERINGKAT LEBIH TINGGI SECARA LANGSUNG

ABSTRAK

Tesis ini menekankan pembangunan Kaedah Pembezaan Blok Ke belakang Langkah Berubah Peringkat Berubah (VSVO-BBDM) bagi menyelesaikan persamaan-persamaan pembezaan biasa (ODEs) kaku peringkat lebih tinggi secara langsung. Kurangnya penyelidikan dalam menyelesaikan ODEs kaku peringkat tinggi secara langsung, terutamanya untuk peringkat tiga dan lebih tinggi, terbukti dalam literatur sedia ada. Oleh itu, adalah penting untuk mengambil peranan bagi mengkaji dan membincangkan penyelesaian langsung untuk ODEs kaku peringkat tinggi ini, khususnya pada peringkat tiga dan empat. Kaedah ini menghasilkan satu set penyelesaian baharu dalam satu blok pada setiap langkah pengamiran sepanjang selang. Bahagian pertama tesis membincangkan kerja pengkomputeran kaedah Langkah Berubah Formula Pembezaan Blok Ke belakang peringkat ke-*m* (*m*VS-BBDF(3)) untuk penyelesaian berangka ODEs kaku peringkat ketiga secara langsung. Masalah ini diselesaikan secara langsung tanpa melalui proses penurunan ke sistem peringkat pertama. Kaedah *m*VS-BBDF(3) dilaksanakan dalam pendekatan saiz langkah berubah. Sementara itu, bahagian kedua tesis ini pula merangkumi kerja pengkomputeran kaedah VSVO-BBDM untuk menyelesaikan ODEs kaku peringkat yang lebih tinggi secara langsung. Kerja pengkomputeran VSVO-BBDM dijalankan dengan menggunakan strategi memvariasikan saiz langkah dan memvariasikan peringkat. Penambahbaikan strategi ini bertujuan untuk meningkatkan kecekapan

kaedah yang dicadangkan bagi menganggarkan penyelesaian dengan berkesan. Selain itu, perbincangan terperinci tentang penumpuan dan sifat kestabilan bagi kaedah-kaedah yang dicadangkan juga turut disertakan. Kemudian, semua kerja pengkomputeran ditulis dalam platform “Microsoft Visual C++” dan eksperimen berangka dilaksanakan bagi mengesahkan kecekapan VSVO-BBDM. Seterusnya, hasil berangka yang diperoleh daripada kaedah yang dicadangkan dan penyelesai-penyelesai yang sedia ada telah dibandingkan dalam jadual dan rajah. Kesimpulannya, keputusan berangka dengan jelas menunjukkan bahawa ketepatan penyelesaian bertambah baik dan kos pengiraan berkurang dengan menggunakan kaedah langsung yang dicadangkan, VSVO-BBDM. Secara keseluruhannya, penggunaan kaedah langsung VSVO-BBDM untuk menganggarkan penyelesaian bagi masalah yang dipertimbangkan adalah lebih cekap daripada kaedah-kaedah setanding. Oleh itu, VSVO-BBDM boleh dicadangkan sebagai penyelesai alternatif bagi menyelesaikan ODEs kaku peringkat yang lebih tinggi secara langsung.

VARIABLE STEP VARIABLE ORDER BLOCK BACKWARD DIFFERENTIATION METHOD FOR SOLVING DIRECTLY HIGHER ORDER STIFF ORDINARY DIFFERENTIAL EQUATIONS

ABSTRACT

This thesis emphasises on developing Variable Step Variable Order Block Backward Differentiation Method (VSVO-BBDM) for solving directly higher-order stiff ordinary differential equations (ODEs). The scarcity of research on solving higher-order stiff ODEs directly, especially for order three and higher, is evident in the existing literature. As a result, it is crucial to take up the mantle of investigating and elucidating the direct solutions for these higher-order stiff ODEs, specifically for orders three and four. This method generates a set of new solutions in a block at each integration step along the interval. The first part of the thesis discusses the computational work m^{th}-order Variable Step Block Backward Differentiation Formula (*m*VS-BBDF(3)) method for direct numerical solutions of third-order stiff ODEs. These problems are directly solved without going through the reduction process to the first-order system. The *m*VS-BBDF(3) method is implemented in the variable step size approach. Meanwhile, the second part of this thesis comprises the computational work of the VSVO-BBDM for solving the higher-order stiff ODEs directly. The computational work of the VSVO-BBDM is carried out using a strategy of varying the step size and varying the order. The advancement of this strategy is intended to enhance the efficiency of the proposed methods to approximate the solutions effectively. Besides, a detailed discussion of the convergence and stability properties of the proposed methods is also included. Then, all the computational works are written in the Microsoft Visual C++ platform, and numerical experiments are

conducted to confirm the efficiency of the VSVO-BBDM. Subsequently, the numerical results obtained from the proposed method and the existing methods are compared in tables and figures. In conclusion, the numerical results clearly demonstrate that the accuracy of the solutions is improved and the computational cost is reduced using the proposed direct method, VSVO-BBDM. As a whole, the use of the direct method VSVO-BBDM for approximating the solutions of the considered problem is more efficient than comparable methods. Hence, the VSVO-BBDM is reliable and can be recommended as an alternative solver for solving the higher-order stiff ODEs directly.

CHAPTER 1

INTRODUCTION

1.1 Introduction

Numerical analysis is a branch of mathematics that solves mathematical problems that commonly occur in science and engineering using various numerical methods introduced by scientists or, more specifically, mathematicians. The numerical method is a technique to obtain the approximated solutions of mathematical problems. Furthermore, it can provide approximate solutions for complicated problems for which analytical solutions may be difficult or sometimes impossible to find.

In addition, the mathematical problems such as differential equation (DE) or a system of DE can be applied to model the problems that occurred in the real-world. In general, the DE is an equation that consists of a function and its one or more derivatives. It describes how things changed. For instance, how fast a disease spread or how fast a population change. In the equation, the function is represented as physical quantities and the derivatives are defined as the rate of change of the function. The DE is categorised into two categories either as an ordinary differential equation (ODE henceforth) or partial differential equation (PDE henceforth). The ODE contains one or more ordinary derivatives of a function (dependent variable) with respect to a single independent variable. Meanwhile, the PDE contains differentials with respect to several independent variables.

ODE can be classified into stiff or non-stiff cases. For solving stiff problems, not all the validated numerical methods for solving ODE can be suitably applied. The numerical methods might be less efficient as the step size used is relatively small, thus requiring more computational time and effort. Implicit methods are therefore used in order to effectively solve this type of problem. One of the most popular implicit method for solving stiff ODEs is Backward Differentiation Formula (BDF) method (Ibrahim et al., 2007a; Akinfewa et al., 2013; Musa, 2013).

There are many numerical methods that can be used to solve the higher-order ODEs. Normally, the existing numerical methods such as sixth-order *P*-stable symmetric multistep method, one leg multistep, or linear symmetric multistep method often necessitate the reduction process in order to solve higher-order differential equations. By using these methods, the higher-order problems will be reduced into their equivalent first-order systems. Afterwards, a suitable first-order numerical method is used to find the approximations. However, applying these classical first-order methods will require a lot of computational efforts and lengthy computation times to generate the solutions since this approach will increase the number of equations to Mn, where M is the order of the problem and n is the number of equations in higher order form (Ibrahim et al., 2019). The direct approach to higher-order equations is believed to offer advantages in terms of speed and accuracy (Gear, 1967).

The evolution of various numerical methods in the literature is intended to improve the existing methods by offering better solutions at which the problems are solved directly without reducing to its first-order system. Other than it being simpler, the direct method can potentially reduce computational burden and save computational time.

1.2 Problem Statement

Various scholars have attempted to solve the higher-order ODEs. Earlier researchers such as Lambert and Watson (1976), Suleiman (1979), Fatunla (1984), and Jain et al. (1984) used first-order ODEs methods to find the numerical solutions of the second-order ODEs by reducing the second-order ODEs problems into their first-order ODEs systems but these methods were found to be inefficient. The deficiency of the approach is that more functions are needed to be computed, which leads to an increase in computational burden and wasted computation time.

Recently, scholars such as Ibrahim et al. (2009), Zainuddin (2011), Yatim (2013b), Zainuddin et al. (2014), Zainuddin et al. (2015), Zainuddin et al. (2016a), Zainuddin (2016b), Ibrahim et al. (2018), Ibrahim et al. (2019), Zawawi et al. (2021), and Zainuddin et al. (2023) have solved the higher-order stiff ODEs problems directly without reducing them into a system of first-order. However, research on solving the higher-order stiff ODEs directly, particularly for order three and higher, has not been widely conducted, as most research focuses on solving the second-order stiff ODEs directly. The scarcity of research on solving higher-order stiff ODEs directly, especially for order three and higher, is evident in the existing literature. As a result, it is crucial to take up the mantle of investigating and elucidating the direct solutions for these higher-order stiff ODEs, specifically focusing on orders three and four. By embarking on this essential study, a critical gap in knowledge for various scientific and engineering domains can be filled. In doing so, the related challenges can be addressed, hence simplifying the computational process.

Motivated by the previous study of Yatim (2013b), this has been the inspiration for us to extend the study on solving the higher-order stiff ODEs directly since the study proved that the direct method such as BBDF is efficient for solving the second-order stiff ODEs.

1.3 Objective of the Thesis

The main objective of this thesis is to develop the numerical methods for solving the higher-order stiff ODEs, namely Variable Step Variable Order Block Backward Differentiation Method. The objectives are:

1. To derive direct methods of second-, third-, and fourth-order Variable Step Block Backward Differentiation Formulae (*m*VS-BBDF(3), $m = 2, 3,$ and 4) by varying the step size for solving third-order stiff ODEs,

2. To derive the Variable Step Variable Order Block Backward Differentiation Method (VSVO-BBDM) by varying the step size and order for solving higher-order stiff ODEs directly,

3. To establish the stability and convergence properties of VSVO-BBDM for solving the higher-order stiff ODEs,

4. To develop the code of the VSVO-BBDM for solving the higher-order stiff ODEs directly; and

5. To analyse the numerical results of the VSVO-BBDM on solving the higher-order stiff ODEs directly in terms of error, time, and number of total steps with the existing methods.

1.4 Scope of the Thesis

This thesis emphasises the direct numerical solutions of higher-order stiff ODEs. The single and system of Initial Value Problems (IVPs) of higher-order stiff ODEs will be solved directly without going through the reduction process to their first-order system. The proposed numerical methods for approximating the solutions of higher-order stiff ODEs of orders three and four are known as Variable Step Variable Order Block Backward Differentiation Method. This method implements variable step size and variable order approach at which this strategy is designed based on the value of Local Truncation Error (LTE). The conclusions are drawn based on the methods' numerical performance with the existing methods in terms of accuracy, computational time, and number of total steps taken when applied to the selected tested problems at various intervals.

1.5 Outline of the Thesis

The thesis is organised into eight chapters. Chapter 1 provides a brief introduction and a basic idea of the numerical methods, along with the highlighted problems. This chapter also discusses the problem statement, objectives, and scope of the study.

In Chapter 2, the related theories and definitions are discussed in detail so as to support the current study. Then, previous studies on the block method, BDF method, and BBDF method are reviewed in relation to the current study.

Chapters 3, 4, and 5 comprise the step-by-step derivation of mVS-BBDF(3) methods (m = 2, 3, and 4) for solving the third-order stiff ODEs. These methods are derived using several backvalues in the previous blocks by applying a variable step size strategy. Thus, these chapters detail the conditions for the successful step and the failure step, as well as a description of how the strategy of varying the step size works. The analysis of the order and stability of the methods is included in these chapters. Also, the convergence of those is analysed in Chapter 5. In addition, the numerical results are presented in Chapter 5 to validate the efficiency of these methods over the existing methods for solving third-order stiff ODEs.

In Chapter 6, the VSVO-BBDM is derived with variable step sizes and orders. Also, the convergence and stability properties of the methods are further discussed in this chapter. The strategy of varying the step size and order is described briefly in this chapter. Besides, the convergence and stability of the method are also analysed.

The problems tested, numerical results, and discussion of the comparison between VSVO-BBDM that was proposed in the previous chapter and the existing methods are included in Chapter 7. Once the programming code of the VSVO-BBDM is developed, several problems of higher-order stiff ODEs are tested to examine the performance of the proposed direct method. The numerical results obtained from the numerical experiments are tabulated, and the performance of the proposed method is discussed and compared to that of the existing methods.

The conclusion of the main findings of the current study is presented in the last chapter, followed by the recommendations for potential future studies.

CHAPTER 2

BACKGROUND THEORY AND LITERATURE REVIEW

2.1 Introduction

The definition of higher-order ODEs and stiffness will be discussed in this chapter. The LMM, which is the characteristic of the BBDF method, is also explained. Following that is the discussion on the types of MATLAB's ODE solvers designed for solving stiff and non-stiff ODE problems. Then, the previous studies on the direct method, block method, BDF method, and BBDF method are reviewed in relation to the current study.

2.2 Background Theory

This section discussed the mathematical theory of the considered problem and method.

2.2.1 Higher-Order Ordinary Differential Equations

Real-world problems also involve the higher-order ODEs. The focus of this thesis is on the IVPs of higher-order ODEs, specifically third- and fourth-order ODEs. Due to the lack of research on these problems, they require a little more attention. The definition of the *n*-dimensional system of the higher-order ODEs of M^{th}-order as written in Zainuddin (2011) is given by:

$$y_i^{(M)} = f_i\left(x,\bar{Y}\right), \quad i = 1,2,\ldots,n, \quad x \in \left[x_0, x_z\right], \tag{2.1}$$

where M is order of the ODE system, x_0 and x_z are finite with initial conditions

$$\bar{Y}\left(x_0\right) = \bar{\xi},$$

where

$$\bar{Y}(x) = (y_1, \ldots, y_1^{(M-1)}, \ldots, y_n, \ldots, y_n^{(M-1)}),$$

$$\bar{\xi}(x) = (\xi_1, \ldots, \xi_1^{(M-1)}, \ldots, \xi_n, \ldots, \xi_n^{(M-1)}).$$

The thesis adopts the following theorem which states the conditions on $f_i\left(x, \bar{Y}\right)$ that guarantee the existence of a unique solution to (2.1).

Theorem 2.1: Existence and Uniqueness (Henrici, 1962) Let $f_i\left(x, \bar{Y}\right)$ be defined and continuous for all points $\left(x, \bar{Y}\right)$ in the region *D* to be defined by interval $x_0 \le x \le x_z$, $\left\|\bar{Y}\right\| < \infty$ where x_0 and x_z are finite, and let there exists a constant *L* known as Lipschitz constant such that for any $x \in \left[x_0, x_z\right]$, and any $\bar{Y}$ and $\bar{Y}^*$ for which $\left(x, \bar{Y}\right)$ and $\left(x, \bar{Y}^*\right)$ are both in *D*,

$$\left| f_i\left(x, \bar{Y}\right) - f_i\left(x, \bar{Y}^*\right) \right| \le L\left\|\bar{Y} - \bar{Y}^*\right\|. \tag{2.2}$$

Then, if $\bar{\xi}$ is any given number, there exist a unique solution $\bar{Y}\left(x\right)$ of (2.1) where $\bar{Y}\left(x\right)$ is continuous and differentiable for all $\left(x, \bar{Y}\right)$ in *D*. The requirement of (2.2) is known as *Lipschitz condition*.

The proof of Theorem 2.1 can be referred to in Henrici (1962). This assumption establishes the existence of a unique solution to (2.1).

2.2.2 Stiff Ordinary Differential Equations

Some of the ODE problems are presented with the phenomenon of stiffness. This phenomenon occurs in a variety of applications, such as the study of motion and mass systems, chemical kinetics, electronic circuits, control systems, and chemical reactions, to name a few (Sandu et al., 1997; Kim & Cho, 1997; Aiken, 1985; Soomro et al., 2023).

The stiffness of the ODE problem is determined when its solutions are varied slowly, but the nearby solutions are varied rapidly. Unfortunately, there is no unique definition of stiffness given in the literature. Hence, the definition given in Lambert (1993) describing how (2.1) exhibits stiffness is considered as follows:

Definition 2.1: Stiffness (Lambert, 1993) The problems are stiff if they satisfy the following conditions:

i. $\mathrm{Re}(\lambda_i) < 0$, $i = 1, 2, \ldots, n$, and

ii. $\max \left|\mathrm{Re}(\lambda_i)\right| >> \min \left|\mathrm{Re}(\lambda_i)\right|$, where λ_i are the eigenvalues of the Jacobian matrix, $\frac{\partial f}{\partial y}$ and the ratio $\frac{\max_i \left|\mathrm{Re}(\lambda_i)\right|}{\min_i \left|\mathrm{Re}(\lambda_i)\right|}$ is called the stiffness ratio of stiffness index.

In fact, conventional methods are quite expensive, and they cannot handle stiff problems efficiently because many steps are required, thus risking round off errors that may invalidate the solution (Shampine & Gear, 1979). To put it simply, a lot of effort and longer time are required for the conventional methods to converge to a satisfactory solution, in which the step size used in the entire integration is extremely small. Stiff problems are, therefore, difficult to solve by using the conventional explicit methods.

2.2.3 Linear Multistep Method

In contrast to the one-step method, a multistep method uses several prior solutions to compute the subsequent approximated solutions. A linear combination of the values of the computed solution and function at the prior points is known as the linear multistep method (LMM). There are two types of LMM: explicit and implicit. An example of explicit LMM is the Adams-Bashforth method, whereas examples of implicit LMM are the Adams-Moulton and BDF methods. Generally, the explicit methods are suitable for solving non-stiff problems and the implicit methods are suitable for solving stiff problems (Alexander, 1977; Cash, 1979; Griffiths & Higham, 2010).

The general expression of *s*-step LMM has the form

$$\sum_{j=0}^{s} \alpha_j y_{n+j} = h \sum_{j=0}^{s} \beta_j f_{n+j} , \tag{2.3}$$

where h is the step size, α_j, β_j are method's coefficients and constant, and s denotes the number of steps in the method. Coefficients are presumed to be real and satisfy the conditions

$$\alpha_s = 1 \text{ and } |\alpha_0| + |\beta_0| \neq 0 .$$

In order for method (2.3) to be implicit, y_{n+s} appears on both sides of the equation where f_{n+s} on the right-hand side is $f_{n+s} = f\left(x_{n+s}, y_{n+s}\right), \quad \beta_s \neq 0.$ Meanwhile, if $\beta_s = 0$, the method is explicit.

Following (2.3), the general form of the LMM for the higher-order ODEs is written as

$$\sum_{j=0}^{s} \alpha_j y_{n+j} = h \sum_{j=0}^{s} \beta_j y'_{n+j} + h^2 \sum_{j=0}^{s} \gamma_j y''_{n+j} + \ldots + h^M \sum_{j=0}^{s} \mu_j f_{n+j}, \tag{2.4}$$

where *M* is the order of the ODEs and $f_{n+j} = y_{n+j}^{(M)}$.

BBDF method is one of the LMM for solving higher-order stiff ODEs and it is also an implicit method. The BBDF method will be covered in greater detail in the next section.

2.2.4 MATLAB's ODE Solver

In approximating the solutions, MATLAB's ODE solvers can only solve first-order equations. In order to use the solvers, the higher-order equations are rewritten into their equivalent first-order equations.

Let,

$$\begin{gathered} y_1' = f_1\left(x, y_1, y_2, y_3, \ldots, y_n\right), \\ \vdots \\ y_n' = f_n\left(x, y_1, y_2, y_3, \ldots, y_n\right). \end{gathered} \tag{2.5}$$

Then, the system of first-order ODEs for (2.5) can also be written as

$$\bar{Y}' = F\left(x, \bar{Y}\right) \tag{2.6}$$

under the initial conditions

$$\bar{Y}\left(x_0\right) = \bar{\xi}.$$

Several options of MATLAB’s ODE solvers are made available for the IVP of ODEs. They can work either on stiff or non-stiff problems. Since some of the problems exhibit stiffness, the execution might be difficult and relatively slow. The type of problem that MATLAB's ODE solvers can efficiently solve must be determined in order to select the appropriate solvers.

The ODE solvers provided by MATLAB are as shown in the following Table 2.1 (for non-stiff problems) and Table 2.2 (for stiff problems):

Table 2.1 MATLAB's ODE solvers for non-stiff problems

Solver	**Accuracy**	**Type**	**Description**
ode45	Medium	Single step method	Medium order method based on an explicit Runge-Kutta (4,5) formula, the Dormand-Prince pair
ode23	Low	Single step method	Low order method which is an implementation of an explicit Runge-Kutta (2,3) pair of Bogacki and Shampine
ode113	Low to High	Multistep method	Variable step, variable-order Adams-Bashforth-Moulton PECE solver of orders 1 to 13

Table 2.2 MATLAB's ODE solvers for stiff problems

Solver	**Accuracy**	**Type**	**Description**
ode15s	Low to Medium	Multistep method	Variable step, variable order method based on the numerical differentiation formulas (NDFs) of orders 1 to 5. Optionally, it can use the backward differentiation formulas (BDFs, also known as Gear's method)
ode23s	Low	Single step method	Low order method based on a modified Rosenbrock formula of order 2
ode23t	Low	Single step method	Trapezoidal rule and can solve differential algebraic equations (DAEs)
ode23tb	Low	Single step method	Trapezoidal rule and Backward Differentiation Formula

2.3 Literature Review

This section will review the previous studies that relate to the current study. The higher-order stiff ODEs problems in this thesis will be solved using the direct block methods. Therefore, this section will review the related studies on the direct methods, block methods, BDF methods, and BBDF methods.

2.3.1 Direct Methods for Solving Higher-Order ODEs

Over the years, scholars have attempted to tackle the higher-order problem directly, thus eliminating the order reduction process. Various approaches have been applied for solving the higher-order ODEs problems directly, either using the non-block or block method. As a result, efforts have been made to explore research on directly solving higher-order ODEs.

One of the pioneers in the study of direct method was Krogh (1968), who proposed the direct integration (DI) method for non-stiff problems by using modified divided difference, where the backvalues of any one of the derivatives are interpolated. Chawla and Sharma (1985) presented independently explicit and implicit Runge-Kutta-Nystrom methods for solving general second-order ODEs. Then, Suleiman (1989) proposed the DI method using the standard divided difference for solving higher-order non-stiff ODEs. Awoyemi (2003) developed a P-stable LMM for general third-order ODEs that was implemented in predictor-corrector mode. Jator (2008) discussed a class of initial value methods (IVMs) for solving second-order IVPs directly. Jator and Li (2009) developed a self-starting LMM for solving the general second-order IVPs directly.

The study on this field has been expanded to date. Then, Rajabi et al. (2016) proposed direct method for solving special third-order ODEs, namely the linear 3- and 5-step methods. Another method introduced by Ghawadri et al. (2019) functions as a direct solver for fourth-order ODE of the structure $u^{(4)} = f\left(x,u,u',u''\right)$. The one-step explicit three-stage of Runge-Kutta of order four (RKTF4) and four-stage of Runge-Kutta of order five (RKTF5) are constructed using the associated B-series and quad-coloured trees theory based on the algebraic order conditions developed in the form of $u^{(4)} = f\left(x,u,u',u''\right)$. Lee et al. (2020) explored a direct approach in handling the general third-order, discussing particularly the special class of explicit two-derivative Runge-Kutta type (STDRKT) methods involving the fourth derivative of the solution, that are two-stages of order five STDRKT2(5) and three-stages of order six STDRKT3(6) methods for solving a particular problem.

From those studies, the direct method has given a benefit when solving the higher-order ODEs which is a reduction in computational effort since the direct method will remove the burden of reducing the higher-order ODEs into a system of first-order ODEs. All the methods mentioned here are non-block methods. There are various block methods that have been proposed for solving the higher-order ODEs directly.

The block method is a method that can provide multiple solutions simultaneously at a time instead of only one solution. Also, it has been proven that block methods are more accurate and require fewer function evaluations, which results in a reduction in the computational cost (Singla et al., 2022). Therefore, there will be another benefit of solving the higher-order ODEs directly using the block method. The problems will be solved more quickly with less computational effort while still producing the desired accuracy. The next subsection will present the reviews on the block method for solving first- and higher-order (second-order and above) non-stiff and stiff ODEs.

2.3.2 Block Methods for Solving First-Order ODEs

Among the earliest to introduce the block method was Milne (1953), who introduced computing the starting values for the predictor-corrector algorithm. Then, Rosser (1967) applied the idea for the Runge-Kutta method where multiple solutions were produced concurrently. Shampine and Watt (1969) and Watanabe (1978) further extended this idea by applying the block method on the one-step method for solving first-order ODEs. Birta and Abou-Rabia (1987) conducted the research on multistep block method. Voss and Abbas (1997) have considered block predictor-corrector scheme of one-step fourth-order block method with variable step size for solving first-order ODEs.

Later, Ibrahim et al. (2007a) discussed the 2-point and 3-point block methods based on the BDF method for solving first-order stiff ODEs. Majid and Suleiman (2009) introduced a 3-point block method with variable step for solving first-order ODEs. Aksah et al. (2016) developed a new block Runge-Kutta method with various weights for solving first-order stiff ODEs. Kashkari and Syam (2019) proposed an optimized one-step hybrid block method for solving general first-order ODEs. Ekoro et al. (2021) developed Adam's block with first and second derivative future points for solving linear and non-linear first-order IVPs in ODEs. Currently, Soomro et al. (2022) designed an optimized hybrid block Adams block method for the solutions of linear and nonlinear first-order IVPs in ODEs.

2.3.3 Block Methods for Solving Higher-Order Non-Stiff ODEs

The study on solving higher-order (second-order and above) ODEs problems was explored by the scholars. They proposed the idea of solving these problems using the block method. There have been a number of scholars who have conducted research on the direct solution of higher-order ODEs using the block method. The first to conduct the study was Fatunla (1991) who discussed the block method for solving second-order non-stiff ODEs directly by proposing zero-stable block method of orders 3 and 4. Ken et al. (2008) developed the explicit and implicit r point block methods with constant coefficients based on Newton-Gregory backward interpolation formula for solving special second-order ODEs.

Then, the solution of second-order ODEs by the block method with variable step size is discussed in Majid et al. (2009). It is suited for the numerical integration of ODEs that are neither stiff nor mildly stiff, and it is presented in the simple form of the Adams-Moulton method. Olabode (2009) derived a 5-step block method of order 7 that solved special third-order ODEs directly which eliminates the use of predictor corrector methods. Awoyemi (2011) introduced a method based on the collocation and interpolation of the power series approximate solution for the direct solution of second-order non-stiff ODEs. Mehrkanoon (2011) proposed the direct variable step three-point block multistep method for solving third-order non-stiff ODEs. The proposed method is based on a pair of explicit and implicit of Adams type formulas.

The research on higher-order ODEs continued with the method proposed by Adeyeye and Omar (2019), known as the implicit five-step block method with generalised equidistant points for treating fourth-order non-stiff ODEs. Allogmany and Ismail (2020) proposed an implicit three-point block method with fourth and fifth derivatives of the solution for solving directly general linear, nonlinear, and applications of third-order non-stiff ODEs. Currently, Abdelrahim (2021) developed four-step implicit hybrid block method for the solutions of the non-stiff fourth-order ODEs. The method was developed using three off-step points, $x_{n+\frac{1}{4}}$, $x_{n+\frac{5}{2}}$, and $x_{n+\frac{7}{2}}$.

In addition, most of the methods that have been developed previously were applied to solve higher-order non-stiff ODEs. In order to solve stiff problems, the BDF method is the most suitable method to be used since it is well-known for its capability to solve stiff problems. Therefore, the next subsection will review the previous studies that applied the BDF method for solving stiff problems.

2.3.4 BDF Methods for Solving Stiff ODEs

The classical BDF method, also known as the Gear method (Gear, 1969), was the earliest method of solving the stiff ODEs problem. The BDF method works best for stiff problems (Oliver, 1982). A number of scholars have made some modifications on the classical BDF method and the some of the methods are listed as follows:

2.3.4(a) Extended Backward Differentiation Formula (EBDF)

Cash (1980) proposed a new method that added one future point called the Extended Backward Differentiation Formula (EBDF) for solving first-order stiff ODEs. The problems considered were predicted using the conventional BDF method and then corrected using the EBDF method. This method is A-stable up to order 4 and A(α)-stable up to order 9. The study also indicated the superiority of the EBDF over certain existing methods through the numerical results.

2.3.4(b) Modified Extended Backward Differentiation Formulae (MEBDF)

Modified Extended Backward Differentiation Formulae (MEBDF) was proposed by Cash (1983). This method was capable of achieving variable step/variable order processes using highly stable formulae of order 2 to 8. When comparing with the BDF method, the MEBDF method required fewer functions, fewer Jacobians, and fewer steps but required more backsolves. However, it managed to provide better accuracy and stability.

2.3.4(c) Adaptive Extended Backward Differentiation Formula (AEBDF)

The Adaptive Extended Backward Differentiation Formula (AEBDF) has been developed by Hojjati et al. (2004). This method is A(α)-stable up to order 9 with a wider angle where its stability region is larger than the EBDF method, and it also gave better accuracy than the EBDF method.

2.3.4(d) Block Backward Differentiation Formula (BBDF)

Then, Ibrahim et al. (2007a) improved the classical BDF method by associating this method with the block method, thus resulting in the BBDF method. Ibrahim et al.'s research was motivated by the need to enhance the well-established solver for stiff ODEs, the Gear method, by speeding up the integration process and also improving the accuracy of the solutions.

2.3.4(e) Hybrid Backward Differentiation Formula (HBDF)

Ebadi and Gokhale (2010) have introduced the BDF method with some off-step points, which is known as the Hybrid Backward Differentiation Formula (HBDF). This method has wider stability regions than the BDF, EBDF, and MEBDF methods, which are A-stable up to order 4 and A(α)-stable up to order 12. Therefore, the HBDF method managed to give more accurate results than comparable methods.

2.3.4(f) Block Extended Backward Differentiation Formula (BEBDF)

Later, Musa et al. (2012) proposed a block method in the form of the EBDF method, which is known as the Block Extended Backward Differentiation Formula (BEBDF). This method is extended from the EBDF method which computes two new solutions concurrently and uses an extra future point, which makes it more advantageous than the conventional BBDF method by producing more accurate solutions.

2.3.4(g) Continuous Block Backward Differentiation Formula (CBBDF)

Akinfewa et al. (2013) introduced a Continuous Block Backward Differentiation Formula (CBDF) that was implemented as a self-starting method and required only one initial value for solving the first-order stiff ODEs problem. As compared to the BDF method, the CBBDF method gave more accurate results and managed to reduce the computational cost with fewer numbers of function evaluations and computational steps.

From the studies, the methods based on BDF were able to solve stiff problems by providing good numerical results. In this study, the idea of the conventional BBDF method was chosen for solving the higher-order stiff ODEs problems since it has proven its superiority of solving the first-, second-, and third-order stiff ODEs through various approaches.

2.3.5 BBDF Methods

Basically, the BBDF method approximates a set of new solutions at each integration step by using the values in the preceding *b*-blocks. This means that every successful integration step will produce *r* new solutions, $y_{n+1},\dots,y_{n+r}$ simultaneously, where each block contains *r*-points. Thus, by proceeding a block at a time, the number of total steps is reduced and a shorter time is needed to complete an integration step instead of producing only a single solution at a time. Also, the BBDF method has the ability to store all the differentiation coefficients to avoid repetitive calculations, which will save computation time. Therefore, the BBDF method is advantageous in terms of computational time and number of total steps in finding the solutions to the problems.

Apart from that, the BBDF method has been extensively used to solve stiff ODEs through various approaches. In relation to this, the efficacy of the BBDF method for solving first- and higher-order stiff ODEs problems will be elaborated in more detail in the following subsections.

2.3.5(a) BBDF Method for Solving First-Order Stiff ODEs

Ibrahim et al. (2007) constructed a set of implicit BBDF formulas known as *r*-point BBDF methods for the solutions of the first-order stiff ODEs where *r* is represented as the block size. The *r*-point BBDF method will produce simultaneously *r* new solutions at the time of discretisation points, $x_{n+1},\dots,x_{n+r}$. They discussed the general approach of computing simultaneously a block of 2 new values (r = 2) and 3 new values (r = 3), using one earlier block with each block containing two and three points. The numerical results showed that the number of total steps was reduced to almost half and one third, and the execution time was faster than non-block BDF method when using the *r*-point BBDF methods.

The Variable Step 2-point BBDF method introduced by Ibrahim et al. (2007b) was the pioneer BBDF method that applied a variation of step size. The step was considered successful if the computed LTE was less than the tolerance value. The value of the step size could be adjusted to 1.6 to gain computation speed or to remain constant. Conversely, the step was considered a failure if the LTE was greater than the tolerance value. Thus, the step needs to be repeated by halving the current value of the step size. When comparing this method to the Variable Step Variable Order Backward Differentiation Formula (BDFVS) method in Suleiman (1979), this method produced better accuracy with less computational time and fewer total steps.

Yatim et al. (2010) introduced a method for finding the solutions of first-order stiff ODEs called the Fifth Order Variable Step Block Backward Differentiation Formulae. A suitable step size was chosen at each step to optimise the performance in terms of precision and computation time. The suitable step size can be either constant, half, or increased to a factor of 1.9. The increment of the step size was inspired by Ibrahim et al. (2007b), where the step size was increased to a factor of 1.9 instead of 1.6. They made a comparison between these increments and found that the increment value of the step size reduced the number of total steps and computed the solution faster.

The research on the BBDF method was extended by adding one back value in the preceding blocks. Four backvalues are used to compute two new solutions in the current block. The resulting method, aimed at solving first-order stiff ODEs, was proposed by Nasir et al. (2011) and known as the Fifth Order Two-Point Block Backward Differentiation Formulas (BBDF(5)). The efficiency of the method was verified by comparing its performance to ode15s in MATLAB and the classic BDF method. The result proved that BBDF(5) was more efficient in both accuracy and execution time, where it converged faster than the other two methods.

The existing BBDF method proposed by Yatim et al. (2013a) was enhanced by research on the most efficient strategy for selecting the step size and order of the method. The strategy was applied to the BBDF method for solving first-order stiff ODEs. Fundamentally, each order of the BBDF method (orders three, four, and five) had a variety of values for step size. The method is known as the Variable Step Size and Variable Order Block Backward Differentiation Formula (VSVO-BBDF). The strategy for selecting the step size focused on reducing the number of iterations and computation time. Compared to ode15s and ode23s in MATLAB, VSVO-BBDF yielded the shortest execution time, reduced the number of total steps, and produced the best solutions.

Abasi et al. (2012) derived a new formula to approximate three new solutions in the current block, known as the Variable Step Size 3-Point Block Backward Differentiation Formulas (3-BBDF) for the solution of the first-order stiff ODEs. The considered step size includes constant step size, halving the step size, and increasing the step size by a factor of 1.196. In comparison with both MATLAB's suite ODEs solvers, ode15s and ode23s, the 3BBDF method performed better in reducing the error and also converged faster for all the considered problems.

Currently, Nasarudin et al. (2020) made a modification to the classical BBDF method by adding two off-step points, which is known as the Block Backward Differentiation Formula with off-step points of order 6, BBDFO(6) for solving the first-order stiff ODEs. The stability region of the proposed method is larger as compared to the order six BBDF method without the off-step points, BBDF(6), that was proposed by Nasir (2012). The numerical experiments were conducted to validate the efficiency of the BBDFO(6) by comparing its performance with that of the BBDF(6) and ode15s. The numerical results proved that the BBDFO(6) was found to produce better accuracy at certain step sizes than the other two methods.

Therefore, it would be preferable to find the solutions of the higher-order stiff ODEs using the direct method, such as BBDF, since it is an implicit method that is suitable for solving the stiff problems.

2.3.5(b) BBDF Method for Solving Second-Order Stiff ODEs

Ibrahim et al. (2009) developed the Variable Step Block Backward Differentiation Formulas as a direct method for solving second-order stiff ODEs known as BBDF2. They considered the value of the step size ratio, $r = 1$ (constant step size), 2 (half of the step size), and 5/8 (increment of the step size by a factor of 1.6). It was proven that the BBDF2 successfully solved the considered problems since this method executed the solutions of the second-order ODEs directly. The results obtained showed a reduction in the number of total steps and execution time as compared to the non-direct BDFVS method developed by Suleiman (1979). Also, the proposed method was able to improve the accuracy of the solutions.

In addition, Zainuddin (2011) proposed 3rd, 4th, and 5th Order Direct Block Backward Differentiation Formula, known as 2BBDF(3), 2BBDF(4), and 2BBDF(5), respectively, for solving second-order stiff ODEs. The method was developed using a fixed step size approach. The performance of the proposed method was validated by comparing with established solvers for solving stiff problems, namely ode15s and ode23s. The results confirmed that the proposed method is more efficient than the