# IMPLICIT RATIONAL RUNGE-KUTTA SCHEMES FOR STIFF ORDINARY DIFFERENTIAL EQUATIONS

BY

BABATOLA, PATRICIA OLAYINKA

A THESIS IN THE DEPARTMENT OF
INDUSTRIAL MATHEMATICS AND COMPUTER
SCIENCE SUBMITTED TO SCHOOL OF
POST GRADUATE STUDIES TOWARDS A PARTIAL
FULFILMENT OF THE REQUIREMENT
FOR THE AWARD OF THE DEGREE OF
MASTER OF TECHNOLOGY
(M. TECH) IN INDUSTRIAL MATHEMATICS OF
FEDERAL UNIVERSITY OF TECHNOLOGY,
AKURE, NIGERIA.

DECEMBER, 1999

## CERTIFICATION

This is to certify that this project was carried out by Mrs Babatola, Patricia Olayinka of Industrial Mathematics and Computer Science Department, Federal University of Technology, Akure, Nigeria.

DR. R.A. ADEMILUYI
SUPERVISOR

3/01/2000
DATE

DR. S.T. ONI
CHIEF EXAMINER

12/01/2000
DATE

PROF. A.U. AFUWAPE
EXTERNAL EXAMINER

17/1/2000
DATE

## DEDICATION

This work is dedicated to the ALMIGHTY GOD, My late Parents, Mr and Mrs Olatunkunbo Sanwo, my husband and children.

## ACKNOWLEDGEMENT

I wish to express my profound gratitude to those who directly or indirectly influenced the successful completion of this research work.

I must express my ineffable gratitude to my supervisor, DR. R.A. ADEMILUYI who gave his books and ideas freely. He equally devoted his precious time to reading through the project and made valuable corrections and suggestions. Also to my co-supervisor and head of Department DR. S.T. ONI whose help guidance and instruction have contributed in no small measure to successful completion of this work.

My special thanks goes to the immediate past Head of Department, DR. S.O. FALAKI for his efforts towards the success of this project.

My appreciation goes to DR. O.C. AKINYOKUN, DR. O.K. KORIKO, MR. O. OLOTU, MR. F.A. ALAO, MR. E.O. OLOWOFESO and other colleagues in the Department for their support during the course of my programme.

I must not fail to mention the immense contribution of my dearly beloved husband ENGR. J.O. BABATOLA who stood by me both morally, financially, materially and prayerfully from the beginning to the end of this programme. I give glory to GOD for him.

To my other friends who contribute to the success of this programme, I say thank you all.

Finally, above all I thank the ALMIGHTY GOD who through his grace and mercy made my dream a reality.

# TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## ABSTRACT

In this thesis, families of Implicit Rational Runge-Kutta Schemes are developed, analysed and computerised to solve stiff ordinary differential equations.

The method is motivated by the conventional Runge-Kutta Schemes and rational function approximation. While its development and analyses make use of Taylor series expansion and Pade's approximation techniques respectively. These schemes are convergent and A-stable.

Numerical results showed that the schemes are effective and efficient. Their development are however tedious.

Possible improvement strategies are suggested. Their use as multipurpose codes are feasible but will require further studies.

## CHAPTER ONE

### INTRODUCTION

There are many processes in the field of science, management and technology which involve rate of change of one variable in relatiion to another.

In mathematics, rate of change are called derivatives. Any equation which connects the derivatives of a differentiable function of one independent variable with respect to itself is called Ordinary Differntial Equation (ODE).

The most general form of an ordinary differential equation is

$$f(x, y, y', y'' \ldots\ldots\ldots\ldots y^{(n)}) = 0 \ldots\ldots\ldots\ldots \quad (1.1)$$

where y is the dependent variable,and n is the order of its derivatives.

The order of the differential equation is the order of the highest derivative and its degree is the exponent or power to which the highest derivatives is raised.

If no product of the dependent variable y(x) with itself or any of the derivative that occur, then the equations is linear, otherwise it is non-linear.

Usually equation (1.1) is expressed in the form

$$y^{(n)} = f(x, y, y' \ldots\ldots\ldots\ldots y^{(n-1)}) \ldots\ldots\ldots\ldots \quad (1.2)$$

called canonical representation.

Differential equation (1.1) together with initial conditions (1.3) is called initial value problems (I.V.P). Thus an $n^{th}$ order initial value problem is of the form

$$f(x, y, y', y'' \ldots\ldots\ldots y^{(n)}) = 0;$$

$$y^{i}(x_o) = \eta_i, \; i = 1(1)n-1 \ldots\ldots\ldots\ldots \quad (1.3)$$

Any $n^{th}$ order initial value problem in ordinary differential equation of the form (1.3) can be reduced to vector form

$$y' = f(x, y), \; y(x_o) = \eta \qquad (1.4)$$

where

$\eta = (\eta_o, \eta_1, \eta_2 \ldots\ldots \eta_{n-1})^T$

$f = [f_1, f_2 \ldots\ldots f_n]^T$

$y = [y_1, y_2 \ldots\ldots y_n]^T$

In attempting to solve this, it will be assumed that f(x,y) satisfies the following conditions.

(i) f(x,y) is a real value vector function

(ii) f(x,y) is defined and continous in the region D of x, y- plane defined by:

$$D = \{(x,y)/a \leq x \leq b, -\infty < y < \infty\} \qquad (1.5)$$

and contains initial point $(x_o, y_o)$.

(iii) There exist a real constant L such that for any $x \in [a,b]$ and numbers $y_1$ and $y_2$ in D

$$|f(x,y_1) - f(x, y_2)| \leq L\, |y_1 - y_2| \qquad (1.6)$$

where L is the Lipschitz constant of order 1

Thus, for any $y_o \in D$, the initial value problem (1.4) satisfying (i) - (iii) has a unique solution y(x) for $x \in$ [a, b] (Apostol 1965)

If conditions (i) and (ii) are satisfied and the partial derivative $f_y$ are continuous and bounded in D, then the Lipschitz constant L of the system may be taken as.

$$L = \left\| \frac{\partial f}{\partial y} \right\| \qquad (1.7)$$

Systems of initial value problem in ordinary differential equations can be classified into stiff if its eigen values $\lambda j$'s are widely separated and non stiff otherwise. We will soon observe that it plays an important role in the development of the numerical methods for solution of ordinary differential equations.

The features of the differential equation (1.4) that require consideration is the phenomenon called stiffness.

In this thesis, we shall be mostly concerned with ordinary differential equations that are stiff.

This class of ordinary differential equations arises in a large variety of application areas such as pharmaco-kinetic theory, heat and matter transfer, chemical and nuclear reactions, electrical transmission network, guidance and control problems and other dynamic process in industries.

The stiff nature of ordinary differential equations will be considered in the next section.

## 1.1 NATURE OF STIFF ORDINARY DIFFERENTIAL EQUATIONS

Stiffness as it is used in the context of ordinary differential equations, is that it is a concept describing the nature of certain subset of ordinary differential equations whose solutions contain components with fast and slow responses.

The fast responding components are called transients while the slow responding components are generally smooth and steady.

The stiff systems were first encountered by Hirchfelder(1952) in the study of the motion of the masses - spring system of varying stiffness from where the problem derives its name.

Stiffness is quite related to the structure of the Jacobian matrix J(x) with the element defined by

$$J_{ij}(x) = \frac{\partial f_i}{\partial y_j} \quad i, j = 1(1)n \qquad (1.8)$$

If $\lambda_j$ are the the eigen values of J with property $Re(\lambda_j) < 0$ and it is large in absolute sense, then the system is either stiff or stiffly oscillatory.

A differential equation of the form (1.4) whose Jacobian possesses eigen values

$$\lambda_j = u_j + iv_j, \quad j = 1(1)n \qquad (1.9)$$

where $i = \sqrt{-1}$, satisfying the following conditions

i $\quad u_j << 0, \quad j = 1(1)n$

ii $\quad \text{Max } |u_j(x)| >> \min|u_j(x)|$

$$\text{or } r(x) = \frac{\text{Max } |u_j(x)|}{\text{Min}|u_j(x)|} >> 1 \qquad (1.10)$$

where r(x) is the stiffness ratio.

iii $\quad u_j(x) << v_j(x), \; 1 \le j \le n$

is stiff oscillatory.

System (1.4) with conditions (i) and (ii) only, is stiff. In this case, condition (i) ensured that the system is stable while (ii) indicates that systems possesses some components which decay very rapidly. For example,

**Model 1**

$$y' = \lambda(y - E(x)) + E'(x), \; y(x_o) = y_o \qquad (1.11)$$

where E(x) is continously differentiable, $\lambda$ is a complex constant with $Re(\lambda) << 0$.

The theoretical solution of this problem (1.11) is

$$y(x) = E(x) + y_o e^{\lambda x} \quad \text{.......... (1.12)}$$

where E(x) is slowly varying in the interval of integration $[x_o, b]$, while the second components decay rapidly inthe transient phase with the order of $-1/\lambda$.

**Model 2**

$$y' = \begin{bmatrix} -0.00005 & 100 \\ -100 & -0.00005 \end{bmatrix} y \quad \text{.......... (1.13)}$$

with

$y(o) = [\,1,\ 1]^T,\ 0 \le x \le 10\,\pi$

whose solution is obtained as

$$y(x) = e^{-0.00005x} \begin{bmatrix} \text{Sin } 100x + \text{Cos } 100x \\ \text{Cos } 100x - \text{Sin } 100x \end{bmatrix} \quad \text{.......... (1.14)}$$

The transitory phase for this problem is the entire interval of integration $0 \le x \le 10\pi$ with $50/\pi$ complete oscillation per unit interval. This model is also considered to be a highly oscillatory problem.

This class of ordinary differential equations will be considered in this thesis.

## 1.2 PROBLEMS ASSOCIATED WITH STIFF ORDINARY DIFFERENTIAL EQUATIONS

The problems associated with numerical solutions of stiff systems were first recognised by Curtis and Hirschfelder (1952).

One of the basic problem is the instability property exhibited by numerical solution of stiff ordinary differential equations.

For stability of numerical solution of stiff ordinary differential equations, it is necessary to adopt a step size h such that every one of the product

$\lambda_j h$, $j = 1(1)n$, shall lie within the region of absolute stability of the numerical integrators.

Other requirements include the necessity for the numerical schemes to be either A-stable, stiffly stable, $A(\alpha)$ -stable or A (0)-stable. All these concepts are fully explained in chapter four of this thesis.

These stability criteria require that the numerical schemes must be implicit (Dalquist,1963). This implicitness introduces the problems of solving at each integration step, a set of simultaneous non-linear equations and this requires a lot of rigorous iterative process and high computations.

## 1.3 EXISTING METHODS FOR STIFF ORDINARY DIFFERENTIAL EQUATIONS

As we have discussed in the last section, the difficult nature of the solution process for stiff ordinary differential equations has made researchers generate a lot of interest.

The existing methods include the backward differentiation formular (B.D.F).

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k} \ldots\ldots\ldots\ldots\ldots 1.15)$$

proposed by Gear (1969), where $y_{n+j}$ is $(n + j)^{th}$ approximation to the solution.

In 1971, Gear implemented BDF of orders one to six jointly with Adam Moulton Formular (AMF) of order one to twelve (1 - 12) in a variable order variable step size and called it DIFSUB.

DIFSUB is known to be one of the most robust algorithm for stiff initial value problems. However, it runs into difficulty when the eigen values of the Jacobian $\partial f/\partial y$ are close to imaginary axis (i.e. stiff oscillatory problem).

Several attempts have been made to improve this codes. Some of the well known versions are GEAR by Hindmarsh (1974) and EPISODE by Byne and Hindmarsh (1975).

These inadequacies in BDF codes, motivate a search and development of other numerical methods.

These methods are hopefully to prove more superior particularly in the area where Backward differentation methods appear to have failed. To overcome the stability problems for BDF, second derivative formular (SDF) was proposed by Enright (1972) and this is given as

$$y_{n+k} - y_{n+k-1} = h\sum_{j=0}^{k} \beta_j f_{n+j} + h^2\beta_{n+k} f_{n+k} \quad ..(1.16)$$

These are A-stable up to order four and $A(\alpha)$ stable up to order nine.

Enright (1974) adopted a variable step size, variable order implementation of (1.16) using a one step, two step error estimates restricting step size to halving and doubling in a code called SDBASIC. It is observed that the code performed much better than DIFSUB on stiff systems whose eigen values have large imaginary parts. However, its performance on stiff oscillatory problem is nothing to write home about. This perhaps motivated Ademiluyi (1987) to propose a variant of second derivative multistep formular in the form

$$y_{n+k} = y_{n+k-1} + h\, B_k y_{n+k} + h^2 \sum_{j=0}^{k} \gamma_j y_{n+j} \ldots\ldots\ldots\ldots\ldots\ldots (1.17)$$

which are A-stable for k ≤ 10 and stiffly stable for sufficiently large k.

Other existing schemes are conventional s-stage Runge-Kutta Schemes

$$y_{n+1} = y_n + \sum_{i=1}^{s} W_i K_i \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.18)$$

where

$$K_i = hf(x_n + a_i h, \quad y_n + \sum_{j=1}^{i} b_{ij} k_j)$$

proposed by Kutta (1901) and improved by Runge (1915).

This family of methods is normally classified into three categories namely:

i. Explicit : When matrix B = {$b_{ij}$} = 0 for j ≥ i., B is a lower triangular matrix.

ii. Semi-Implicit : B = {$b_{ij}$} = 0 for j > i. That is B is a lower triangular matrix with non-zero diagonal elements.

iii. Implicit: B = {$b_{ij}$} ≠ 0, for at least one j > i. That is B is not a diagonal matrix.

Other people who have contributed to the development of the conventional Runge-Kutta Schemes includes:

Cash (1983) who proposed class of variable order Implicit Runge-Kutta methods for stiff initial value problems in ODES. The code based on this problem have been found to perform satisfactorily.

Enright (1975) observed that the only problem associated with Implicit Runge-Kutta formular codes is that the amount of linear algebra involved is too large and that the code based on them cannot be expected to be efficient.

Later, Burrage, Butcher and Chipman (1979) implementation of the family of Singly Implicit Runge-Kutta (SIRK) formular led to the development of a code called STRIDE which are A($\alpha$) stable with $\alpha \geq 83°$.

Gafney (1984) established that the SIRK code are only competitive with BDF for stiff oscillatory equations.

Norsett (1974) and Alexander (1977) proposed the diagonally Implicit Runge-Kutta (DIRK) formular with $\lambda$ as the non-zero diagonal element of the matrix B.

Kaps and Kentrops (1979) developed new codes based on Rosenbrock formular GRK - 4A and GRK - 4AT which are A-stable. The performance test by Addison (1980) showed that the codes are reliable.

However, due to small region of absolute stability and high computation cost, conventional Explicit Runge-Kutta Schemes were ignored among stiff ODES integrators.

Perhaps, their ease of computerisation motivated Fatunla (1982) to propose a one-step scheme.

$$y_{n+1} = \frac{y_n^2}{y_n + hf_n} \ldots (1.19)$$

called Inverse Euler Formular for stiff ordinary differential equation. The performance of this scheme was very satisfactory.

However, in the same year Hong Yuan Fu (1982) proposed a more general form of this scheme and called it Explicit Rational Runge-Kutta Scheme. This general form of the scheme is given by

$$y_{n+1} = \frac{y_n + \sum_{i=1}^{r} W_i K_i}{1 + y_n \sum_{i=1}^{s} V_i H_i} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (20)$$

where $K_1 = hf(x_n, y_n)$

$$k_i = hf(x_n + C_{i-1}h, y_n + \sum_{j=1}^{i} a_{i-1j} K_j), \; i = 1(1)r$$

$$H_1 = hg(x_n, z_n)$$

$$H_i = hg(x_n + d_{i-1}h, Z_n + \sum_{j=1}^{i} b_{i-1j} H_j)$$

$g(x_n, Z_n) = -Z_n^2 f(x_n, y_n)$.

and $Z_n = 1/y_n$

In his development; $a_{ij} = 0$, $b_{ij} = 0$ for $j \geq i$. He developed families of methods of orders two and three of these schemes.

During analysis of these schemes, he discovered that the schemes are A - stable. This property stimulated him to use it to solve some stiff ODEs especially widely stiff equations.

Perhap, this A - stability property and simplicity of programming of Explicit Rational R - K Schemes stimulated Okunbor (1985) in extendidng the schemes to family of order four.

From Okunbor's work, it is observed that the higher the stage of the methods, the poorer the stability. Their performance on stiff ODES is satisfactory, but their performance on stiff Oscillatory problem is nothing to write home about. However, experience with the conventional R-K Schemes have shown that implicit R-K scheme have better resolution properties (than explicit ones). This expectation is the chief mover of the present consideration.

## 1.4 AIM AND OBJECTIVES

Owing to superior resolution properties of implicit methods on stiff equations, we will consider in this thesis, the development, analysis and implementation of a family of implicit rational Runge-Kutta scheme.

In particular, we will consider one - stage of order two, two - stage of order three and four for solving stiff initial value problems in ODEs.

## 1.5 MOTIVATION

The main motivation in the implicit rational R - K scheme lies in their stability characteristics which are more superior to those of explicit schemes.

Their large region of absolute stability (RAS) and high resolution property with respect to problem with stiff oscillatory and those with singularity make them particularly suitable for stiff Ordinary differential equations.

## 1.6 RESEARCH METHODOLOGY

To achieve the above objectives, we shall adopt power series algorithm (Taylor and Binomial) to generate the parameters of the implicit rational R-K Schemes.

Analysis of the convergence and stability properties will be carried out by adopting the Pade's approximation, Richardson extrapolation techniques and Felhberg error control method.

The workability and accuracy of the schemes would be confirmed by carrying out its implementation using sample problems on a microcomputer.

The remaining part of the thesis will be organised as described below.

In Chapter two, the general principle of one - step schemes are discussed because they form the basis of the proposed schemes.

Chapter three deals with the development of the proposed schemes. In particular, one stage scheme of order two and two stage scheme of order three and four methods are considered.

Chapter four analyses and discusses the error, convergence and stability properties as they are related to the proposed schemes.

Chapter five considers the computer implementation of the methods with sample problems and results recorded in tables.

Finally, chapter six summaries the whole thesis, and gives some general remarks on the results of the proposed schemes.

# CHAPTER TWO

## PRELIMINARY CONCEPTS AND PRINCIPLES

### 2.1 PRINCIPLES OF ONE - STEP SCHEMES

Since the proposed schemes are based on one step methods, it is necessary to discuss some of these techniques, particularly the conventional implicit Runge - Kutta schemes which form the basis of the proposed schemes.

A general one - step schemes for solution of the differential equation (1.4) is the one in which the approximation $y_{n+1}$ to the solution at point $x_{n+1}$ can be calculated from the knowledge of $y_n$ at $x_n$.

In general one - step schemes are written in the form

$$y_{n+1} = y_n + h\phi(x_n, y_n; h) \ldots\ldots\ldots\ldots (2.1)$$

where $\phi(x, y; h)$ is a function of the arguement x, y, h and in addition depends on the right hand side of equation (1.4). The function $\phi(x, y; h)$ is called the increament function.

Families of one step schemes include

I. Euler Scheme

$$y_{n+1} = y_n + hf(x_n, y_n) \ldots\ldots\ldots (2.2)$$

ii. The Taylor series scheme

$$y_{n+1}=y_n+\sum_{r=1}^{n} \frac{h^r}{r!} y_n^{(r)} \ldots\ldots\ldots (2.3)$$

iii. The Conventional S-Stage Runge - Kutta Schemes

$$y_{n+1}=y_n+\sum_{i=1}^{s} W_i K_i \ldots\ldots\ldots (2.4)$$

where

$$K_i=hf(x_n+a_i h, y_n+\sum_{j=1}^{i} b_{ij} k_j) \ldots\ldots\ldots\ldots (2.5)$$

with the constraints

$$C_i=\sum_{j=1}^{i} b_{ij} \ldots\ldots\ldots\ldots\ldots\ldots (2.6)$$

And

$$\sum_{j=1}^{i} W_i=1 \ldots\ldots\ldots\ldots\ldots (2.7)$$

Because this family of one step method forms the basis of our proposed scheme, we will limit our discussion of principles of one step method on it.

## 2.2 CONVENTIONAL RUNGE - KUTTA SCHEME

Runge - Kutta Schemes is one of the oldest numerical method for solution of Ordinary Differential Equations (ODEs).

These schemes were proposed by Kutta (1901) and later improved by Runge (1915).

An s-Stage Runge Kutta scheme is defined in (2.4). These schemes are often divided into three classes, namely.

i. Explicit : $B = \{b_{ij}\} = 0$ for $j \geq i$

ii. Semi-Implicit : $B = \{b_{ij}\} = 0$ for $j > i$.

iii Implicit : $b_{ij} \neq 0$ For at least $j \geq i$.

The numerical values of the unknown coefficients $c_i$ , $w_i$, $b_{ij}$ are normally obtained from sets of non - linear equations generated by adopting the follow steps below :

I. Taylor series expansion of $k_i$ about point $x_n$ for $i = 1\ (1)s$,

ii. Insertion of the results of the expansion into (2.4)

iii. Comparism of the final expansion with the Taylor series expansion of $y_{n+1}$ about $x_n$ in powers of h.

The numbers of parameters normally exceed the numbers of equations, therefore, some parameters are chosen, so as to satisfy some of the following conditions.

i. Minimization of bound of the local truncation error (Raltson, 1962).

ii. Maximization of the attainable order of the scheme (King, 1866).

iii. Optimization of the interval of absolute stability (Lawson 1966, 1976).

iv. Attainment of optimal storage space (Gill, 1951)

Butcher (1854) showed that s-stage implicit R - K schemes of orders 2 and 5 are all A-stable (Properties to be discussed later).

These stability properties make implicit R - K methods more suitable for solving stiff problems than any other one - step schemes.

Some popular Runge - Kutta schemes are :

i. The implicit Euler scheme

$$y_{n+1} = y_n + hk_1 \ldots\ldots\ldots\ldots (2.8)$$

where

$$k_1 = f(x_n + h, y_n + hk_1)$$

ii. Two stage trapezoidal scheme of order three

$$y_{n+1} = y_n + \frac{h}{2}[k_1 + k_2] \ldots\ldots\ldots\ldots (2.9)$$

where

$K_1 = f(x_n + h, y_n + k_1)$.

$K_2 = f(x_n + h, y_n + k_2)$.

iii. Two - stage implicit R - K scheme of order four as defined by Harmmer and Hollingsworth (1955)

$$y_{n+1} = y_n + \frac{h}{2}[k_1 + k_2] \ldots\ldots\ldots\ldots (2.10)$$

where

$$K_1 = f\left(x_n + \left(\tfrac{1}{2} - \frac{\sqrt{3}}{6}\right)h,\ y_n + \frac{1}{4}K_1 + \left(\frac{1}{4} - \frac{\sqrt{3}}{6}\right)K_2\right).$$

$$K_2 = F\left(x_n + \left(\tfrac{1}{2} + \frac{\sqrt{3}}{6}\right)h,\ y_n + \left(\tfrac{1}{4} + \frac{\sqrt{3}}{6}\right)hk + \tfrac{1}{4}hk_2\right).$$

iv. Three - stage implicit Runge-Kutta scheme defined by Butcher (1964).

$$y_{n+1} = y_n + \frac{h}{18}[5k_1 + 8k_2 + 5k_3] \quad \ldots\ldots\ldots (2.11)$$

Where

$$K_1 = F\left(x_n + \left(\tfrac{1}{2} - \frac{\sqrt{15}}{10}\right)h,\ y_n + \frac{5}{36}hk_1 + \left(\frac{2}{9} - \frac{\sqrt{15}}{15}\right)k_2 + \left(\frac{5}{36} - \frac{\sqrt{15}}{30}\right)hk_3\right].$$

$$K_2 = F\left(x_n + \tfrac{1}{2}h,\ y_n + \left(\frac{5}{3} - \frac{\sqrt{15}}{34}\right)hk_1 + \frac{5}{3}hk_2 + \left(\frac{2}{9} + \frac{\sqrt{15}}{34}\right)k_3\right).$$

$$k_3 = F\left(x_n + \left(\tfrac{1}{2} + \frac{\sqrt{15}}{10}\right)h,\ y_n + \left(\frac{5}{36} + \frac{\sqrt{15}}{30}\right)hk_1 + \left(\frac{2}{9} + \frac{\sqrt{15}}{18}\right)hk_2 + \frac{5}{36}hk_3\right).$$

The reliability of numerical one - step schemes is measured in terms of its ability to control the errors it generates per step.

In the next section, we will discuss the nature of the errors which frequently affect computer implementation of this class of algorithms for solving stiff ordinary differential equations.

## 2.3 ITS ERROR ORDER, CONVERGENCE AND STABILITY PROPERTIES

One of the major and very significant aspect of numerical scheme is its ability to reliably control the global error

$$e_{n+1} = y_{n+1} - y(x_{n+1}) \quad \ldots\ldots\ldots (2.13)$$

where $y_{n+1}$ is the numerical solution at step $x_{n+1}$ and $y(x_{n+1})$ is the theoretical solution.

A general requirement is that this errror be made as small as possible by making h sufficiently close to zero, this is one of the concept of convergence.

To make this error concept clearer, we give the following definitions:

**Definition 1:**

The local truncation error $T_{n+1}$ associated with one - step schemes (2.2) is defined as

$$T_{n+1} = y(x_{n+1}) - y(x_n) - h\phi(x_n, y(x_n);h) \quad \text{.......} \quad (2.14)$$

and it is the amount by which the theoretical solution of initial value problem (1.4) fails to satisfy the difference equation (2.1).

There is obviously a relationship between the global error defined in (2.12) and local truncation error defined in (2.13). The relationship is

$$|e_{n+1}| \leq K\,|T_{n+1}| \quad \text{.......} \quad (2.15)$$

where K is a constant. From (2.14), it can be seen that the local truncation error is directly proportional to the global error introduced at each step, particularly when the derivation and computation of the local truncation error is rigorous and all previous solutions are exacts.

This establishes the linear convergence of one - step schemes which leads to the following definitions.

**Definition 2**

One step - scheme of type (2.1) is said to be convergent, if when applied to initial value problem of type (1.4) and the corresponding approximation $y_n$ obtained satisfies,

$y_n \to y(x_n)$ as $n \to \infty$ Or $\lim_{n \to \infty} y_n = y(x_n)$

**Definition 3**

One - step schemes (2.1) said to be of order p, if p is the largest positive integer such that the local truncation error $T_{n+1}$ satisfies

$$T_{n+1} = O(h^{p+1}) \quad \ldots\ldots\ldots\ldots\ldots\ldots \ldots\ldots (2.16)$$

when $O(h^{p+1})$ implies the existence of finite constant c and h >o such that

$$T_{n+1} = C(h^{p+1}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.17)$$

as $h \to 0$.

**Definition 4**

One - step schemes (2.1) is said to be consistent if

$$\phi(x,y;0) = f(x, y) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.18)$$

Where $\phi(x, y; h)$ is the increment function as define in equation (2.1).

The consistency of One - Step numerical scheme ensure that the scheme is at least of order one. That is $p \geq 1$.

**Definition 5**

One - step scheme (2.1) is said to be regular if function $\phi(x, y; h)$ is defined and continuous in the domain $a \leq x \leq b$, $|y| < \alpha$ and $0 \leq h \leq h_0$, ($h_0$ is a positive constant) and if there exist a constant L such that

$$|\phi(x,y;h) - \phi(x,z;h)| < L|y - z) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.19)$$

for every $x \in [a, b]$ and $|z| < \alpha$, $h \in [0,h_0]$

A theorem which guarrantes the convergent of one-step scheme is stated here without proof.

**Theorem 1**

A necessary and sufficient condition for convergence of a regular one - step scheme is that it must be consistent.

A numerical solution to an initial value problem in ordinary differential equation will be said to be accurate, if its numerical result does not deviate significantly from the corresponding values of the exact solution while a numerical solution that is not stable is said to be unstable.

Instability exist in various forms but there are two basic types, namely, the inherent instability and the induced instability.

Inherent instabillity is a property of the differential equation itself, while the induced instability is the characteristic of the numerical methods.

To explain this concepts, we consider the following scalar initial value problem

$$y' = \lambda y,\ y(x_0) = y_0 \quad \text{.......} \quad (2.20)$$

with $Re(\lambda) < 0$ over the interval $a \le x \le b$; its theoretical solution is

$$y(x) = y_0 e^{\lambda x} \quad \text{.......} \quad (2.21)$$

Now suppose we slightly change the initial condition (2.20) by $\beta > O$, so the initial condition becomes

$$y\,(x_0) = y_0 + \beta \quad \text{.......} \quad (2.22)$$

Then the solution of (2.20) together with condition (2.22) yields

$$y(x) = (y_0 + \beta\,)e^{\lambda x} \quad \text{.......} \quad (2.23)$$

Irrespective of the integrations scheme used, for $h > O$, no matter how small, it can be seen that the second term $\beta e^{\lambda x}$ in (2.22) eventually grow exponentially as the computation proceeds, if $Re(\lambda) > O$. Thus the solution

$$y(x) = y_0 e^{\lambda x} \quad (2.24)$$

will become unstable for slight change in the initial condition.

This kind of instability is called inherent, the differential equation may be stable, while the numerical scheme gives unstable solution. This class of instability is termed induced instability. It arises as a result of adopting finite instead of infinite iteration process in our computer implementation.

This is normally detected by applying the integration scheme to solve the scalar test equation (2.20).

The instability will show up inform of a spurious exponentials and it is minimized by reducing the step size. This stability of one step scheme can be defined in terms of global error. The following definition give the clearer concept.

**Definition 6.**

One - step scheme is said to be stable, if for any initial error $e_0$, there exist a constant K and $h_0 > 0$, such that when (2.1) is applied to initial value problem (1.4) with step size $h \in [0, h_0]$ the ultimate error $e_n$ satisfies the following inequality.

$$e_n \leq ke_o, \quad k < \alpha \quad (2.25)$$

The effect of induced instability associated with solution of ordinary differential equation is particularly strong for stiff ordinary differential equations, we can then expect difficulty, if we attempt to solve it by ordinary methods.

One - step schemes will be absolutely stable for a given step size and for initial value problem (1.4), if the errors tends to zero as the step size decreases.

The absolute stability property of one - step scheme (2.1) can be investigated by applying this to scalar test problem (2.20) which yields the following first order difference equation.

$$y_{n+1} = \mu(\hbar)y_n, \quad \hbar = \lambda h \qquad (2.26)$$

And $\mu(\hbar)$ is called the stability function of one - step schemes which can either be polynomial or rational function.

For example stability function for an S-stage implicit Runge - Kutta schemes of order 2s is given as

$$\mu(\hbar) = I + \hbar W^T(I - \hbar A)^{-1}e \qquad (2.27)$$

where $A = \{a_{ij}\}$, $i, j = 1(1)s$

$e = [1, 1 \ldots\ldots\ldots 1]^T$

$W = [W_1, W_2 \ldots\ldots\ldots W_s]^T$

and I a unit S x S matrix.

The relatively large region of absolute stability of implicit Runge - Kutta Scheme makes them suitable for stiff and stiff oscillatory problems in ordinary differential equations.

The stability properties of the proposed schemes are to be discussed in broader sense in chapter four of this thesis.

# CHAPTER THREE

## THE PROPOSED SCHEME

### 3.1 RATIONAL RUNGE-KUTTA SCHEMES

Rational functions are quotients of polynomials and so constitute a much richer class of functions. Their response to problems with singularities make them principal targets for function approximation.

This property of rational funcitons perhaps motivated Hong Yuan Fu (1982) to propose rational scheme (1.20) with general form

$$y_{n+1} = \frac{y_n + \sum_{i=1}^{r} W_i K_i}{1 + y_n \sum_{i=1}^{s} V_i H_i} \ldots\ldots\ldots\ldots 3.1$$

where

$$K_i = hf(x_n + C_i h, y_n + \sum_{j=1}^{i} a_{ij} k_j), i=1(1)r$$
$$H_i = hg(x_n + d_i h, Z_n + \sum_{j=1}^{i} b_{ij} H_j), i=1(1)s$$
$$g(x_n, Z_n) = -Z_n^2 f(x_n, y_n) \text{ and } Z_n = 1/y_n \ldots\ldots (3.2)$$

which form the basis of the proposed scheme.

From the experience of conventional Runge-Kutta schemes, we discover that Rational R - K scheme (3.1) can be classified into three categories, namely: Explicit, Semi-Implicit and Implicit family of methods

(I) <u>Explicit</u>

The method is explicit when A = $\{a_{ij}\}$, i,,j = 1(i)s and B = $\{b_{ij}\}$ i,,j = 1(i)r are zero for $j \geq i$. In which case A and B are lower triangular matrices with zero diagonal elements. These represent the Yuan's family of methods.

(ii) Semi - Implicit

It is semi-Implicit, if $A = \{a_{ij}\}$, $I,j = 1(i)s$ and $B = \{b_{ij}\}$, $I,j = 1(i)r$ are zero for $j>i$. In this case A and B are bounded diagonal matrices with non-zero diagonal elements.

(iii) Implicit

The method is Implicit, if the elements of matrices A and B are different from zero for at least one $j>i$. These implies that A and B are upper triangular matrices. These class of rational Runge-Kutta will be the object of this thesis.

Yuan developed explicit families of one-stage, two stage and three stage of orders one, two and three respectively.

The one stage scheme of order one is of the form

$$y_{n+1} = \frac{y_n^2}{y_n + hf_n} \quad \ldots\ldots\ldots\ldots (3.3)$$

which incidentally coincide with Fatunla (1982) inverse Euler method.

A two stage formular of order two which was obtained by setting r = s =2 in (3.1) is given by:

$$y_{n+1} = \frac{y_n + \frac{1}{4}(K_1 + K_2)}{1 + \frac{y_n}{4}(H_1 + H_2)} \quad \ldots\ldots (3.4)$$

where

$K_1 = hf(x_n, y_n)$

$K_2 = hf(x_n + h, y_n + hf_n)$.

$H_1 = hg(x_n, z_n)$.

$H_2 = hg(x_n + h, y_n + H_1)$.

while the three stage scheme of order three is given by;

$$y_{n+1} = \frac{y_n + \frac{1}{2}(K_1 + 4K_2 + K_3)}{1 + y_{\frac{n}{12}}(H_1 + 4H_2 + H_3)} \quad \ldots\ldots\ldots\ldots (3.5)$$

where

$K_1 = hf(x_n, y_n)$.

$K_2 = hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1)$.

$K_3 = hf(x_n + h, y_n - k_1 + 2k_2)$.

$K_1 = hg(x_n, z_n)$

$H_2 = hg(x_n + \frac{1}{2}h, Z_n + \frac{1}{2}H_1)$.

$H_3 = hg(x_n + h, y_n - K_1 + 2k_2)$.

During the analysis of the schemes, he discovered that the schemes are A-stable. This property stimulated the use of the formular for numerical solution of stiff ordinary differential equations especially mild stiff equations.

These stability properties and its ease of programming probably stimulated or encouraged Okunboh in extending the schemes to family of order four in 1985. Example of order four formular is;

$$y_{n+1} = \frac{y_n + \frac{1}{12}(K_1 + 2K_2 + 2K_3 + K_4)}{1 + \frac{y_n}{12}(H_1 + 2H_2 + 2H_3 + H_4)} \quad \ldots\ldots\ldots\ldots\ldots\ldots (3.6)$$

where

$K_1 = hf(x_n, y_n)$.

$K_2 = hf(x_n + ½h, y_n + ½k_1)$

$K_3 = hf(x_n + ½h, y_n + ½K_2)$.

$K_4 = hf(x_n + h, y_n + K_3)$.

$H_1 = hg(x_n, z_n)$

$H_2 = hg(x_n + ½h, z_n + ½H_1)$.

$H_3 = hg(x_n + ½h, z_n + ½H_2)$.

$H_4 = hf(x_n + h, z_n + H_3)$.

These family of Explicit rational Runge-Kutta schemes perform well on certain class of mildly stiff equations.

However, when they are applied to sophisticated stiff equations whose Jacobian eigen values are close to imaginary axis, their performance were found to be very poor.

In view of the inadequacies of explicit Rational R-K schemes and the success history of the conventional Implicit Runge-Kutta schemes, we are motivated to consider in this chapter the possible extension of rational Runge-Kutta schemes to Implicit formular which up to this time has not been taken up by any authority.

In particular, we shall develop some implicit one-stage and two stage scheme of order four for numerical approximation of stiff ordinary differential equations.

## 3.2 THE DEVELOPMENT OF THE PROPOSED SCHEME

Recalling from (3.1) that an r-stage implicit rational Runge-Kutta scheme is

$$y_{n+1}=\frac{y_n+\sum_{i=1}^{r} W_i K_i}{1+y_n\sum_{i=1}^{r} V_i H_i}\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.7)$$

where

$$K_i=hf(x_n+C_ih,y_n+\sum_{j=1}^{i} a_{ij}k_j)\ i=1(1)r\ldots\ldots\ldots (3.8)$$

$$H_i=hg(x_n+d_ih,z_n+\sum_{j=1}^{i} b_{ij}H_j)\ \ldots\ldots\ldots\ldots\ldots (3.9)$$

$$g(x_n, z_n) = -Z_n^2 f(x_n, y_n)\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.10)$$

$$z_n = 1/y_n\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.11)$$

with the constraints

$$C_i=\sum_{j=1}^{i} a_{ij},\ i=1(1)r$$
$$d_i=\sum_{j=1}^{i} b_{ij},\ i=1(1)r\ldots\ldots\ldots\ldots (3.12)$$

and h is the step size or grid-spacing. These constraints ensure the cosistency of the schemes. The parameters $V_i$, $W_i$, $C_i$, $d_i$, $a_{ij}$ and $b_{ij}$ are to be determined from the systems of (non-linear) equations generated by adopting the following steps:

I. Obtained the Taylor series expansion of $K_i$'s and $H_i$'s about point $(x_n, y_n)$ for $i = 1(1)r$.

II. Insert the series expansion into (3.7).

III. Compare the final expansion with the Taylor series expansion of $y_{n+1}$ about $(x_n, y_n)$ in the powers of h.

The numbers of parameters normally exceeds the number of equations, but in the spirit of King (1966), Gill (1951) and Blum (1952), these parameters are chosen to ensure that (one or more of the following conditions are satisfied):

I Adequate order of accuracy of the scheme is achieved.

Ii Minimum bound of local truncation error exists.

Iii The method has maximize interval of absolute stability.

Iv Minimize computer storage facilities are utilized.

The determination of these parameters are taking up in the next unit.

### 3.2.1 <u>One Stage Schemes</u>

By equation (3.7), the general one-stage implicit rational Runge-Kutta scheme of order two is of the form:

$$y_{n+1} = \frac{y_n + W_1 K_1}{1 + y_n V_1 H_1} \quad \ldots\ldots\ldots\ldots\ldots\ldots 3.13$$

where

$$K_1 = hf(x_n + c_1 h,\ y_n + a_{11} K_1) \quad \ldots\ldots\ldots\ldots\ldots (3.14)$$

$$H_1 = hg(x_n + d_1 h,\ z_n + b_{11} H_1) \quad \ldots\ldots\ldots\ldots\ldots (3.15)$$

$g(x_n, z_n) = -z_n^2 f(x_n, y_n)$

and $z_n = 1/y_n$ .................................. (3.16)

with the constraints

$c_1 = a_{11}$ .......................................... (3.17)

$d_1 = b_{11}$ .......................................... (3.18)

Adopting binomial expansion theorem on the right hand side of (3.13) and ignoring terms of order higher than one, yields

$$y_{n+1} = y_n + W_1 K_1 - y_n^2 V_1 H_1 + (\textit{higher order terms}) \ldots\ldots\ldots\ldots (3.19)$$

The Taylor series expansion of $y_{n+1}$ about $y_n$ gives

$$y_{n+1} = y_n + hy_n' + \frac{h^2}{2!} y_n'' + \frac{h^3}{3!} y_n^{(3)} + \frac{h^4}{4!} y_n^{(4)} + 0(h^5) \ldots\ldots\ldots\ldots\ldots (3.20)$$

Now

$y_n' = f(x_n, y_n) = f_n$

$y_n'' = f_x + f_n f_y = Df_n$.

$y_n^{(3)} = f_{xx} + 2f_n f_{xy} + f_n^2 f_{yy} + f_y(f_x + f_n f_x) = D^2 f_n + f_y Df_n$.

$y_n^{(4)} = f_{xxx} + 3f_n f_{xxy} + 3f_n^2 f_{xyy} + f_n^3 f_{yyy} + f_y (f_{xx} + 2f_n f_{xy} + f_n^2 f_{yy})$

$+ (f_x + f_n f_y)(3f_{xy} + 3f_n f_y + f_y^2)$.

$= \quad D^3 f_n + f_y D^2 f_n + 3Df_n Df_y + f_y^2 Df_n$ .................................................. (3.21)

where

$D^2 f_n = f_{xx} + 2f_n f_{xy} + f_n^2 f_{yy}$

$Df_y = f_{xy} + f_n f_{yy} + f_y^2$.

$D^3 f_n = f_{xxx} + 3f_n f_{xxy} + 3f_n^2 f_{xyy} + f_n^3 f_{yyy}$.......................................................... (3.22)

Substitute (3.21) into (3.20).

$$y_{n+1}=y_n+hf_n+\frac{h^2}{21}Df_n+\frac{h^3}{31}(D^2f_n+f_yDf_n)$$
$$+\frac{h^4}{4!}(D^3f_n+f_yD^2f_n+3Df_nDf_y+f_y^2Df_n)+0(h^5)\ldots\ldots\ldots\ldots\ldots(3.23)$$

Similarly expanding $K_1$ about $(x_n, y_n)$, we have

$$K_1=h(f_n+(C_1f_x+a_{11}K_1+y)+½(C_1^2h^2f_{xx}+2C_1ha_{11}K_1f_{xy}+a_{11}K_1^2f_{yy})+ 0(h^4)\ldots\ldots\ldots\ldots\ldots 3.24)$$

Collecting Coefficients of equal powers of h, equation (3.24) can be expressed in the form

$K_1 = hA_1 = h^2B_1 + h^3D_1 + 0(h^4)$.................................................. (3.25)

where

$A_1 = f_n.$

$B_1 = c_1(f_x + f_nf_y) = c_1\, Df_n.$

$$D_1=C_1B_1f_y+½C_1^2(f_{xx}+2f_nf_{xy}+f_n^2f_{yy})=C_1^2(Df_nf_y+½D^2f_n)\ldots(3.26)$$

In a similar manner, expansion of $H_1$ about $(x_n, z_n)$ yields

$H_1 =hN_1 + h^2M_1 + h^3R_1 + 0h^4$ ............................................................ (3.27)

where

$N_1 = g(x_n, z_n) = g_n$

$M_1 = d_1(g_x + g_ng_z) = d_1Dg_n$

$R_1 = d_1M_1g_z + ½d_1^2(g_{xx} + 2g_ng_{xz} + g_n^2g_{zz}).$

$= d_1^2(g_zDg_n + {}^1/_2D^2g_n)$........................................................ (3.28) with

$$Dg_n = g_x + g_n g_z$$

$$D^2 g_n = g_{xx} + 2g_n g_{xz} + g_n^2 g_{zz} \quad \text{.......} (3.29)$$

Here we have assumed that f and g in (3.11) are sufficiently differentiable functions. Thus, we can express g and its partial derivatives in terms of f and its partial derivatives to facilitates the comparison of coefficients.

That is

$$g_n = \frac{-f_n}{y_n^2}, \ g_x = \frac{-f_x}{y_n^2}, \ g_{xx} = \frac{-f_{xx}}{y_n^2}$$

$$g_z = \frac{-2f_n}{y_n} + f_y, \ g_{xz} = \frac{-2f_x}{y_n} + f_{xy}$$

$$g_{xxz} = \frac{-2f_{xx}}{y_n} + f_{xxy}, \quad g_{zz} = -2f_n - y_n^2 f_{yy}$$

$$g_{xzz} = -2f_x - y_n^2 f_{xyy}$$

$$g_{zzz} = 4y_n^2 f_y + 6y_n^2 f_{yy} + y_n^4 f_{yyy} \quad \text{.......} (3.30)$$

Substituting (3.30) into (3.28),

We obtained

$$N_1 = -f_n/y_n^2$$

$$M_1 = \frac{-d_1}{y_n^2} \left( Df_n + 2\frac{f_n^2}{y_n} \right)$$

$$R_1 = \frac{-d_1^2}{y_n^2} \left[ \left( -\frac{2f_n}{y_n} + f_y \right) \left( Df_n + \frac{f_{n_2}}{y_n} \right) \right.$$

$$\left. + \tfrac{1}{2} \left( D^2 f_n - \frac{2f_n}{y_n} \left( \frac{f_n^2}{y_n} + f_x \right) \right) \right] \quad \text{.......} (3.31)$$

Adopting (3.25) and (3.27) in (3.19)

we have

$$y_{n+1}=y_n+W_1(hA_1+h^2B_1+h^3R_1+0h^4)-y_n^2(V_1(hN_1+h^2M_1+h^3R_1+0(h^4)$$
$$=y_n+(W_1A_1-y_n^2V_1N_1)h\ +(W_1B_1-y_n^2V_1M_1)h^2+(W_1D_1y_n^2V_1-y_n^2V_1R_1)h^3+$$
$$0(h^4)\ldots\ldots\ldots\ldots\ldots\ldots(3.32)$$

Comparing the coefficients of the powers of h in equations (3.23) and (3.32), we obtained

$$W_1A_1 - y_n^2V_1N_1 = f_n \ldots\ldots\ldots\ldots\ldots\ldots (3.33)$$

From (3.26) and (3.31), we have

$$A_1 = f_n,\ N_1 = -f_n/y_n^2 \qquad \ldots\ldots\ldots\ldots\ldots\ldots (3.34)$$

(3.33) yields

$$W_1 + V_1 = 1 \ldots\ldots\ldots\ldots\ldots\ldots (3.35)$$

Comparing coefficients of $h^2$ in equations (3.23) and (3.32), we obtained

$$W_1B_1 - y_n^2V_1M_1 = Df_n \ldots\ldots\ldots\ldots\ldots\ldots (3.36)$$

and from (3.26) and (3.31), we have

$$B_1 = c_1Df_n.$$

$$M_1=\frac{-d_1}{y_n^2}(Df_n+\frac{2fn^2}{y_n})\ldots\ldots\ldots\ldots\ldots\ldots(3.37)$$

Substituting (3.37) to (3.36), we obtain

$$(W_1c_1+V_1d_1)Df_n+\frac{2f_n^2}{y_n}V_1d_1=\frac{Df_n}{2}\ldots\ldots\ldots\ldots(3.38)$$

From (3.38) we obtain

$$W_1C_1 + V_1d_1 = ½ \ldots\ldots\ldots\ldots\ldots\ldots (3.39)$$

Taking coefficients of h and $h^2$ into consideration, and imposing condition

$$T_{n+1} = 0(h^3) \quad \ldots\ldots \quad (3.40)$$

We obtained the following systems of equations for family of one-stage schemes of order two.

$W_1 + V_1 = 1$

$W_1 c_1 + V_1 d_1 = ½$

With

$a_{11} = c_1$

$b_{11} = d_1 \quad \ldots\ldots \quad (3.41)$

and a local truncation error

$$T_{n+1} = y(x_{n+1}) - \frac{y(x_n) + W_1 K_1}{1 + y(x_n) V_1 H_1} \ldots (3.42)$$

which can be shown to be

$$T_{n+1} = (D^2 f_n + f_y D f_n)\left(\frac{1}{6} - ½ C_1^2 W_1 - ½ V_1 d_1^2\right) - V_1 d_1^2 \left(\frac{2f_n}{y_n}\right) - \frac{2 f_y f_n}{y_n}) h^3$$
$$+ 0(h^4) \ldots\ldots (3.43).$$

i. $w_1 = 0,\ v_1 = 1$

$c_1 = d_1 = ½$

$a_{11} = b_{11} = 1/2$

in (3.13) resulting in

$$y_{n+1} = \frac{y_n}{1 + y_n H_1} \quad \ldots\ldots \quad (3.44)$$

where

$$H_1 = hg\,(x_n + \tfrac{1}{2} h, \quad Z_n + \tfrac{1}{2} H_1)$$

ii $\quad V_1 = W_1 = \tfrac{1}{2}, \qquad c_1 = a_{11} = \tfrac{3}{4},$

$d_1 = b_{11} = \tfrac{1}{4}$

equation (3.13) becomes

$$y_{n+1} = \frac{y_n + \tfrac{1}{2} K_1}{1 + \frac{y_n}{2} H_1} \qquad \text{.............................(3.45)}$$

where

$$K_1 = hf(x_n + \tfrac{3}{4}h, \quad y_n + \tfrac{3}{4} K_1).$$

$$H_1 = hg(x_n + \tfrac{1}{4}h, \quad Z_n + \tfrac{1}{4} H_1).$$

iii $\quad W_1 = \tfrac{1}{4}, V_1 = \tfrac{3}{4}, \qquad d_1 = c_1 = \tfrac{1}{2},$

$a_{11} = b_{11} = \tfrac{1}{2}.$

equation (3.13) yields.

$$y_{n+1} = \frac{y_n + \tfrac{1}{4} K_1}{1 + \tfrac{3}{4} y_n H_1} \qquad \text{........................(3.46)}$$

where

$$K_1 = hf(x_n + \tfrac{1}{2}h, \quad y_n + \tfrac{1}{2} K_1)$$

$$H_1 = hg(x_n + \tfrac{1}{2}h, \quad Z_n + \tfrac{1}{2} H_1)$$

iv. $W_1 = 1,\ V_1 = 0,\ C_1 = d_1 = ½$.

$a_{11} = b_{11} = ½$

equation (3.13) yields

$$y_{n+1} = y_n + K_1 \qquad \text{.........................(3.47)}$$

where,

$$K_1 = hf(x_n + ½h, \quad y_n + ½ K_1).$$

which coincides with implicit Euler's Scheme of order two.

v. $W_1 = {}^1/_3\ ,\ V_1 = {}^2/_3,\ a_{11} = c_1 = {}^1/_3,$

$b_{11} = d_1 = {}^7/_{12}.$

equation (3.13) becomes

$$y_{n+1} = \frac{y_n + {}^1/_3 K_1}{1 + {}^2/_3\, y_n H_1} \qquad \text{.........................(3.48)}$$

where

$$k_1 = hf\,(x_n + {}^1/_3\, h, \quad y_n + {}^1/_3\, K_1)$$

$$H_1 = hg\,(x_n + {}^7/_{12} h, \quad x_n + {}^7/_{12}\, H_1)$$

3.2.2 **Two –Stage Schemes**

The general two –stages (implicit rational Runge-Kutta) Scheme is of the form

$$y_{n+1} = \frac{y_n + \sum_{i=1}^{2} W_i K_i}{1 + y_n \sum_{i=1}^{2} V_i H_i} \qquad \text{(3.49)}$$

with local truncation error

$$T_{n+1} = \frac{y(x_{n+1}) - y(x_n) + \sum_{i=1}^{2} W_i K_i}{1 + y(x_n) \sum_{i=1}^{2} V_i H_i} \qquad \text{(3.50)}$$

where

$$K_i = hf(x_n + c_i h,\ y_n + \sum_{j=1}^{2} a_{ij} k_j), \quad i = 1\ (1)\ 2 \qquad \text{(3.51)}$$

$$H_i = hg(x_n + d_i h\ z_n + \sum_{j=1}^{2} b_{ij} H_j), \quad i = (1)2 \qquad \text{(3.52)}$$

$$g(x_n, z_n) = -Z_n^2 f(x_n, y_n) \text{ and } z_n = 1/y_n$$

Adopting Binomial expansion theorem on the right side of equation (3.49) and ignoring higher terms, we obtained.

$$y_{n+1} = y_n + \sum_{i=1}^{2} W_i K_i - y_n^2 \sum_{i=1}^{2} V_i H_i + (\text{higher order term}) \qquad 3.53$$

with

$$K_i = hf(x_n + c_i h,\ y_n + \sum_{j=1}^{2} a_{ij} k_j), \quad i = 1\ (1)\ 2$$

$$H_i = hg\,(x_n + d_i h,\ Z_n + \sum_{j=1}^{2} b_{ij}\ H_j), \qquad i = 1\ (1)\ 2$$

and the constraints

$$c_i = \sum_{j=1}^{2} a_{ij}, \qquad i = 1\ (1)\ 2$$

$$d_i = \sum_{j=1}^{2} b_{ij} \qquad i = 1\ (1)\ 2 \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.54)$$

Expanding $K_i$'s $i = 1(1)2$ in equation (3. 51) about $(x_{n1}\ y_n)$, we obtain

$$K_i = h\,[f_n + (c_i h\ f_x + (a_{i1} k_1 + a_{12}\ K_2)\ f_y) + \tfrac{1}{2}\ (C_i^2 h^2 f_{xx} + 2C_i h(a_{i1} K_1 + a_{12} K_2)\ f_{xy} + (a_1 K_i + a_2 K_2)^2\ f_{yy} + 1/6\ (C_i^2 h^3\ f_{xxx} + 3C_i^2 h^2\ (a_1 k_i + a_2 K_2)\ f_{xxy} + 3C_i h\ (a_1 K_i + a_2 K_2)^2 f_{xyy} + (a_1 K_i + a_{12} K_2)^3 f_{yyy} \ldots\ldots(3.55)$$

By collecting coefficients of terms of the equal powers of h (3.55) becomes

$$K_i = h\,A_i + h^2 B_i + h^3 E_i + h^4 D_i + O(h^5)\ ,i = 1(1)2 \ldots\ldots\ldots\ldots\ldots\ldots(3.56)$$

where

$A_i = f_n$

$B_i = c_i f_x + (a_{i1} A_1 + a_{i2} A_2)\, f_y$

$E_i = (a_{i1} B_i + a_{12} B_2) f_y + \tfrac{1}{2} c_i^2\ f_{xx} + (a_{i1} A_1 + a_{12} A_2)^2 f_{xy} + c_i\, (a_{i1} A_1 + a_{12} A_2)^2\ f_{yy}$

$D_i = [(a_{i1} E_i + a_{12} E_2)\, f_y + c_i\, (a_{i1} B_i + a_{12} B_2)\, f_{xy} + (a_i A_1 + a_{i2} A_1)\,(a_{i1} B_1 + a_{i2} B_2)\, f_{yy} + {}^1/_6\, c_i^3 f_{xxx} + \tfrac{1}{2} c_i\, (a_{i1} A_1 + a_{12} A_2)\, f_{xxy} + \tfrac{1}{2} c_i (a_{i1} A_1 + a_{12} A_2)\ f_{xyy} + {}^1/_6\, (a_{i1} A_1 + a_{i2} A_2)^3\ f_{yyy},\quad i = 1(1)\,2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.57)$

using (3.21) in (3.57)

we have

$A_i = f_n$

$B_i = c_i Df_n$

$E_i = (a_{i1} c_1 + a_{12} c_2) f_y Df_n + ½ c_i {}^2 D^2 f_n$

$D_i = [a_{i1} (a_{11}c_1 + a_{12}c_2) + a_{12} (a_{21}c_1 + a_{22}c_2)] f_y^2 Df_n + ½ c_i (a_{i1}c_1 + a_{12} c_2) Df_n \; Df_y$

$+½ (a_{i1} c^2{}_1 + a_{12} c^2{}_2) f_y D^2 f_n + {}^1/_6 c^3{}_1 D^3 f_n$ ............(3.58)

In a similar manner expanding Hi, $i = 1(1)2$ about $(x_n, z_n)$ in equation (3.48), we obtained

$H_i = hN_i + h^2 M_i + h^3 R_1 + h^4 L_1 + O(h^5), \quad i = 1(1)\,2$ ....................(3.59)

where

$N_i = g(x_n, z_n) = g_n$

$M_i = d_i g_x + (b_{i1} N_1 + b_{i2} N_2) = d_i Dg_n$

$R_i = (b_{i1} M_1 + b_{12} M_2) g_z + ½ d^2{}_i g_{xx} + d_i (b_{i1} N_1 + b_{i2} N_2) g_{xz} + ½ (b_{i1} N_1 + b_{i2} N_2)^2 g_{zz}$

$= (b_{i1} d_i + b_{i2} N_2) g_z Dg_n + ½ d_i D^2 g_n$

$L_i = (b_{ii} R_1 + b_{i2} R_2) g_z + (b_{i1} M_1 + b_{12} M_2) g_{xz} + [(b_{i1} N_1 + b_{i2} N_2)(b_{i1} M_1 + h_{12} M_2)] g_{zz}$

$+1/6\, d^3{}_1 g_{xxx} + ½ d^2{}_i (b_{i1} N_1 + b_{i2} N_2) g_{xxz} + ½ d_i (b_{i1} N_1 + b_{i2} N_2) g_{xzz} + 1/_6 (b_{i1} N_1$

$+ b_{i2} N_2) g_{zzz}$

$= [b_{i1}(b_{i1} d_1 + b_{i2} d_2) + b_{i2} (b_{21} d_1 + b_{22} d_2)] g^2{}_z Dg_n + (d_i (b_{i1} d_i + b_{i2} d_2) Dg_n Dg_z +$

$½ (b_{i1} d^2{}_1 + b_{12} d^2{}_2) g_z D^2 g_n + {}^1/_6 d^3{}_1 \; D^2 g_n, \quad i = 1(1)2$ ........................(3.60)

where

$Dg_n = g_x + g_n g_z$

$D^2 g_n = g_{xx} + 2g_n g_{xz} + g^2{}_n g_{zz}$

$D^3 g_n = g_{xxx} + 3g_n g_{xxz} + 3g^2{}_n g_{xzz} + g^3{}_n g_{zzz}$

$$Dg_z = g_{xz} + g^2_z + g_n g_{zz} \qquad (3.61)$$

Using equation (3.30) in (3.60), we get

$$N_i = f_n/y^2_n$$

$$M_i = -\frac{d_i}{y^2_n}(Df_n + \frac{2f^2_n}{y_n})$$

$$R_i = \frac{1}{y^2_n}[[( b_{i1} d_1 + b_{i2} d_2) (\frac{-2 f_n}{y_n} + f_y)](Df_n + \frac{2f^2_n}{y_n}) + ½ (d_i^2 (D^2fn + \frac{2f_n}{y_n} (fx + \frac{2fn}{y_n})))]$$

$$L_i = \frac{-1}{y^2_n} [ b_{i1} (b_{11}d_1 + b_{i2}d_2) + b_{i2}(b_{21}d_1 + b_{22} d_1](\frac{-2 f_n}{y_n} + f_y) (Df_n + \frac{2f^2_n}{y_n}) + (d_i(b_{i1} d_1 + b_{i2} d_2)Df_n + \frac{2f^2_n}{y_n} ( \frac{-2f_n}{y_n}+f_y)] + 1/2 [(b_{i1} d^2_1 + b_{12} d_2^2)(D^2f_n - \frac{2f_n}{y_n} ( f_x - \frac{f_n^2}{y_n})] + 1/6 d^3_1 f_n f_x ) + 2f_n^3 (2f_y + 3f_{yy} )) \qquad (3.62)$$

Recalling ( 3.53), we have

$$y_{n+1} = y_n + (W_1 K_1 + W_2 K_2) - y_n^2 (V_1 H_1 + V_2 H_2) + (\text{higher order terms}) \qquad 3.63)$$

Now, using (3.56) and (3.59), we obtain

$$y_{n+1} = [ (W_1 A_1 + W_2 A_2) - y_n^2 (V_1 N_1 + V_2 N_2)] h + [ (W_1 B_1 + W_2 B_2) - y_n^2 (V_1 M_1 + V_2 M_2)] h^2 + (W_1 E_1 + W_2 E_2) - y_n^2 (V_1 R_1 + V_2 R_2)] h^3 + [ (W_1 D_1 + W_2 D_2) - y_n^2 (V_1 L_1 + V_2 L_2)] h^4 + oh^5 \qquad (3.64)$$

Comparing the Coefficients of h, $h^2$ and $h^3$ in equations (3.21) and (3.64) where condition

$$T_{n+1} = O (h^4) \qquad (3.65)$$

is imposed. We obtained the following systems of equations for family of two-stage schemes of order three.

$$W_1 + W_2 + V_1 + V_2 = 1$$

$$W_1 c_1 + W_2 c_2 + V_1 d_1 + V_2 d_2 = ½$$

$W_1 ( a_{11} c_1 + a_{12} c_2 ) + W_2 ( a_{21} c_1 + a_{22} c_2 ) + V_1 ( b_{11} d_1 + b_{12} d_2 ) + ( b_{21} d_1 + b_{22} d_2 )$

$$= 1/6$$

$$W_1 C_1^2 + W_2 C_2^2 + V_1 d_1^2 + V_2 d_2^2 = {}^1/_3 \quad \ldots\ldots (3.66)$$

with the constraints

$$a_{11} + a_{12} = c_1$$
$$a_{21} + a_{22} = c_2$$
$$b_{11} + b_{12} = d_1$$
$$b_{21} + b_{22} = d_2 \quad \ldots\ldots (3.67)$$

The associated local truncation error term for the family of Two-stage implicit rational Runge- Kutta Schemes of order three can be shown to be

$$T_{n+1} = -\frac{h^4}{24} [( D^3 f_n + fy \; D^2 f_n + 3Df_n Df_y + fy^2 Df_n) + {}^1/_6 (V_1 ( b_{11} d_1 + b_{12} d_2) +$$

$$V_2 ( b_{21} d_1 + b_{22} d_2)] [-\frac{2 f_n}{y_n} Df_n - \frac{2f_n^3}{y_n} + \frac{2f_n^2}{y_n} f_y) - 1/2 (V_1 d_1^2 + V_2 d_2^2)$$

$$( \frac{2f_n}{y_n} (fx - f_n)] \ldots\ldots 3.68.$$

Family of two stage schemes of order three are obtained as

i. $W_1 = W_2 = 0, \qquad V_1 = ¼, \; V_2 = ¾$

$c_1 = d_1 = a_{12} = b_{12} = 1$

$a_{11} = b_{11} = a_{21} = b_{21} = 0$

$c_2 = d_2 = a_{22} = b_{22} = 1/3$

equation( 3.49) yields

$$y_{n+1} = \frac{y_n}{1 + \frac{y_n}{4} ( H_1 + 3H_2)} \ldots\ldots 3.69$$

where

$$H_1 = hg\,(x_n + h,\ \ Z_n + H_2)$$

$$H_2 = hg\,(x_n + 1/3\,h,\ \ Z_n + 1/3H_2)$$

ii. $W_1 = W_2 = 0 \quad V_1 = ¾\ ,\ V_2 = ¼$

$a_{11} = a_{22} = b_{11} = b_{22} = ½$

$b_{12} = a_{12} = {}^1/_6,\ a_{21} = b_{21} = -{}^1/_2$

$c_2 = d_2 = 0\ ,\ \ d_1 = c_1 = {}^2/_3$

equation (3.49) becomes

$$y_{n+1} = \frac{y_n}{1 + \frac{y_n}{4}\,(3H_1 + H_2)} \qquad \text{..................................} (3.70)$$

where

$$H_1 = hg(x_n + {}^2/_3\,h,\ \ Z_n + {}^1/_2\,H_1 + {}^1/_6\,H_2)$$

$$H_2 = hg(x_n\,,\ \ Z_n - {}^1/_2\,H_1 + {}^1/_2\,H_2)$$

iii. $V_1 = W_1 = 0, \qquad V_2 = W_2 = ½$

$c_1 = d_1 = 0, \qquad c_2 = ½ + \sqrt{3}/_6,$

$d_2 = {}^1/_2 - \sqrt{3}/_6,\ \ a_{22} = b_{22} = {}^1/_3$

$a_{21} = \dfrac{1 + \sqrt{3}}{6} = \qquad b_{21}\ \dfrac{1 - \sqrt{3}}{6}$

$b_{11} = a_{11} = {}^1/_3,\ b_{12} = a_{12} = -{}^1/_3$

equation (3.49) yields

$$y_{n+1} = \frac{y_n + ½\,K_2}{1 + \frac{y_n}{2}\ H_2} \qquad \text{..................................} (3.71)$$

where

$$K_1 = hf\,(x_n, -y_n + {}^1/_3\,K_1 - {}^1/_3\,K_2)$$

$$K_2 = hf\left(x_n, +\left(\tfrac{1}{2} + \frac{\sqrt{3}}{6}\right)h,\ y_n + \frac{(1+\sqrt{3})}{6}K_1 + {}^1/_3 K_2\right)$$

$$H_1 = hg\left(x_n, Z_n + {}^1/_3 H_1 - {}^1/_3 H_2\right)$$

$$H_2 = hg\left(x_n, + \left(\tfrac{1}{2} - \frac{\sqrt{3}}{6}\right)h,\ Z_n + \frac{(1+\sqrt{3})}{6}H + {}^1/_3 H_2\right)$$

iv. $V_1 = W_1 = 0, \qquad V_2 = {}^1/_3, \quad W_2 = {}^2/_3$

$C_2 = 1 + \frac{\sqrt{4}}{12}, \quad d_2 = \tfrac{1}{2} - \frac{\sqrt{6}}{6}$

$a_{22} = b_{22} = \tfrac{1}{2}, \; a_{11} = b_{11} = \tfrac{1}{2}$

$c_1 = d_1 = 0$

$a_{21} = \frac{1}{6} + \frac{\sqrt{6}}{12}, \quad b_{21} = \frac{1}{6} - \frac{\sqrt{6}}{6}$

$a_{12} = b_{12} = -{}^1/_2$

equation (3.49) yields

$$y_{n+1} = \frac{y_n + {}^2/_3 K_2}{1 + \frac{y_n}{3} H_2} \qquad \text{.............................(3.72)}$$

where

$$K_1 = hf\left(x_n, y_n + K_1 - \tfrac{1}{2} K_2\right)$$

$$K_2 = hf\left(x_n + \left({}^1/_2 + \frac{\sqrt{6}}{12}\right)h,\ y_n + \frac{\sqrt{6}}{12}K_1 + \tfrac{1}{2}K_2\right)$$

$$H_1 = hg\left(x_n + \left({}^1/_3 H_1 - 1/_3 H_2\right)\right.$$

$$H_2 = hg\left(x_n + \tfrac{1}{2} - \frac{\sqrt{6}}{6}\right)h,\ Z_n - \frac{\sqrt{6}}{6}H_1 + \tfrac{1}{2}H_2\Big)$$

Similarly setting

v. $V_1 = V_2 = {}^1/8,\ W_1 = W_2 = {}^3/_8$

$c_1 = c_2 = 1,\quad d_1 = d_2 = {}^1/_3$

$a_{11} = a_{22} = {}^2/_3,\ a_{21} = a_{12} = {}^1/_3$

$b_{11} = b_{22} = {}^1/_9,\ b_{21} = b_{12} = {}^2/_9$

in equation (3.49), we get

$$y_{n+1} = \frac{y_n + {}^3/_8(K_1 + K_2)}{1 + \frac{y_n}{8}(H_1 + H_2)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.73)$$

where

$K_1 = hf\,(x_n + h,\ \ y_n + {}^2/_3 K_1 + {}^1/_3 K_2)$

$K_2 = hg\,(x_n + h,\ \ y_n + {}^1/_3 K_1 + {}^2/_3 K_2)$

$H_1 = hg\,(x_n\ 1/_3 h,\ \ Z_n + {}^2/_9 H_1 + {}^1/_9 H_2)$

$H_2 = hg\,(x_n + {}^1/_3 h,\ \ Z_n + {}^1/_9 H_1 + {}^2/_9 H_2)$

Also, if

vi. $V_1 = W_1 = {}^1/_8$ $\qquad W_2 = V_2 = {}^3/_8$

$d_1 = c_1 = 1,$ $\qquad d_2 = c_2 = {}^1/_3$

$a_{22} = b_{22} = {}^1/_3$ $\qquad b_{11} = a_{11} = 0$

$a_{12} = b_{12} = 1,$ $\qquad a_{21} = b_{21} = 0$

and substitute in equation (3.49), we have

$$y_{n+1} = \frac{y_n + {}^1/_8(K_1 + 3K_2)}{1 + \frac{y_n}{8}(H_1 + 3H_2)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.74)$$

where

$$K_1 = hf(x_n + h, \quad y_n + K_2)$$

$$K_2 = hf(x_n + {}^1/_3h \quad y_n + {}^1/_3 K_2)$$

$$H_1 = hg(x_n + h, \quad Z_n + H_2)$$

$$H_2 = hg(x_n + {}^1/_3h, \quad Z_n + {}^1/_3H_2)$$

Imposing Condition

$$T_{n+1} = 0(h^5) \quad \text{.................................(3.75)}$$

Coefficient of h, $h^2$, $h^3$ and $h^4$ into consideration in equations (3.21) and (3.64).

we obtain the following equations for two stage family of order four.

$$V_1 + V_2 + W_1 + W_2 = 1$$

$$W_1 C_1 + W_2 C_2 + V_1 d_1 + V_2 d_2 = ½$$

$$W_1 C^2{}_1 + W_2 C^2{}_2 + V_1 d^2{}_1 + V_2 d^2{}_2 = {}^1/3$$

$$W_1 c_1{}^3 + W_2 c_2{}^3 + V_1 d_1{}^3 + V_2 d_2{}^3 = ¼$$

$$W_1 ( a_{11} c_1 + a_{12} c_2) + W_2 ( a_{21}c_1 + a_{22}c_2) + V_1 ( b_{11} d_1 + b_{12} d_2 ) + V_2 ( b_{21} d_1 + b_{22} d_2) = {}^1/_6$$

$$W_1 c_1 (a_{11}c_1 + a_{12}c_2) + W_2c_2 (a_{21} c_1 + a_{22}c_2) + V_1 d_1 ( b_{11} d_1 + b_{12} d_2 ) + V_2 d_2 ( b_{21} d_1 + b_{22} d_2) = ¼$$

$$W_1 (a_{11} c^2{}_1 + a_{12} c^2{}_2) + W_2 (a_{21}c^2{}_1 + a_{22}c^2{}_2) + V_1 (.b_{11} d^2{}_1 + b_{12} d_2 ) + V_2 ( b_{21} d^2{}_1 + b_{22} d^2{}_2) = ½$$

$$W_1 [a_{11} ( a_{11}c_1 + a_{12}c_2) + a_{12} (a_{21}c_1 + a_{22}c_2)] + W_2 [a_{21} ( a_{11}c_1 + a_{12}c_2) + a_{22} (a_{22}c_1 + a_{22}c_2)] + V_1 [ b_{11}( b_{11} d_1 + b_{12} d_2 ) + b_{12} ( b_{21} d_1 + b_{22} d_2)] + V_2 [b_{21} ( b_{11} d_1 + b_{12} d_2 ) + b_{22} ( b_{21} d_1 + b_{22} d_2)] = {}^1/_{24} \quad \text{..........................(3.76)}$$

With the following constraints

$C_1 = a_{11} + a_{12}$
$C_2 = a_{21} + a_{22}$
$d_1 = b_{11} + b_{12}$
$d_2 = b_{21} + b_{22}$ .......................................... (3.77)

Examples of possible family of two-stage schemes of order four are obtained by setting:

i $\quad V_1 = V_2 = 0, \quad W_1 = W_2 = ½$

$$d_1 = C_1 = ½ + \frac{\sqrt{3}}{6}$$
$$d_2 = C_2 = ½ - \frac{\sqrt{3}}{6},$$
$$a_{22} = b_{22} = b_{11} = b_{22} = ¼$$
$$a_{12} = b_{12} = ¼ + \frac{\sqrt{3}}{6}$$
$$a_{21} = b_{21} = ¼ - \frac{\sqrt{3}}{6}$$

these into equation (3.49) yields

$$y_{n+1} = y_n + ½(K_1 + K_2) \text{ ......... } (3.78)$$

where

$$k_1 = hf(x_n + (½ + \frac{\sqrt{3}}{6})h, y_n + ¼K_y + (¼ + \frac{\sqrt{3}}{6})k_2$$
$$K_2 = hf(x_n + (½ - \frac{\sqrt{3}}{6})h, y_n + (¼ - \frac{\sqrt{3}}{6})K_1 + ¼K_2)$$

which incidentally coincide with 2-stage implicit Runge-Kutta scheme of order four proposed by Harmmer and Holling Worth (1955).

ii. $W_1 = W_2 = 0,\quad v_1 = v_2 = ½,\ c_2 = d_2 = (½ - \sqrt{3}/6).$

$a_{11} = b_{11} = a_{22} = b_{22} = ¼$

$a_{12} = b_{12} = ¼ + \sqrt{3}/6$

$a_{21} = b_{21} = ¼ - \sqrt{3}/6.$

equation (3.49) becomes

$$y_{n+1} = \frac{y_n}{1 + y_{\frac{n}{2}}(H_1 + H_2)} \quad \ldots\ldots\ldots\ (3.79)$$

where

$$H_1 = hg\left(x_n + \left(½ + \frac{\sqrt{3}}{6}\right)h,\ z_n + ¼H_1 + \left(¼ + \frac{\sqrt{3}}{6}\right)H_2\right)$$
$$H_2 = hg\left(x_n + \left(½ - \frac{\sqrt{3}}{6}\right)h,\ z_n + \left(¼ - \frac{\sqrt{3}}{6}\right)H_1 + ¼H_2\right)$$

iii Also, with

$W_1 = W_2 = V_1 = V_2 = ¼$

$$C_1 = d_1 = ½ + \frac{\sqrt{3}}{6}$$
$$C_2 = d_2 = \left(½ - \frac{\sqrt{3}}{6}\right),\ a_{11} = b_{11} = b_{22} = b_{22} = \frac{1}{4}$$
$$a_{12} = b_{12} = \frac{1}{4} + \frac{\sqrt{3}}{6},\qquad a_{21} = b_{21} = ¼ - \frac{\sqrt{3}}{6}$$

equation (3.49) results into

$$y_{n+1} = \frac{y_n + ¼(K_1 + K_2)}{1 + \frac{yn}{4}(H_1 + H_2)} \quad \ldots\ldots\ldots\ldots\ (3.80)$$

where

$$K_1 = hf(x_n + (½ + \frac{\sqrt{3}}{6})h, \quad y_n + ¼K_1 + (¼ + \frac{\sqrt{3}}{6})K_2)$$

$$K_2 hf(x_n + (½ - \frac{\sqrt{3}}{6})h, y_n + (¼ - \frac{\sqrt{3}}{6})K_1 + ¼K_2)$$

$$H_1 = hg(x_n + (½ + \frac{\sqrt{3}}{6})h, z_n + ¼H, + (½ + \frac{\sqrt{3}}{6})$$

$$H_2 = h_g(x_n + ½ - \frac{\sqrt{3}}{6})h, z_n + (½ - \frac{\sqrt{3}}{6})H_1)$$

iv. $V_1 = V_2 = \frac{1}{6}, \quad W_1 = W_2 = \frac{1}{3} \quad C_2 = d_2 = (½ - \frac{\sqrt{3}}{6}), \quad C_1 = d_1 = ½ + \frac{\sqrt{3}}{6}$

$$a_{11} = a_{22} = b_{11} = b_{22} = ¼ + \frac{\sqrt{3}}{6}$$

$$b_{12} = a_{12} = b_{11} = b_{22} = ¼ + \frac{\sqrt{3}}{6}$$

$$y_{n+1} = \frac{y_n + \frac{1}{3}(K_1+K_2)}{1+y_{\frac{n}{6}}(H_1+H_2)} \ldots\ldots\ldots\ldots\ldots (3.81)$$

$$K_1 = hf(x_n + (\frac{1}{2}+\frac{\sqrt{3}}{6})h, y_n + (\frac{1}{4}+\frac{\sqrt{3}}{6})K_1 + \frac{1}{4}K_2.$$

$$K_2 = hf(x_n + (\frac{1}{2}-\frac{\sqrt{3}}{6})h, y_n + (\frac{1}{4}+\frac{\sqrt{3}}{6})H_1 + \frac{1}{4}H_3$$

$$H_2 = hg(x_n + (\frac{1}{2}-\frac{\sqrt{3}}{6})h, z_n + (\frac{1}{4}-\frac{\sqrt{3}}{3})H_1 + (\frac{1}{4}+\frac{\sqrt{3}}{6})h_2.$$

Next chapter analyses the error, convergence and stability properties of these schemes.

## CHAPTER FOUR

## ERROR, CONVERGENCE, AND STABILITY PROPERTIES

A characteristic of numerical schemes is that errors are generated when they are adopted for approximation of solutions of ordinary differential equations.

The magnitude of these errors determines the degree of accuracy of the schemes and its effects can be great. It can make the solution unstable.

This reason makes the analysis of error, convergence and stability properties essential for the new schemes. This will enable us to know whether the methods have the capability to handle the types of problems we envisage.

Therefore, in this chapter, we shall consider the error, convergence and stability properties of the schemes proposed in chapter three.

### 4.1 ERROR ANALYSIS

Errors of numerical approximation techniques for stiff ordinary differential equations arise from different causes that can be majorly classified into discretization, truncation and round off errors respectively. Other sources of errors include inherent errors often called data or model simplification error and personal errors.

Round off errors is an error introduced as a result of the computing devices.

Mathematically, it can be expressed as

$$r_{n+1} = y_{n+1} - p_{n+1} \quad \text{.......................................} \quad (4.1)$$

where $y_{n+1}$ is the expected solution of the difference equations (3.7) while $p_{n+1}$ is the computer output at the $(n + 1)^{th}$ iteration.

It is the amount by which the computed approximation $p_{n+1}$ differs from the expected approximation $y_{n+1}$ of the schemes (3.7) at point $x_{n+1}$.

Accumulation of these errors depend upon many factors namely storage and manipulation of numbers, that is, the way and manners machine operation are performed.

However, the existing literature on error analysis (Fatunla, 1987 and Lambert, 1962) indicated that the effects of round off error can be disastrous. This is because there will be inevitable loss of accuracy. It is not amenable to analysis but it effect can only be controlled by employing double precision arithmetic.

Truncation error on the other hand is the error introduced as a result of ignoring some of the higher terms of the power series (Taylor and Binomial series expansion) during the development of the new formulas.

Mathematically, truncation errors are defined as the amount by which the true solution $y(x_{n+1})$ of the differential equation (1.4) fails to satisfy the difference equation (3.7). That is the truncation error of scheme (3.7) can be defined as

$$T_{n+1} = y(x_{n+1}) - \frac{y(x_n) + \sum_{i=1}^{r} W_i K_i}{1 + y(x_n) \sum_{i=1}^{r} V_i H_i} \quad \ldots\ldots\ldots (4.2)$$

where

Therefore, according to Lotkin (1951) arguement, the bound of equation (4.8) can be found if the bound for f and its partial derivatives

$$\left|\frac{\partial^{i+j} f(x,y)}{\partial x^{i} \partial y^{j}}\right| < N^{i+j}/M^{j-1} \ldots\ldots\ldots (4.9)$$

hold for $x \in [a, b]$, $y \in [-\infty, \infty]$, then the bound (4.8) yields inequality

$$|T_{n+1}| < \left[6|P_1|N^2M - 4|P_2|M\left(\frac{N^2}{2} - MN + M^3\right)\right]h^3 \ldots (4.10)$$

where

$$P_1 = \frac{1}{6} - ½W_1C_1^2 - ½V_1d_1^{2'} \quad , P_2 = \frac{V_1}{y_n}d_1^2 .$$

However, it can be seen that in (4.10) that the effect of the truncation error may be reduced by retaining a large number of terms in the series or making the step size h as small as possible.

Error introduced as a result of transforming a differential equation (1.4) into difference equation (3.1) leads to error often called discretization error.

Mathematically, the discretization error $e_{n+1}$ associated with the formula, (3.7) is the difference between, the exact solution $y(x_{n+1})$ and the numerical solution $y_{n+1}$ generated by (3.7) at point $x_{n+1}$. That is

$$e_{n+1} = y_{n+1} - y(x_{n+1}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.11)$$

However since the numerical solution of scheme (3.7) involve iteration process there will be propagation of error from step to step.

By the propagation of error, we mean the process by which the various errors mentioned above grow or decay from step to step.

When iterating with a numerical scheme, we obtained a sequence of values$\{y_i\}$ i = 1(1)n. If the value of $y_1$ has an error and since $y_2$ depends on $y_1$, $y_2$ will be subject to error. Since $y_3$ depends on $y_2$ and so on Errors are accumulated and the final solutions may be in serious inaccuracy. Such errors resulting from inheritance of errors in preceeding calculations which may either grow or decay is called propagation of error.

The accuracy of the numerical techniques will depends on the magnitude of these errors. The smaller the error, the better is the numerical solution.

To improve the quality of the integration techniques, it is essential to have estimate of these errors.

Methods of estimations of errors shall be discussed in the next section.

### 4.1.1 <u>Principles of Estimation of local Errors</u>

There are many forms of error estimation, each depends on the type of integration schemes used.

However, in the case of one step schemes, there are two primary approaches. The first was suggested by Felhberg (1964). It requires that we compute two approximations to y with implicit rational Runge - Kutta of orders P and P+1. The differences give the local error estimates.

The efficiency of this form of error can be justified for large stiff systems of ordinary differential equations when the Lipschitz constant is very large.

However, it's only disadvantage is the need to solve some implicit systems of equation at each step.

The second form of error estimate that can be considered for stiff Ordinary differential equations solver is one based on Richardson extrapolation process where by the local truncation error is estimated from the difference between two predictions using the same scheme but with different step sizes.

If $y(x_{n+1})$ designate the solution by single step size h, the local truncation error $T_{n+1}$ in equation (4.2) and (4.7) can be written as

$$y(x_{n+1})-y_{n+1} = \psi(x_n, y(x_n), h)h^{p+1}+0(h^{p+2})\ldots(4.12)$$

where p is refered to as the order of accuracy and $y_{n+1}$ denote the approximate value of y at $x_{n+1}$ with h as step size.

Suppose, we compute another approximation $L_{n+1}$ to $y(x_{n+1})$ by applying scheme (3.7) with h/2 as step size. Then it can be shown that

$$y(x_{n+1})-L_{n+1}=\psi(x_n, y(x_n)\frac{h}{2})(\frac{h}{2})^{p+1}+0(h^{p+2})\ldots\ldots\ldots\ldots(4.13)$$

Subtracting (4.12) from (4.13) and simplifying, we obtain

$$\psi(x_n, y(x_n))h^{p+1}=[y_{n+1}-L_{n+1}][1-\frac{1}{2^{p+1}}]^{-1}\ldots(4.14)$$

Therefore, the accuracy of the scheme can be estimated from

$$\Delta = |[y_{n+1}-L_{n+1}][1-\frac{1}{2^{p+1}}]^{-1}|\ldots.(4.15)$$

Thus,

$$\Delta=|\psi(x_n,\ y(x_n), h)|h^{p+1}\ldots\ldots(4.16)$$

The approximation $y_{n+1}$ from step $x_n$ to $x_{n+1}$ is accepted as the approximation to the exact solution. If the accuracy is less than error tolerance (Tol).

That is

$\Delta$ < Tol.................................................... (4.17)

Then this form of error estimate has been found to be adequate (Lambert,1963) for stiff initial value problem in ODES. This estimate can then be used to choose a reasonable step size that will accelerate convergence. However, it involves considerable amount of computational efforts.

## 4.2 THE CONVERGENCE PROPERTY

The numerical scheme (3.7) for solving ordinary differential equations (1.4) will be said to be Convergent, if the numerical approximation $y_{n+1}$ that is generated by it tends to the exact solution $y(x_{n+1})$ of the ordinary differential equation (1.4) as the step size tends to zero.

That is $\lim\limits_{n \to \infty} [y(x_{n+1}) - y_{n+1}] = 0$

To analyse the Convergence of the proposed scheme, we consider the following standard theorem which we state without proof.

**Theorem 1:** Let $\{e_i, j = 0(1)n\}$ be sets of real number, if there exist finite constants R and S such that

$$|e_j| < R|e_{j-1}| + s, j = 0(1)n-1 \ldots (4.18)$$

then

$$|e_j| \le \left(\frac{R^j-1}{R-1}\right)s+R^j|e_o|, R \neq 1 \ldots\ldots\ldots (4.19).$$

Let $e_{n+1}$ and $T_{n+1}$ denote the discretization and truncation errors generated by (3.7) respectively.

Adopting binomial expansion and ignoring higher terms in equation (3.7) and (4.2), we obtain

$$y(x_{n+1}) = y(x_n) + h\psi_2(x_n, y(x_n);h) + h\Phi_1(x_n, y(x_n);h) + (higher\ terms) + T_{n+1} \ldots\ldots\ldots\ldots (4.20)$$

where $\Phi_1, \psi_2$ and $\Phi_2$ are continuous functions in the domain $a \le x \le b$, $|y| < \infty$, $O \le h \le h_0$ defined as

$$h\Phi_1(x_n, y(x_n);h) = \sum_{i=1}^{r} W_i K_i \ldots\ldots (4.21).$$

$$h\Phi_2(x_n, z(x_n);h) = \sum_{i=1}^{r} V_i H_i$$
$$= -\frac{h}{y^2(x_n)}\psi_2(x_n, y(x_n);h) \ldots\ldots (4.22)$$

where

$$\psi_2(x_n, y(x_n);h) = (1+y(x_n)\sum_{j=1}^{r} b_{ij}H_j)$$
$$(\Phi_2(x_n, y(x_n);h) \ldots (4.23)$$

(From (4.3) - (4.6) ).

Similarly equation (3.7) yields

$$y_{n+1} = h\psi_2(x_n, y_n; h) + h\phi_1(x_n, y_n; h) + higher\ terms \ldots (4.24)$$

Subtract equation (4.20) from (4.24) and use equation (4.11) leads to

$$e_{n+1} = e_n + h[\psi_2(x_n, y(x_n); h) - \psi_2(x_n, y_n; h)] + h[\Phi_1(x_n, y(x_n); h) - \Phi_1(x_n, y_n; h)] + T_{n+1} \ldots\ldots\ldots\ldots (4.25)$$

By taking the absolute value on both sides of equation (4.25), we have the inequality.

$$|e_{n+1}| \le |e_n| + kh|e_n| + hL|e_n| + T \ldots\ldots\ldots (4.26)$$

Where L and K are the Lipschitz constants for $\Phi_1(x,y;h)$ and $\psi_2$ $(x,y;h)$ respectively and

$$T = \underset{a \le x \le b}{\mathrm{Sup}} |T_{n+1}| \ldots\ldots\ldots (4.27)$$

By setting $N = L+K$ .......................................................... (4.28)

inequality (4.26) becomes,

$$|e_{n+1}| \le |e_n|(1 + hN) + T,\ n = 0,1 \ldots \ldots\ldots\ldots (4.29)$$

From theorem 1, expression (4.29) becomes

$$|e_n| \le \frac{(1+hN)^n - 1}{hN} + (1 + hN)^n |e_o| \ldots\ldots\ldots\ldots (4.30)$$

Since

$$(1+hN)^n - e^{nhN} = e^{N(xn-a)}.$$

and

$x_n \leq b$, then $x_n - a \leq b - a$

Consequently,

$$e^{N(x_{n-a})} \leq e^{N(b-a)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.31)$$

$$|e_n| \leq \left(\frac{e^{N(b-a)} - 1}{hN}\right) T + e^{N(b-a)} |e_o| \ldots\ldots\ldots (4.32)$$

Considering equation (4.40) and adopting first mean value theorem

$$\begin{aligned} T_{n+1} &= h[\psi_2(x_n + \theta h, y(x_n + \theta h)) - \psi_2(x_n, y(x_n)) \\ &\quad + h[\phi_1(x_n + \theta h, y(x_n + \theta h)) - \phi_1(x_n, y(x_n))] \\ &= h[\psi_2(x_n + \theta h, y(x_n + \theta h)) - \psi_2(x_n + \theta h, y(x_n)) \\ &\quad + \psi_2(x_n + \theta h, y(x_n)) - \psi_2(x_n, y(x_n))] \\ &\quad + h[\phi_1(x_n + \theta h, y(x_n + \theta h)) - \phi_1(x_n + \theta h, y(x_n)), \\ &\quad + \phi_1(x_n + \theta h, y(x_n)) - \phi_1(x_n, y(x_n))], \quad 0 \leq \theta \leq 1 \ldots\ldots (4.33) \end{aligned}$$

By taking the absolute value of (4.33) on both sides, taking equation (4.27) into consideration, we have

$$T = hL|y(x_n + \theta h) - y(x_n| + jh^2\theta + hk|y(x_n + \theta h) - y(x_n)| + Mh^2\theta$$
$$T = h^2\theta N y^1(\xi) + (J + M)h^2\theta, \quad x n \leq \xi \leq x_{n+1} \ldots\ldots\ldots\ldots (4.34).$$

where M and J are the partial derivative of $\phi_1$ and $\psi_2$ with respect to x respectively.

By setting

$$Q = J + M \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.35)$$

and

$$Y = \sup_{a \leq x \leq b} |y'(x)| \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.36)$$

Therefore, equation (4.34) yields

$$T = h^2\theta\,(NY + Q) \quad \ldots\ldots \quad (4.37)$$

By substituting (4.37) into (4.32), we have

$$|e_n| \le h\theta e^{N(b-a)}(NY + I) + e^{N(b-a)}|e_o| \ldots\ldots \quad (4.38)$$

Assuming no error in the input data. That is, $e_o = 0$, then in the limit, as h tends to zero, we obtain

$$\lim_{\substack{h \to 0 \\ n \to \infty}} |e_n| = 0 \quad \ldots\ldots \quad (4.39)$$

which implies from (2.14) and (4.11) that

$$\lim_{\substack{h \to 0 \\ n \to \infty}} y_n = y(x_n) \quad \ldots\ldots \quad (4.40)$$

Thus establishing the Convergence of scheme (3.7).

Next section considers the stability properties of the scheme.

## 4.3 STABILITY PROPERTIES

As mentioned earlier that any error introduced at any stage of the computation which is not bounded can produced unstable numerical results.

Therefore, we consider the stability analysis of the proposed implicit rational R-K scheme defined in (3.7) to access its suitability.

To achieve this, we apply scheme (3.7) to Dalquist (1963) stability scalar test initial value problem.

$$y' = \lambda y\ ,\ y(x_o) = y_o \qquad (4.40)$$

under the assumption that $Re(\lambda) \ll 0$. This leads to a set of r simultaneous linear equations which when put in matrix form, is

$$\begin{bmatrix} 1-\lambda ha_{11} & -\lambda ha_{12} \ldots\ldots & \lambda ha_{1r} \\ -\lambda ha_{21} & (1-\lambda ha_{22}) \ldots\ldots & -\lambda ha_{2r} \\ \cdot & \cdot & \\ \cdot & \cdot & \\ -\lambda ha_{r1} & -\lambda ha_{r2} \ldots\ldots & (1-\lambda ha_{rr}) \end{bmatrix} \begin{bmatrix} K_1 \\ K_2 \\ \cdot \\ \cdot \\ K_r \end{bmatrix} = \lambda hy_n \begin{bmatrix} 1 \\ 1 \\ \cdot \\ \cdot \\ 1 \end{bmatrix} \qquad (4.41)$$

In compact form equation (4.41) becomes

$$DK = Y \qquad (4.42)$$

where

$$Y = \lambda h\, y_n [1\ 1 \ldots\ldots 1]^T$$

and

$$D = \begin{bmatrix} 1-\lambda ha_{11} \ldots\ldots & & -\lambda ha_{1r} \\ -\lambda ha_{21} \ldots\ldots & (1-\lambda ha_{22}) \ldots\ldots & -\lambda ha_{21} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ -\lambda har_1 \ldots\ldots & & (1-\lambda ha_{rr}) \end{bmatrix} \qquad 4.43$$

By assuming $D^{-1}$ exist, then

$$K = D^{-1} Y \qquad (4.44)$$

Thus, the $K_i$'s are obtained from (4.43) as

$K = (I - A\hbar)^{-1} e \hbar y_n$ .................................................................................... (4.45)

Similarly,

$H = -(I + \hbar B)^{-1} e \hbar Z_n$ ...........................................................(4.46)

where

$A = \{a_{ij}\}, \quad i, j = 1(1)r$

$B = \{b_{ij}\}, \quad i,j = 1(r)r$

$e = [1, 1 \ldots\ldots\ldots 1]^T$

and $\hbar = \lambda h$,

$Z_n = 1/y_n$, ...............................................................(4.47)

and I an r x r unit matrix.

Substituted (4.44) and (4.45) into (3.7) we obtained a difference equation.

$$y_{n+1} = \left(\frac{1 + \hbar W^T (I - \hbar A)^{-1} e}{1 - \hbar V^T (1 + \hbar B)^{-1} e}\right) y_n \ldots\ldots\ldots\ldots (4.48)$$

where

$W^T = (W_1, \ W_2 \ldots\ldots\ldots\ldots W_r)$

$V^T = (V_1 \ \ V_2, \ldots\ldots\ldots\ldots V_r)$.

The general resultant equation is the first order difference equation of the form.

$y_{n+1} = \mu(\hbar)\, y_n, \quad \hbar = \lambda h$ ............................................................ (4.49)

where,

$$\mu(\hbar) = \frac{1 + \hbar W^T (1 - \hbar A)^{-1} e}{1 - \hbar V^T (I + \hbar B)^{-1} e} \ldots\ldots\ldots\ldots (4.50)$$

which is called the stability function.

The parameters ($a_{ij}$, $b_{ij}$, $C_i$, $d_i$, $W_i$, $V_i$) in the scheme, (3.7) are chosen to ensure $\mu(\hbar)$ is a Pades approximation to $e^{\hbar}$. To understand the concept of Pade's approximation we give the following definitions.

**Definition 1**

Let $Q_r(\hbar)$ denote a polynomial of degree r in $\hbar$ specified as

$$Q_r(\hbar) = 1 + \frac{r^{\hbar}}{2r!} + \frac{r(r-1)}{2r(2r-1)}\frac{\hbar^2}{2!} + \frac{r(r-1)\ldots\ldots\ldots1}{2r(2r-1)\ldots\ldots(r+1)}\frac{\hbar^r}{r!} \ldots\ldots\ldots\ldots\ldots (4.51)$$

then

$$R_{r,s}(\hbar) = \frac{Q_r(\hbar)}{P_s(\hbar)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.52)$$

where $P_s(\hbar)$ denote a polynomial of degree s. Then we say $R_{r,s}(\hbar)$ is an (r, s) Pade's approximation to $e^{\hbar}$

Thus $R_{r,s}(\hbar)$ is rational approximation of order r + s to exponential of $\hbar$ 1f

$$R_{r,s}(\hbar) = e^{\hbar} + 0(\hbar^{r+1}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.53)$$

which can be achieved by expressing $e^{\hbar}$ as a power series in $\hbar^i$ and equating the Coefficient of $\hbar^i$, i = 1 (1) r+s in equation.

$$\sum_{i=1}^{n} a_i\hbar^i = \left(\sum_{i=0}^{s} b_i\hbar^i\right)\left(\frac{\hbar^i}{i!}\right) \ldots\ldots\ldots\ldots\ldots (4.54)$$

to define uniquely the Coefficient $a_i$s, $b_i$'s appearing in $R_{r,s}(\hbar)$.

The usefulness of this deduction will be manifested later in this chapter.

**Definition 2:**

The scheme (3.7) is said to be absolutely stable at a point $(\hbar, \mu(\hbar))$ in the complex plane, if the stability function (4.48) satisfies,

$$|\mu(\hbar)| < 1 \quad \text{.......... (4.55)}$$

The corresponding region R of absolute stability of the scheme can then be defined as

$$R = \{ \hbar \,/\, |\mu(\hbar)| < 1 \} \quad \text{.......... (4.56)}$$

**Definition 3 :**

The numerical scheme (3.7) is said to be A - stable if the region of absolute stability specified in (4.56) includes the entire half of the complex plane denoted by

$$AS = \{ \hbar \in C \,/\, Re(\hbar) < 0 \} \quad \text{.......... (4.57)}$$

A - stability property is one of the desirable properties for any numerical schemes as suggested by Dalquist (1963), particularly if the initial value problems are both stiff and stiff Oscillatory.

Other stability criterion that qualify numerical scheme to perform effectively and efficiently on stiff equations include L - stability and A ($\propto$) stability which normally are associated with one step schemes.

These concept are explained briefly in the following definitions.

**Definition 4:**

The integration scheme (3.7) is said to be A ($\propto$) stable, $\propto \in (0, \pi/2)]$, if its region of absolute stability contains infinite wedge $S\alpha$ define by

$S\alpha = \{\hbar / |Arg(-\hbar)| < \propto, \hbar \neq 0\}$ ........................................ (4.58).

The largest $\alpha$, is called the angle of absolute stability . The region of $S\alpha$ is shown in figure (4.1) below.

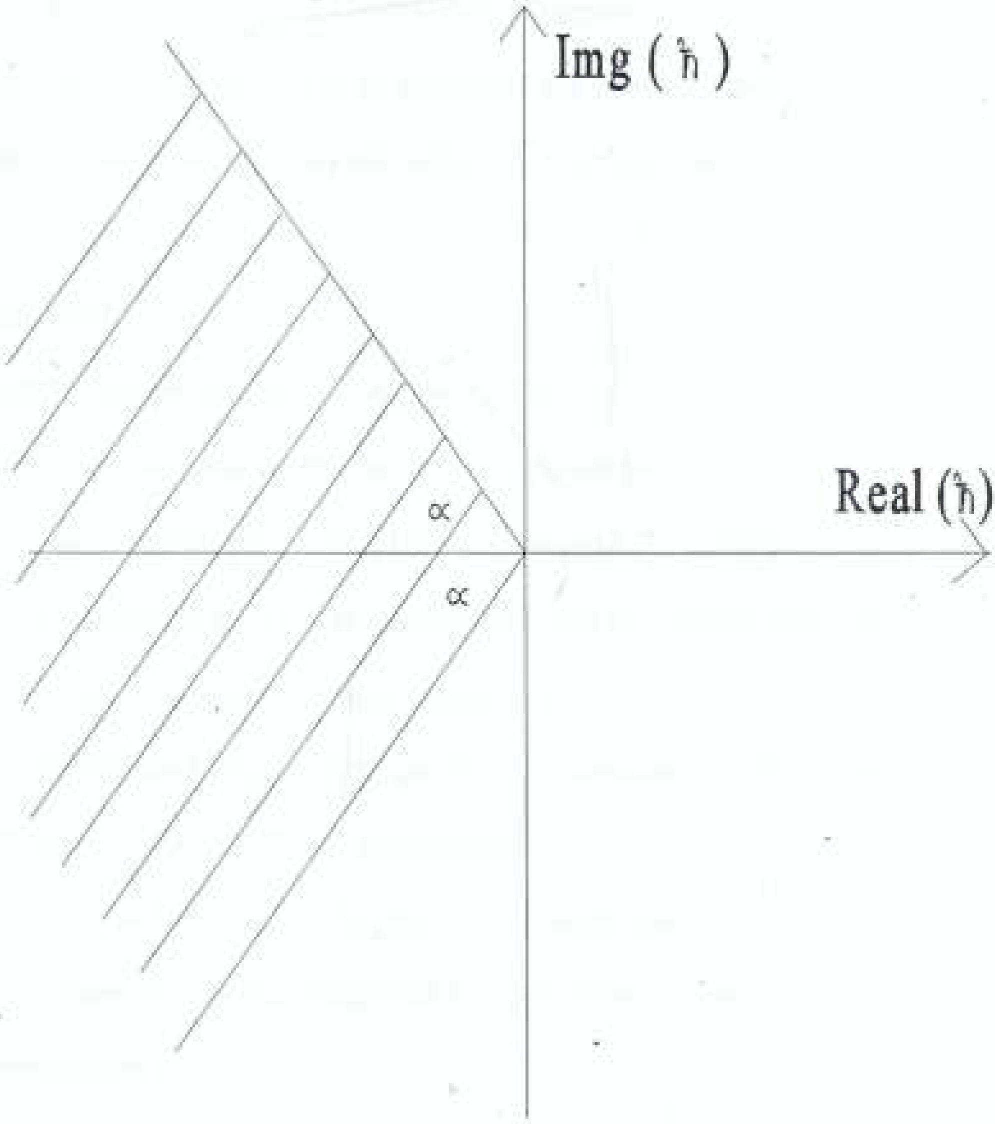
In order to apply $A(\alpha)$ stability conditions, we verify wether the eigenvalues of the systems lie within a certain wedge $S\alpha$. It is said to be A(o) stable if it is $A(\alpha)$ stable for some $\alpha \in [0, \pi/2]$.

**Definition 5:**

The numerical scheme (3.7) is said to be L - stable if it is A - stable and in addition it's stability function (4.50) satisfies the condition.

$\lim_{\hbar \to -\infty} |\mu(\hbar)| = 0$ ................................................ (4.59)

An L - stable method is necessarily A - stable and A ($\propto$) - stable. Based on this recognition we shall establish the L - stabillity property for the proposed




Region of A (∝) Stability

Fig 4.1

schemes (in chapter three).

As mentioned earlier, to relate Pade's approximation to $e^{\hbar}$ with definitions above and theorem stated below which apparently highlight the adequacy of Pade's approximation to $e^{\hbar}$ in investigating the stability properties of the numerical schemes, we now consider the following definitions and theorems.

**Definition 6:**

A Pade's approximation to $e^{\hbar}$ is said to be

I A - acceptable If $|R_{r,s}(\hbar)| < 1$ whenever $Re(\hbar) < 0$.

Ii A(o) - acceptable if $|R_{r,s}(\hbar)| < 1$ whenever $\hbar$ is real and negative.

Iii L - acceptable, If it is A - acceptable and in addition satisfies $|R_{r,s}(\hbar)|$ tends to zero as $Re(\hbar)$ tends to negative infinity.

Birkoff and Virga (1965) gave further theorem on $R_{r,s}(\hbar)$ as

I If $r = s$, $R_{r,s}(\hbar)$ is A - acceptable

ii If $r = s+1$ or $r = s+2$, $R_{r,s}(\hbar)$ is L - acceptable.

Linger and Willoughby (1967) considered stages one and two of Pade's approximation to $e^{\hbar}$

$$R_{1,1}(\hbar,\gamma) = \frac{1 + \frac{1}{2}(1-\gamma)\hbar}{1-\frac{1}{2}(1+\gamma)\hbar} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.60)$$

and

$$R_{2,2}(\hbar;\gamma,\beta) = \frac{1 + \frac{1}{2}(1-\gamma)\hbar + \frac{1}{4}(\beta-\gamma)\hbar^2}{1 - \frac{1}{2}(1+\beta)\hbar + \frac{1}{4}(\beta+\gamma)\hbar^2} \quad \ldots\ldots (4.61)$$

with the following conditions.

i. $R_{1,1}(\hbar, \gamma)$ is A -acceptable, If and only if $\gamma > 0$, L-acceptable, if and only if $\gamma = 1$.

ii. $R_{2,2}(\hbar; \gamma, \beta)$ is A-acceptable if and only if $\gamma > 0$ and $\beta \geq 0$ and L - acceptable, if and only if $\propto = \beta > 0$

With the above definitions and theorems, we will now analyse the stability properties of the family of one - stage and two - stage of the proposed scheme.

### 4.3.1 One - stage schemes

To analyse, the stability properties, we shall adopt the definitions and theorems we discussed earlier. First we consider in equation (3.7) when r = 1. That is general one stage scheme of this form

$$y_{n+1} = \frac{y_n + W_1 K_1}{1 + y_n V_1 H_1} \qquad (4.62)$$

where

$K_1 = hf(x_n + c_1 h,\ y_n + a_{11} k_1)$.

$H_1 = hg(x_n + d_1 h,\ z_n + b_{11} H_1)$.

$g(x_n, z_n) = -z_n^2 f(x_n, y_n)$.

applying (4.62) to the stability test equation (4.39), we obtain the recurrent relation

$$y_{n+1} = \left(\frac{1 + W_1^T \hbar (1 - a_{11}\hbar)^{-1}}{1 - V_1 \hbar (1 + b_{11}\hbar)^{-1}}\right) y_n \qquad (4.63)$$

For the approximation to the solution for the convergence of the solution we consider the function.

$$\mu(\hbar) = \frac{1 + W_1\hbar(1 - a_{11}\hbar)^{-1}}{1 - V_1\hbar(1 + b_{11}\hbar)^{-1}} \ldots\ldots\ldots\ldots\ldots (4.64)$$

which can be shown to satisfy the Pade's approximation to $e^{\hbar}$, that is If $\mu(\hbar)$ can be expressed in the form

$$\mu(\hbar) = \sum_{i=1}^{2} a_i\hbar^i + 0(\hbar^3) \ldots\ldots\ldots\ldots\ldots (4.65)$$

For example, the associated stability function for (3.41) - (3.45) is

$$\mu(\hbar) = \frac{1 + \frac{1}{2}\hbar}{1 - \frac{1}{2}\hbar} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.66)$$

which is (1,1) Pade's approximation to $e^{\hbar}$ (see equation 4.50 of definition 1) since

$$\mu(\hbar) = 1 + \hbar + \tfrac{1}{2}\hbar^2 + 0(\hbar^3) \ldots\ldots\ldots (4.67)$$

However the stability function (4.64) satisfies (4.52) with $(-\infty, 0)$ as the corresponding interval, of absolute stability. This implies that the schemes are A - stable (see definition 3, equation 4.53).

This property can stimulate the use of the schemes to solve stiff ODES.

Also according to Liniger and Willoughby (1976). One stage family of (3.41) - (3.45) are A - acceptable since,

$$|R_{11}(\hbar)| \leq 1 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.68)$$

with Re ($\bar{h}$) <0 and A(o) acceptable since it satisfies (4.66) with negative real $\bar{h}$.

These schemes are also A(o) stable since it is A($\propto$) stable with $\propto = 0°$ (see fig 4.2 below).

**Two stage schemes**

From (3.7), a two - stage scheme is

$$y_{n+1} = \frac{y_n + W_1K_1 + W_2K_2}{1 + y_n(V_1H_1 + V_2H_2)} \quad \ldots\ldots\ldots\ldots\ldots\ldots (4.69)$$

where

$k_1 = hf(x_n + c_1h,\ y_n + a_{11}\ k_1 + a_{12}\ k_2)$ ............................................ (4.70)

$k_2 = hf(x_n + c_2h,\ y_n + a_{12}k_1 + a_{22}\ k_2)$............................................(4.71)

$H_1 = hg\ (x_n + d_1h,\ y_n + b_{11}\ H_1 + b_{12}\ H_2)$............................................(4.72)

$H_2 = hg\ (x_n + d_1h, y_n + b_{12}\ H_2 + b_{22}\ H_2)$............................................(4.73)

Applying this formular to the test equation (4.40), we obtained systems of linear equations for $K_i$' s which when written in matrix notation yields

$$\begin{bmatrix} 1 - \lambda h a_{11} & -a_{12}\lambda h \\ -a_{21}\lambda h & 1-\lambda h\ a_{22} \end{bmatrix} \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} = \begin{bmatrix} 1\lambda h y_n \\ 1\lambda h y_n \end{bmatrix} \quad \ldots\ldots\ldots\ldots (4.74)$$

which in compact form becomes

$DK = Y$ ............................................................................................(4.75)

where

$$D = \begin{bmatrix} 1 - \lambda h a_{11} & -\lambda h a_{12} \\ -\lambda h a_{21} & 1 - \lambda h a_{22} \end{bmatrix} \quad \text{.......... (4.76)}$$

and $Y = [1 \quad 1]^T hy_n$ .......... (4.77)

By assuming that $D^{-1}$ exist, equation (4.75) becomes

$$K = D^{-1} Y \quad \text{.......... (4.78)}$$

or

$$K = \hbar(I - A\hbar)^{-1} e y_n \quad \text{.......... (4.79)}$$

Similarly, equations (4.72) and (4.73) yield

$$H = (I + B\hbar)^{-1} e Z_n \quad \text{.......... (4.80)}$$

where

$e = [1, 1]^T, \hbar = \lambda h.$

$A = [a_{ij}], \quad B = (b_{ij}), \; i, j = 1 \, (1) \, 2$

I being (2 x 2) unit matrix

Putting (4.77) and (4.78) into (4.67) we obtain

$$y_{n+1} = \frac{1 + \hbar W^T (I - \hbar A)^{-1} e}{1 - \hbar V^T (I + hB)^{-1} e} \, y_n \quad \text{.......... (4.81)}$$

which is a difference equation with stability function

$$\mu(\hbar) = \frac{1 + \hbar W^T (I - \hbar A)^{-1} e}{1 - hV^T (I + \hbar B)^{-1} e} \quad \text{.......... (4.82)}$$

where

$W^T = [W_1 \; W_2]$

$V^T = [V_1, V_2]$.

As usual, it can be shown to satisfy a (2,2) Pade's approximation to $e^{\hbar}$ If we simplify (4.82) further, the general stability function for family of two stage schemes becomes

$$\mu(\hbar) = \frac{1 + \dfrac{(W_1 + W_2)\hbar + (W_1(a_{12} - a_{22}) + W_2(a_{21} - a_{11})\hbar^2}{1 - (a_{11} + a_{22})\hbar + [(a_{11}a_{22} - a_{12}a_{21})]\hbar^2}}{1 - \dfrac{(V_1 + V_2)\hbar + [V_1(b_{12} + b_{22}) + V_2(b_{21} + b_{11})]\hbar^2}{1 + (b_{11} + b_{22})\hbar + (b_{11}b_{22} - b_{12}b_{21})\hbar^2}} \quad \ldots (4.83$$

For example, the stability function of scheme (3.76) is

$$\mu(\hbar) = \frac{1 + \frac{1}{2}\hbar + \frac{1}{2}\hbar^2}{1 - \frac{1}{2}\hbar - \dfrac{5}{12}\hbar^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.84)$$

with parameters

$$V_1 = V_2 = \tfrac{1}{2}, \quad W_1 = W_2 = 0$$
$$C_1 = d_1 = \tfrac{1}{2} - \frac{\sqrt{3}}{6}, \quad C_2 = d_2 = \tfrac{1}{2} + \frac{\sqrt{3}}{6}$$
$$b_{11}\ b_{22} = a_{11} = a_{22} = \tfrac{1}{4}$$
$$b_{12} = a_{12} = \tfrac{1}{4} + \frac{\sqrt{3}}{6}, \quad a_{21} = b_{21} = \tfrac{1}{4} - \frac{\sqrt{3}}{6}$$

which is a (2, 2) Pade's approximation to $e^{\hbar}$, since it can be expressed as

$$\mu(\hbar) = 1 + \hbar + \frac{3}{4}\hbar^2 + \frac{1}{4}\hbar^3 + \frac{5}{144}\hbar^4 + 0(\hbar^5) \quad \ldots\ldots\ldots\ldots (4.85)$$

This scheme is A - stable with ( $-\infty$, 0) as corresponding interval of absolute stability since it satisfies

$$\lim_{\hbar \to -\infty} |\mu(\hbar)| < 1 \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.86)$$

The scheme is also A - acceptable since $R_{2,2}(\hbar)$ is less than 1 with Re $(\hbar)$ < 0. It is also A(o) acceptable since $|R_{2,2}(\hbar)| < 1$ with negative real part (i.e Re $(\hbar)$ = -3). It is also A($\propto$) stable with $\propto = 33.6°$.

The stability function of scheme (3.75) with

$$V_1 = V_2 = 0, \quad W_1 = W_2 = ½,$$
$$C_1 \; d_1 = \frac{1}{2} + \frac{\sqrt{3}}{6}, \quad C_2 = d_2 = \frac{1}{2} - \frac{\sqrt{3}}{6}$$
$$b_{11} = b_{22} = a_{11} = a_{22} = \frac{1}{4}$$
$$b_{12} \; a_{12} = \frac{1}{4} - \frac{\sqrt{3}}{6}, \quad a_{21} = b_{21} = \frac{1}{4} + \frac{\sqrt{3}}{6}$$

is

$$\mu(\hbar) = \frac{1 + \frac{1}{2}\hbar + \frac{1}{12}\hbar^2}{1 - \frac{1}{2} + \frac{1}{12}\hbar^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.87)$$

This scheme is A - stable with interval of absolute stability $(-\infty, O)$. Since it satisfies (4.84), thus it is A - acceptable and A(o) acceptable since Re$(\hbar) < 1$ with negative real part i.e

(Re $(\hbar)$ = - 17.31) . It is A ($\propto$) stable with $\propto = 11.3°$.

The stability function of scheme (3.65) is obtained as

$$\mu(\hbar) = \frac{1 + \frac{1}{3}\hbar}{1 - \frac{2}{3}\hbar - \frac{1}{3}\hbar^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.88)$$

by setting

$$W_1 = W_2 = 0, \quad V_1 = \frac{1}{4}, \; V_2 = \frac{3}{4}$$
$$C_1 = d_1 = a_{12} = b_{12} = 1$$
$$a_{11} = b_{11} = a_{21} = b_{21} = 0$$
$$C_2 = d_2 = a_{22} = b_{22} = \frac{1}{3}$$

This scheme is L - stable and A - stable since it satisfies

$$\lim_{\bar{h} \to -\infty} |\mu(\bar{h})| = 0 \qquad \text{.......} \qquad (4.89)$$

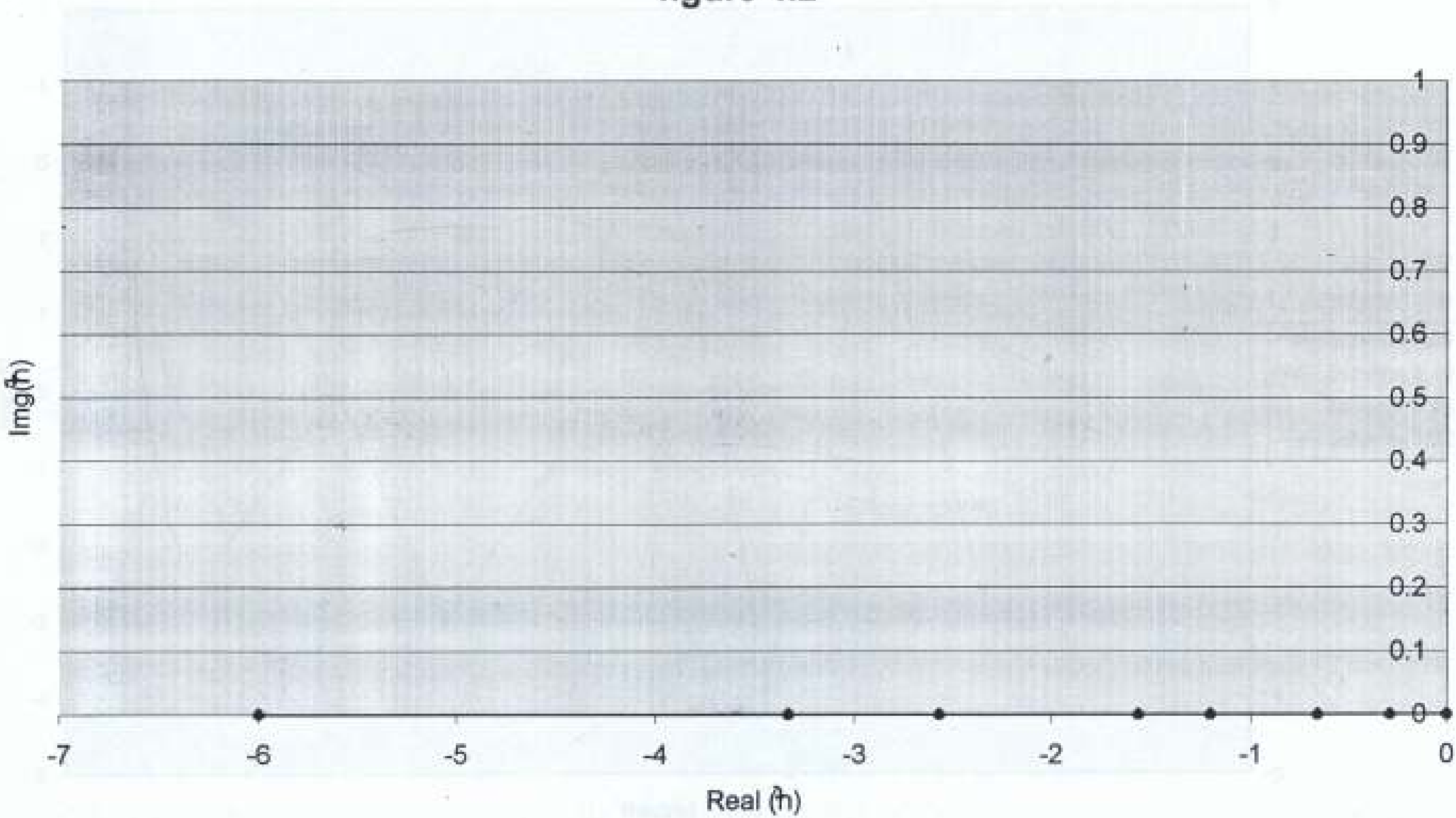
and, equation (4.86). It is A ($\alpha$) stable with $\alpha = 45°$.

The stability region of family of one-stage schemes of order two and two-stage scheme of order three and four are shown below in fig. 4.2, 4.3 & 4.4 respectively.

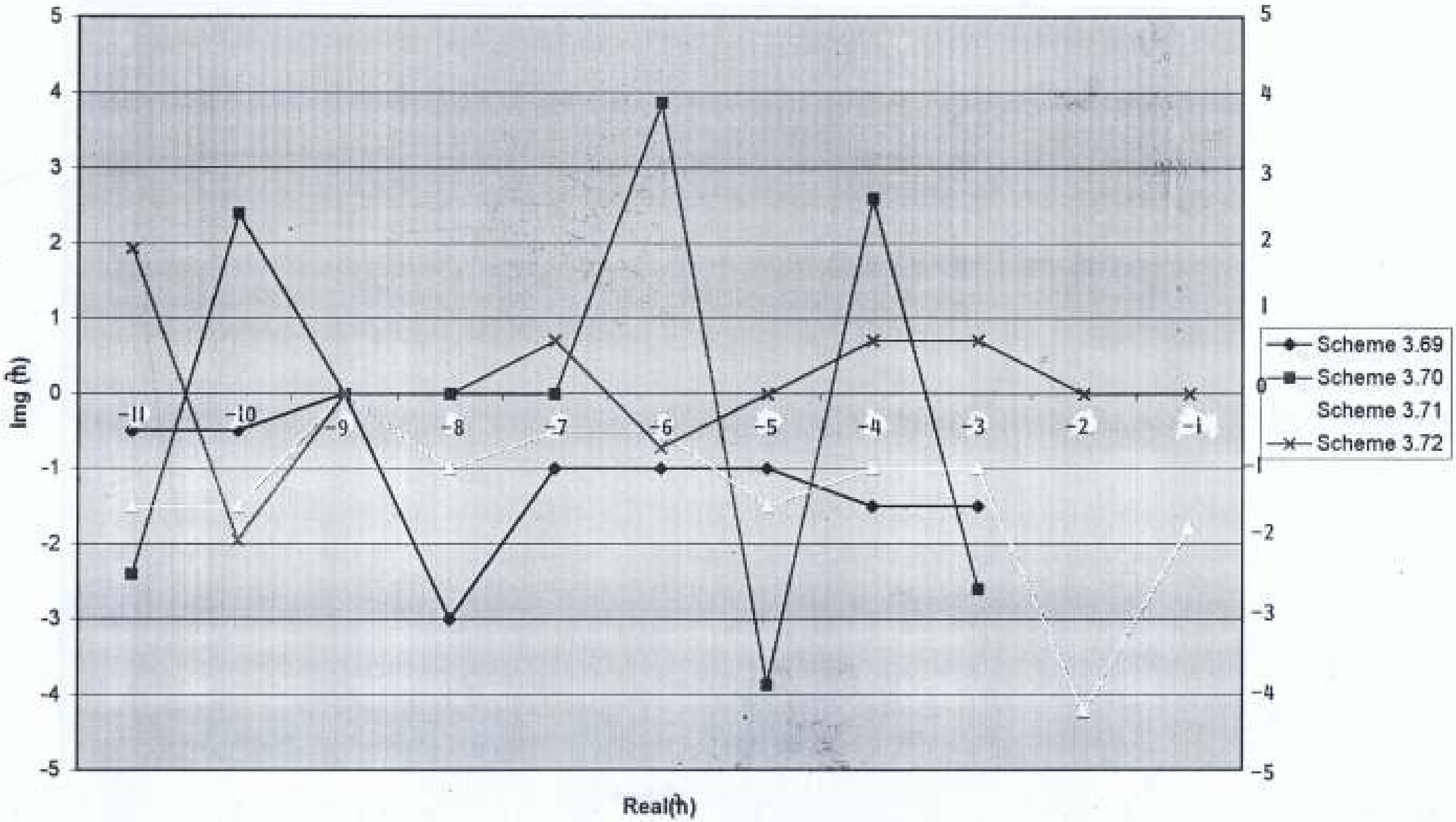
These large stabillity properties encourage us to adopt the use of the proposed schemes for solving stiff ODES.

The importance of the properties will be manifested in the implementation of the schemes in Chapter five.

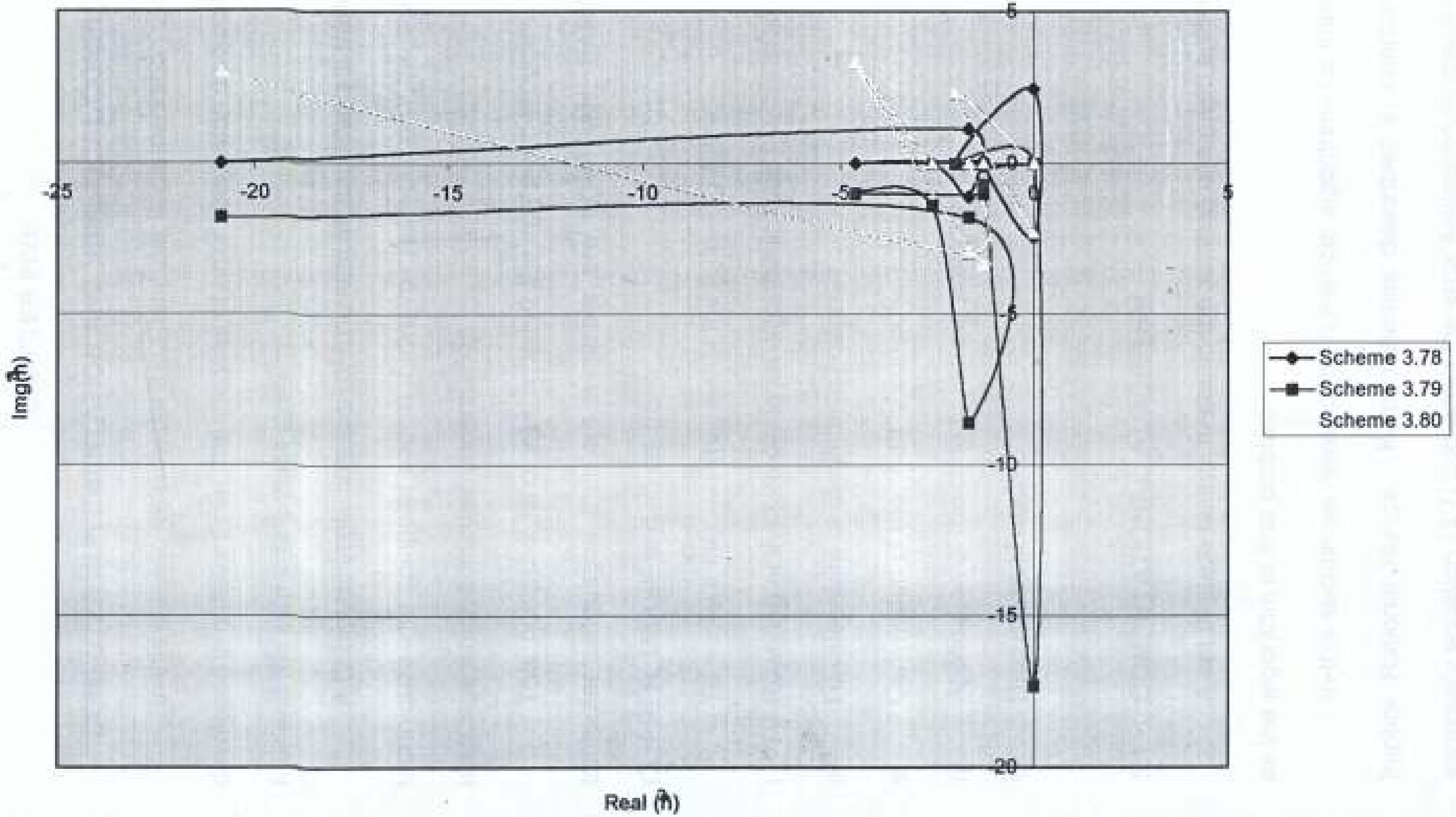
**Region of Absolute stability of family of one stage schemes of order two**
**figure 4.2**



**Region of Absolute Stability of family of Two-Stage of order three**
**Fig 4.3**



**Region of Absolute stability of family of two-stage schemes of order four.**
**figure 4.4**



## CHAPTER FIVE

## COMPUTER IMPLEMENTATION AND NUMERICAL COMPUTATION.

In this chapter, we consider the Computer Implementation of the schemes developed in chapter three, the theories of error estimation discussed in chapter four and the step size control strategy.

A Computer Implementation of a numerical formular is a translation of the techniques into computer codes. It involves the writting of the algorithm in the language which computer understands.

This computer language is called the computer program. Some of the computer languages that are available are FORTRAN, BASIC, PASCAL, CLIPPER, DBASE and so on.

To achieve this, we shall adopt the following steps:

i Rewriting of the formula in an algorithmic form.

ii Development of algorithmic flowchart

iii Conversion of the chart to computer code

iv Application of the code to sample problem

v Discussion of results.

### 5.1 COMPUTATIONAL ALGORITHM

A set of steps taking to obtain the solution of a given problem is defined as the algorithm of that problem.

In this section we develop the numerical algorithms for implementing the Implicit Rational Runge - Kutta schemes described in chapter three, most especially equation (3.7) and adopt the error estimation in chapter four and the step size control measures.

The algorithm is given below:

**STEP I**: Declaration of Variables

**STEP 2**: Define function

$F(x, y)$, yexact $(x,y)$.

**STEP 3**: Selection of input values

$X_o$, $x_{last}$, $y_o$, $h_{old}$, tol,

**STEP 4**: Initialise Variables by setting

m = 0 (counter)

$X = X_o$

$Y = y_o$

$H = h_{old}$

$P = 0$

**STEP 5**: Compute the approximate value of $y(x)$ adopting routine

IMPRRK $(x, y, h_{old}, tol, y_{n+1})$

For i = 1, 2, do

Set $K_i^{(P)} = 0.0$

$H_i^{(P)} = 0.0$

and for j = 1, 2 define

$d_i = D$, $c_i = C$, $a_{ij} = A$, $b_{ij} = B$

Set $z_n = 1/y_n$.

$z_{ni} = z_n$ + sum B * $H_i^{(P)}$.

$y_{ni} = x_n$ + sum A * $K_i^{(P)}$

$X1_i = x_n + C.h$

$X2_i = X_n + D.h$

$y_i = 1/z_{ni}$

Set $k_i^{(P+1)} = hf(x1_i, y_{ni})$.

$H_i^{(P+1)} = -hZ_{ni}^2 f(x2_i, y_i)$.

While ((abs($K_i^{(P+1)} - K_i^{(P)}$) and ($H_1^{(P+1)} - H^{(P)}$)) < Tol

Then, For i = 1, 2 do

$Q^{(P+1)} = y_n$ + Sum $W_i K_i^{(P+1)}$.

$R^{(P+1)} = 1 + y_n$ . Sum $V_i H_i^{(P+1)}$

$y_{n+1} = Q^{(P+1)}/R^{(P+1)}$.

$X_{n+1} = X_n + h$

$y(x_{n+1})$ = Y exact ($X_{n+1}$, $Y_{n+1}$)

Return the results.

Else, for i = 1, 2 do

Set $K_i^{(P)} = K_i^{(P+1)}$

$H^{i(P)} = H_i^{(P+1)}$.

Repeat step 5

**STEP 6**: Estimate the Local Truncation Error (L.T.E)

Using subroutine ADAPT (x, y, h, tol, Lte, $H_{new}$, $y_{n+1}$).

Set $y_{new} = y + hf(x, y)$.

$D_{new}$ = Abs (tol * $y_{new}$)

Call IMPRRK ($x_o$, $y_o$, h, tol, $y_{ni}$, )

Set $h_2 = 0.5 \times h_{old}$

Call IMPRRK ($x_o$, $y_o$, $h_2$, tol, $yn_2$).

Set $X_1 = X_o + h_2$.

Call IMPRRK ($x_1$, $y_{n2}$, $h_2$, tol, $y_{n3}$).

Set

$$LTe = \left(1 - \frac{1}{2^{r+1}}\right)^{-1} (y_{n3} - y_{n1})$$

$D_{old}$ = Abs ($y_{n3} - y_{n1}$).

$D_{np}$ = ($D_{new}/D_{old}$)

While ($D_{old} < D_{new}$).

then

Set $h_{new} = h_{old} \times (D_{np})^{0.2}$

else

Set

$h_{new} = h_{old} \times (D_{np})^{0.25}$.

**STEP 7**: While (LTE < tol)

then

Return the results

Else

**STEP 8**: Adjust the step size and Replace $h_{old}$ by $h_{new}$

$h_{old} = h_{new}$

and repeat step 6 and 7

**STEP 9** Output the results

**STEP 10**: Stopping Criterion

While ($X_n < X_{last}$). Then set

$X_n = X_{n+1}$

$y_n = y_{n+1}$

$h_{old} = h_{new}$

$m = m+1$

and repeat step 5 - 10.

Else

**STEP 11**: Stop

For N systems of equations we can adopt Fehlberg technique with routine called IMPRRK43 (n, a, x, y, h, tol, $h_{new}$, Lte, $y_{new}$) to estimate the local truncation error and control the step size.

For proper understanding of the operation structure of the above algorithm, we normally employ the so-called computer flowchart.

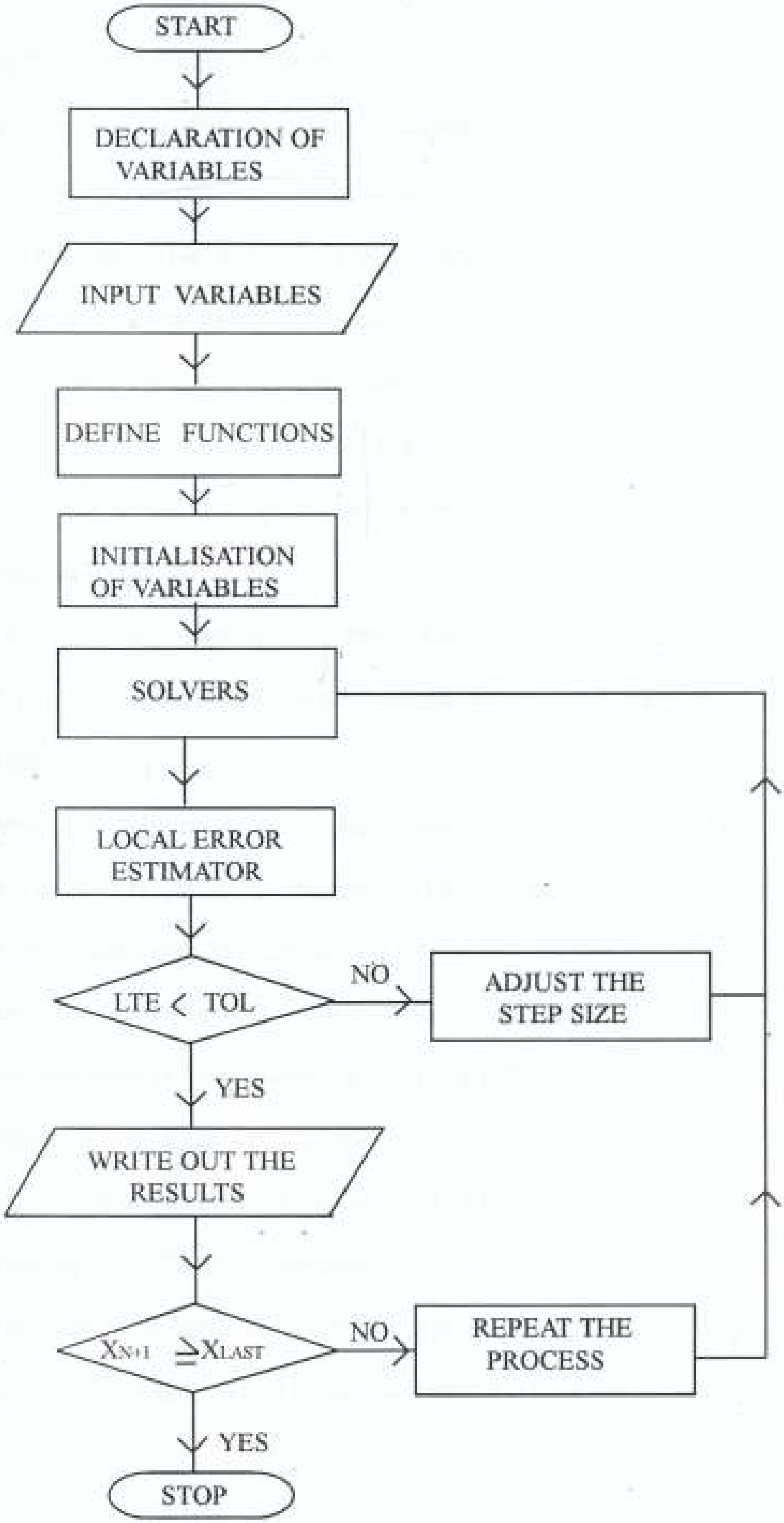
## 5.2 PROGRAM FLOW CHART

A computer flow chart is a diagramatic representation of the algorithm or the plan of solution of a problem. It indicates the process of solution, the relevant operation and computations, the point of decision, and other information which is a part of the solution.

Flow charts are of particular interest because of its documenting feature. They are constructed by using special geometrical symbols, such as squares, rectangles, diamonds shapes or circles. Each symbol represent some activities which could be input/output of data, taking a decision, terminating the solution and so on.

The symbols are joined by directed line segments to indicate direction of flow. The flow chart of the above algorithms is given in fig. 5.1.

FLOW CHART OF THE IMPLEMENTATION



## 5.3 PROGRAMMING IMPLEMENTATION

This section considers the computer implementation of the above algorithm. The implementation is done in a variable step size fixed order method. The program consists of three modules called FUNC, IMPRRK, ADAPT respectively and (IMPRRK43) for system of equation.

The program starts by declaring the value of the variables in double precision mode in order to reduce round off error. After this , the program chooses initial estimate for the variables and function FUNC was defined as a function subprogram to evaluate f(x,y).

This was followed by adopting the solver called IMPRRK to call on subroutine FUNC to evaluate the slopes of the integral curve of the solution on the receipt of the estimates.

IMPRRK generates the approximates solution $y_{n+1}$ to y(x) at $x_{n+1}$ and called on ADAPT to estimate the value of the error (LTE) associated with the computation, control the step size to increase the rate of convergence.

On the receipt of the error estimate from the subroutines ADAPT, IMPRRK then test for the convergence of the solution by comparing the magnitude of the error with allowable error tolerance. As soon as the condition $|LTE| < TOL$ is met, the program will ask whether the upper point is being reached, if yes, output result. Otherwise, ADAPT computes another error with that step size.The process will be repeated and continues until the upper end point is reached. When the program reached the endpoint ($x_{last}$) it will stop the process.

## 5.4 NUMERICAL COMPUTATIONS AND RESULTS

To demonstrate the applicablility of the various theories discussed (in previous chapters) we consider the following sample problems.

**Problem 1;**

Consider initial value problem

$$y' = \lambda(y - x^3) + 3x^2,\ y(o) = 1 \quad \text{.........} \quad (5.14)$$

where $\lambda$ is a complex constant with negative real part i.e. Re ($\lambda$) << O.

The theoretical solution is

$$y(x) = x^3 + e^{\lambda x} \quad \text{.........} \quad (5.15)$$

which has a rapidly decaying component $e^{\lambda}x$ (transient) as well as slowly decaying components $x^3$ (steady state).

The numerical results for $\lambda = -10$, $\lambda = 100$, $\lambda = -1000$ are shown in tables 1- 3. Also local error estimates adopting Richardson extrapolation techniques with $y(x_n)$, $y_n$, $e_n$, LTE representing exact solution, numerical soution, local discretization error, and local truncation error respectively, as shown in table 4. We also consider system of equations (5.16) by adopting Fehlberg (1964) techniques to estimate the local truncation error.

**Problem 2 :**

Consider the stiff system of ODEs of the form

$$Y' = AY \quad \text{.........} \quad (5.16)$$

where

$$A = \begin{bmatrix} 1.0 & -4.99 & 0 \\ 0 & -5.0 & 0 \\ 0 & 7.0 & -12.0 \end{bmatrix} \qquad (5.17)$$

with initial conditions

$$y(0) = [\ 2 \quad 1 \quad 2\ ]$$

in the theoretical interval $0 \le x \le 1$.

The eigen values of the Jacobian of (5.16) are $\lambda_1 = -1$, $\lambda_2 = -5.0$, $\lambda_3 = 12.0$ while the theoretical solutions are given as

$$y_1(x) = e^{-x} + e^{-5x}$$

$$y_2(x) = e^{-5x}$$

$$y_3(x) = e^{-5x} + e^{-12x} \qquad (5.18)$$

The local truncation error (LTE) and the variable step size $h_n$ are computed as described earlier with initial step size $h_{old} = 0.001$.

The results are shown in table 5.

**Problem 3**:

Another stiff system of ODEs considered is

$$Y' = \begin{bmatrix} -0.5 & 0 & 0 & 0 \\ 0 & -1.0 & 0 & 0 \\ 0 & 0 & -9.0 & 0 \\ 0 & 0 & 0 & -10 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \qquad (5.19)$$

with $Y(0) = [1, \ 1 \ \ 1 \ \ 1]$

in the interval $0 \le x \le 1$

The results are shown in table 6.

# TABLE 1

RESULTS OF TWO STAGE IMPLICIT RATIONAL RUNGE - KUTTA SCHEME OF ORDERS FOUR AND CLASSICAL RUNGE - KUTTA SCHEME OF ORDER FOUR WITH $\lambda = -10$.

| H | YEXACT | PROPOSED TWO STAGE RRK METHOD OF ORDER FOUR $Y_N$ | E1 | CLASSICAL RUNGE - KUTTA OF ORDER FOUR $Y_N$ | E2 |
|---|---|---|---|---|---|
| .10000000D-01 | .36788044D+00 | .36842207D+00 | .54162770D-03 | .37500103D+00 | .71205901D-02 |
| .50000000D-02 | .60653078D+00 | .60655749D+00 | .26707574D-04 | .60677096D+00 | .24017443D-03 |
| .25000000D-02 | .77880080D+00 | .77880183D+00 | .10358951D-05 | .77880861D+00 | .781070515D-05 |
| .12500000D-02 | .88249690D+00 | .88249694D+00 | .37210565D-07 | .88249715D+00 | .24910879D-06 |
| .62500000D-03 | .93941306D+00 | .93941306D+00 | .12426106D-08 | .93941307D+00 | .78652557D-08 |
| .31250000D-03 | .96923323D+00 | .96923323D+00 | .23452584D-09 | .96923323D+00 | .24706548D-09 |
| .15625000D-03 | .98449644D+00 | .98449644D+00 | .75417450D-11 | .98449644D+00 | .77408080D-11 |
| .78125000D-04 | .99221794D+00 | .99221794D+00 | .15305301D-09 | .99221794D+00 | .24225066D-12 |
| .39062500D-04 | .99610137D+00 | .99610137D+00 | .99332942D-11 | .99610137D+00 | .75495166D-14 |
| .19531250D-04 | .99804878D+00 | .99804878D+00 | .60418337D-12 | .99804878D+00 | .22204460D-15 |
| .97656250D-05 | .99902391D+00 | .99902391D+00 | .15495538D-09 | .99902391D+00 | .00000000D+00 |
| .48828125D-05 | .99951184D+00 | .99951184D+00 | .19385937D-10 | .99951184D+00 | .00000000D+00 |
| .24414063D-05 | .99975589D+00 | .99975589D+00 | .24242830D-11 | .99975589D+00 | .00000000D+00 |
| .12207031D-05 | .99987794D+00 | .99987794D+00 | .30320191D-12 | .99987794D+00 | .00000000D+00 |

# TABLE 2

RESULTS OF TWO STAGE IMPLICIT RATIONAL RUNGE - KUTTA SCHEME OF ORDERS FOUR AND CLASSICAL RUNGE - KUTTA SCHEME O F ORDER FOUR WITH $\lambda = -100$.

| H | YEXACT | PROPOSED TWO STAGE RRK METHOD OF ORDER FOUR $Y_N$ | E1 | CLASSICAL RUNGE - KUTTA OF ORDER FOUR $Y_N$ | E2 |
|---|---|---|---|---|---|
| .10000000D-02 | .36787944+00 | .3684109D+00 | .54164391D-03 | .37500000D+00 | .71205589D-02 |
| .50000000D-03 | .60653066D+00 | .60655737D+00 | .26707970D-04 | .60677083D+00 | .24017362D-03 |
| .25000000D-03 | .77880078D+00 | .77880182D+00 | .10359074D-05 | .77880859D+00 | .78106786D-05 |
| .12500000D-03 | .88249690+00 | .88249694+00 | .37210967D-07 | .88249715D+00 | .24910811D-06 |
| .62500000D-04 | .93941306D+00 | .93941306D+00 | .12426234D-08 | .93941307D+00 | .78652351D-08 |
| .31250000D-04 | .9693323D+00 | .9693323D+00 | .23452629D-09 | .9693323D+00 | .24706492D-09 |
| .15625000D-04 | .98449644D+00 | .98449644D+00 | .75418560D-11 | .98449644D+00 | .77408080D-11 |
| .78125000D-05 | .99221794D+00 | .99221794D+00 | .15305313D-09 | .99221794D+00 | .24225066D-12 |
| .39062500D-05 | .99610137D+00 | .99610137D+00 | .96331831D-11 | .99610137D+00 | .75495166D-14 |
| .19531250D-05 | .99804878D+00 | .99804878D+00 | .60418337D-12 | .99804878D+00 | .22204460D-15 |
| .97656250D-06 | .99902391D+00 | .9902391D+00 | .15495538D-09 | .99902391D+00 | .00000000D+00 |
| .48828125D-06 | .99951184D+00 | .99951184D+00 | .19385937D-10 | .99951184D+00 | .00000000D+00 |
| .24414063D-06 | .99975589D+00 | .99975589D+00 | .24242830D-11 | .99975589D+00 | .00000000D+00 |
| .12207031D-06 | .99987794D+00 | .99987794D+00 | .30321091D-12 | .99987794D+00 | .00000000D+00 |

## TABLE 3

RESULTS OF TWO STAGE IMPLICIT RATIONAL RUNGE - KUTTA SCHEME OF ORDERS FOUR AND CLASSICAL RUNGE - KUTTA SCHEME OF ORDER FOUR WITH λ = -1000.

| H | YEXACT | PROPOSED TWO STAGE RRK METHOD OF ORDER FOUR $Y_N$ | E1 | CLASSICAL RUNGE - KUTTA OF ORDER FOUR $Y_N$ | E2 |
|---|---|---|---|---|---|
| .10000000D+00 | .36887944D+00 | .36940452D+00 | .52508172D-03 | .37603125D+00 | .71518088D-02 |
| .50000000D-01 | .60665566D+00 | .60668197D+00 | .26309584D-04 | .60689665D+00 | .24098742D-03 |
| .25000000D-01 | .77881641D+00 | .77881743D+00 | .10234944D-05 | .77882424D+00 | .78335668D-05 |
| .12500000D-01 | .88249886D+00 | .88249889D+00 | .36809087D-07 | 88249911D+00 | .24978363D-06 |
| .62500000D-02 | .96941331D+00 | .96941331D+00 | .12297566D-08 | .96941331D+00 | .78857242D-08 |
| .31250000D-02 | .96923326D+00 | .96923326D+00 | .23411650D-09 | .96923326D+00 | .24769542D-09 |
| .15625000D-02 | .98449644D+00 | .98449644D+00 | .75289774D-11 | .98449644D+00 | .77603479D-11 |
| .78125000D-03 | .99221794D+00 | .99221794D+00 | .15305257D-09 | .99221794D+00 | .24280578D-12 |
| .39062500D-03 | .99610137D+00 | .99610137D+00 | .96332942D-11 | .99610137D+00 | .75495166D-14 |
| .19531250D-03 | .99804878D+00 | .99804878D+00 | .60418337D-12 | .99804878D+00 | .22204460D-15 |
| .97656250D-04 | .99902391D+00 | .99902391D+00 | .15495538D-09 | .99902391D+00 | .00000000D+00 |
| .48828125D-04 | .99951184D+00 | .99951184D+00 | .19385937D-10 | .99951184D+00 | .11102230D-15 |
| .24414063D-04 | .99975589D+00 | .99975589D+00 | .24242830D-11 | .99975589D+00 | .00000000D+00 |
| .12207031D-04 | .99987794D+00 | .99987794D+00 | .30320191D-12 | .99987794D+00 | .11102230D-15 |

## TABLE 4

RESULTS OF TWO STAGE IMPLICIT RATIONAL RUNGE - KUTTA SCHEME OF ORDERS FOUR ,ADOPTING RICHARDSON EXTRAPOLATION METHOD TO ESTIMATE THE LOCAL TRUNCATION ERROR AND CONTROL THE STEP SIZE.

| STEP SIZE | VALUE OF X | YEXACT | NUMERICAL VALUE RESULTS | LOCAL TRUNCATION ERROR | ERROR DISCRETIZATION |
|---|---|---|---|---|---|
| .37478282D-02 | .50000000D-02 | .60653078D+00 | .71484827D+00 | .39532754D-05 | .10831784D+00 |
| .37350898D-02 | .87478282D-02 | .41695323D+00 | .49141969D+00 | .46767903D-05 | .74466457D-01 |
| .36982578D-02 | .12482918D-01 | .28699657D+00 | .33825635D+00 | .33065320D-02 | .51259779D-01 |
| .37072651D-02 | .16181176D-01 | .19827582D+00 | .23369112D+00 | .21737896D-05 | .35415303D-01 |
| .37070564D-02 | .19888441D-01 | .13686139D+00 | .16130776D+00 | .15183818D-05 | .24446368D-01 |
| .37075601D-02 | .23595497D-01 | .94475884D-01 | .11135100D+00 | .10472097D-05 | .16875116D-01 |
| .37084491D-02 | .27303057D-01 | .65219706D-01 | .76867865D-01 | .72251920D-05 | .11648159D-01 |
| .37101812D-02 | .31011506D-01 | .45027221D-01 | .53066723D-01 | .49821761D-06 | .80395025D-02 |
| .37071438D-02 | .34721688D-01 | .31091478D-01 | .36639345D-01 | .34590322D-06 | .55478665D-02 |
| .37141789D-02 | .38428832D-01 | .21488472D-01 | .25318081D-01 | .23605399D-06 | .38296089D-02 |
| .37253789D-02 | .42143010D-01 | .14857498D-01 | .17499160D-01 | .16220566D-06 | .26416617D-02 |
| .37459051D-02 | .45868389D-01 | .10281506D-01 | .12101682D-01 | .11075038D-06 | .18201758D-02 |
| .37829702D-02 | .49614294D-01 | .71250399D-02 | .83766169D-02 | .74960976D-07 | .12515770D-02 |
| .38538715D-02 | .53397265D-01 | .49494326D-02 | .58068480D-02 | .49688667D-07 | .85741535D-03 |
| ..39952380D-02 | .57251136D-01 | .34506342D-02 | .40338698D-02 | .31558321D-07 | .58323559D-03 |

| | | | | | |
|---|---|---|---|---|---|
| .43177111D-02 | .61246374D-01 | .24180268D-02 | .28091859D-02 | .17761024D-07 | .39115907D-03 |
| .56111438D-02 | .65564085D-01 | .17028170D-02 | .19568243D-02 | .48568757D-08 | .25400728D-03 |
| .43485534D-02 | .71175229D-01 | .11713402D-02 | .13162377D-02 | .31172560D-07 | .14489748D-03 |
| .42823525D-02 | .75523782D-01 | .95563610D-03 | .10494257D-02 | .10610869D-07 | .93789641D-04 |
| .53466292D-02 | .79806135D-01 | .85031633D-03 | .91143211D-03 | .29185940D-08 | .61115779D-04 |
| .43554410D-02 | .85152764D-01 | .81782584D-03 | .85365089D-03 | .18707107D-07 | .35825051D-04 |
| .67877616D-02 | .12928594D+00 | .21634235D-02 | .21638156D-02 | .20655679D-07 | .39211789D-06 |
| .69158364D-02 | .13607371D+00 | .25207794D-02 | .25209540D-02 | .23525128D-07 | .17461042D-06 |
| .71087955D-02 | .14298954D+00 | .29241822D-02 | .29242427D-02 | .26124609D-07 | .60473119D-07 |
| .73361701D-02 | .15009834D+00 | .33819451D-02 | .33819442D-02 | .29630227D-07 | .88150513D-09 |
| .75829141D-02 | .15743451D+00 | .39022581D-02 | .39022226D-02 | .33924503D-07 | .35441924D-07 |
| .78419504D-02 | .16501742D+00 | .44936163D-02 | .44935594D-02 | .38973139D-07 | .56832608D-07 |
| .81098218D-02 | .17285937D+00 | .51651319D-02 | .51650597D-02 | .44804642D-07 | .72191163D-07 |
| .83830477D-02 | .18096919D+00 | .59267277D-02 | .59266424D-02 | .51535060D-07 | .85273840D-07 |
| .86635058D-02 | .18935224D+00 | .67890927D-02 | .67889948D-02 | .59109163D-07 | .97881517D-07 |
| .89497166D-02 | .19801575D+00 | .77642468D-02 | .77641358D-02 | .67745541D-07 | .11107300D-06 |
| .92414175D-02 | .20696546D+00 | .88653054D-02 | .88651800D-02 | .77528668D-07 | .12539381D-06 |
| .95385040D-02 | .21620688D+00 | .10106681D-01 | .10106540D-01 | .88586865D-07 | .14117777D-06 |
| .98409108D-02 | .22574539D+00 | .11504206D-01 | .11504047D-01 | .10106553D-06 | .15866956D-06 |
| .10151111D-01 | .23558360D+00 | .13075252D-01 | .13075074D-01 | .11498236D-06 | .17794337D-06 |

| | | | | | |
|---|---|---|---|---|---|
| .10465229D-01 | .24573741D+00 | .14839314D-01 | .14839114D-01 | .13086880D-06 | .19949337D-06 |
| .10785050D-01 | .25620275D+00 | .16817109D-01 | .16816886D-01 | .14861924D-06 | .22336000D-06 |
| .11110385D-01 | .26698780D+00 | .19031553D-01 | .19031303D-01 | .16852117D-06 | .24977657D-06 |
| .11441386D-01 | .27809818D+00 | .21507724D-01 | .21507445D-01 | .19081248D-06 | .27899732D-06 |
| .11778086D-01 | .28953957D+00 | .24273017D-01 | .24272706D-01 | .21574812D-06 | .31129198D-06 |
| .12120513D-01 | .30131765D+00 | .27357331D-01 | .27356984D-01 | .24360616D-06 | .34694960D-06 |
| .12474374D-01 | .31343817D+00 | .30793258D-01 | .30792872D-01 | .27406529D-06 | .38567572D-06 |
| .12831962D-01 | .32591254D+00 | .34618099D-01 | .34617670D-01 | .30893224D-06 | .42903403D-06 |
| .13196632D-01 | .33874447D+00 | .38870187D-01 | .38869710D-01 | .34731515D-06 | .47656098D-06 |
| .13569647D-01 | .35194110D+00 | .43592318D-01 | .43591789D-01 | .38983681D-06 | .52860748D-06 |
| .16728465D-01 | .47114946D+00 | .10458661D+00 | .10458542D+00 | .95452878D-06 | .11944316D-05 |
| .17151405D-01 | .48787793D+00 | .11612708D+00 | .11612577D+00 | .10544049D-05 | .13125559D-05 |
| .17583768D-01 | .50502933D+00 | .12881007D+00 | .12880862D+00 | .11700620D-05 | .14439100D-05 |
| .18023101D-01 | .52261310D+00 | .14273842D+00 | .14273683D+00 | .12980771D-05 | .15886880D-05 |
| .18380138D-01 | .54063620D+00 | .15802120D+00 | .15801942D+00 | .14733984D-05 | .17812033D-05 |
| .18773706D-01 | .55901634D+00 | .17469220D+00 | .17469025D+00 | .16168860D-05 | .19487900D-05 |
| .19147490D-01 | .57779004D+00 | .19289020D+00 | .19288805D+00 | .17986446D-05 | .21486463D-05 |
| .19490965D-01 | .59693753D+00 | .21270939D+00 | .21270702D+00 | .20028926D-05 | .23742035D-05 |
| .20190093D-01 | .61642850D+00 | .23423302D+00 | .23423060D+00 | .20213367D-05 | .24224096D-05 |
| .20781260D-01 | .63661859D+00 | .25801084D+00 | .25800815D+00 | .22986772D-05 | .26947522D-05 |

| | | | | | |
|---|---|---|---|---|---|
| .21288654D-01 | .65739985D+00 | .28411150D+00 | .28410848D+00 | .25919316D-05 | .30312814D-05 |
| .22097693D-01 | .67868851D+00 | .31261620D+00 | .31261308D+00 | .26702624D-05 | .31278562D-05 |
| .22649392D-01 | .70078620D+00 | .34415701D+00 | .34415335D+00 | .31313493D-05 | .36583639D-05 |
| .22989474D-01 | .72343559D+00 | .37861657D+00 | .37861235D+00 | .36172503D-05 | .42172649D-05 |
| .21384989D-01 | .74642507D+00 | .41587101D+00 | .41586624D+00 | .41034689D-05 | .47707926D-05 |
| .23577448D-01 | .76961006D+00 | .45583976D+00 | .45583471D+00 | .43151585D-05 | .50491070D-05 |
| .23829860D-01 | .79318750D+00 | .49903107D+00 | .49902542D+00 | .48715650D-05 | .56538239D-05 |
| .24069904D-01 | .81701736D+00 | .54537328D+00 | .54536708D+00 | .53444935D-05 | .62010260D-05 |
| .24239227D-01 | .84108427D+00 | .59500214D+00 | .59499530D+00 | .59125226D-05 | .68367491D-05 |
| .24426484D-01 | .86532349D+00 | .64794104D+00 | .64793362D+00 | .64211820D-05 | .74177801D-05 |
| .26639731D-01 | .88974998D+00 | .70437504D+00 | .70436920D+00 | .47020733D-05 | .58407201D-05 |
| .29850060D-01 | .91638971D+00 | .76955668D+00 | .76955047D+00 | .44861781D-05 | .62129854D-05 |
| .26765177D-01 | .94623977D+00 | .84723442D+00 | .83710691D+00 | .75558956D-05 | .10127507D-01 |
| .10368365D-01 | .97300495D+00 | .92118137D+00 | .87131803D+00 | .83858234D-05 | .49863339D-01 |
| .71228255D-01 | .98337331D+00 | .95094468D+00 | .91591031D+00 | .81602330D-05 | .35034372D-01 |

# TABLE 5

RESULTS OF IMPLICIT RATIONAL RUNGE - KUTTA SCHEMES FOR SOLVING STIFF SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS ADOPTING FEHLBERG (1964) APPROACH FOR ESTIMATING THE LOCAL TRUNCATION ERROR AND CONTROL THE STEP SIZES.

| | | Y1 | Y2 | Y3 |
|---|---|---|---|---|
| X | CONTROL STEP SIZE | E1 | E2 | E3 |
| | | .1980099667D+01 | .9706425830D+00 | .8869204674D+00 |
| .3000000000D-01 | .3000000000D-01 | .8291942688D-09 | .3281419103D-07 | .8161313500D-05 |
| | | .1885147337D+01 | .8379203859D+00 | .4917945068D+00 |
| .1774236000D+00 | .1771470000D-01 | .9577894033D-10 | .3422855333D-08 | .5357828618D-06 |
| | | .1791235536D+01 | .7191953586D+00 | .2663621637D+00 |
| .3307246652D+00 | .1046033532D-01 | .11050933794D-10 | .3558725536D-09 | .3474808041D-07 |
| | | .1694213422D+01 | .6088845946D+00 | .1365392880D+00 |
| .4977858155D+00 | .6176733963D-02 | .1269873096D-11 | .3655098446D-10 | .2146555961D-08 |
| | | .1556933815D+01 | .4729421983D+00 | .4953161076D-01 |
| .7512863895D+00 | .3647299638D-01 | .1425978891D-08 | .3505060447D-07 | .1010194837D-05 |
| | | .1435390902D+01 | .3709037123D+00 | .1867601194D-01 |
| .9951298893D+00 | .2153693963D-01 | .1594313570D-09 | .3316564301D-08 | .4481540687D-07 |

## TABLE 6

RESULTS OF IMPLICIT RATIONAL RUNGE - KUTTA SCHEMES FOR SOLVING STIFF SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS ADOPTING FEHLBERG (1964) APPROACH FOR ESTIMATING THE LOCAL TRUNCATION ERROR AND CONTROL THE STEP SIZES.

| | | Y1 | Y2 | Y3 | Y4 |
|---|---|---|---|---|---|
| X | CONTROL STEP SIZE | E1 | E2 | E3 | E4 |
| | | .9950124792D+00 | .9900498337D+00 | .9139311928D+00 | .9048374306D+00 |
| .3000000000D-01 | .3000000000D-01 | .2597677629D-10 | .4145971344D-09 | .2617874150D-05 | .3971726602D-05 |
| | | .9708623323D+00 | .9425736684D+00 | .5872698932D+00 | .5535451450D+00 |
| .1774236000D+00 | .1771470000D-01 | .3078315380D-11 | .4788947017D-10 | .2005591107D-06 | .2890213078D-06 |
| | | .9402798026D+00 | .8841261072D+00 | .3300866691D+00 | .2918382654D+00 |
| .3694667141D+00 | .1046035320D-01 | .3621547506D-12 | .5454525720D-11 | .1355160001D-07 | .1829417523D-07 |
| | | .9144602205D+00 | .8362374949D+00 | .1999708940D+00 | .1672231757D+00 |
| .5365278644D+00 | .6176733963D-02 | .4285460875D-13 | .6268319197D-12 | .9915873955D-09 | .1265158728D-08 |
| | | .8693495443D+00 | .7557686301D+00 | .8044517344D-01 | .6079796167D-01 |
| .8400599835D+00 | 3647299638D-01 | .4961209221D-10 | .6922001861D-09 | .5087490103D-06 | .5899525189D-06 |

## DISCUSSION

In all the cases, the computed errors show that the schemes are very accurate, stable and convergent.

As can be seen from the results in tables 1-3 these schemes compare favourably with the existing Runge - Kutta Schemes of the same order.

# CHAPTER SIX

## GENERAL CONCLUSION

### 6.1 SUMMARY

In the project, we have developed families of implicit rational Runge - Kutta schemes for solving stiff Ordinary differential equations.

It is motivated by the Rational function approximation scheme proposed by Hong Yuan Fu (1982).

It is analysed, computerised and implemented on sample problems. The results shows that the new schemes are capable of solving both stiff and non-stiff problems.

The advantages of these new schemes over the R-K scheme of the same order is that it has large region of absolute stability and improved accuracy.

### 6.2 LIMITATIONS

Based on the above work, the starting step size may affect the implementation because efficiency results were observed to depend on the starting size h.

However, if the step size ($h_{old}$) is too large, the program will not respond or fail to terminate.

Also, introduction of too small step size may result into round off error and consuming computer time because of the implicit nature of the schemes.

Another limitation is that the methods involve large parameters which lead to requirement of large computer facilities.

This also affects the stability of the schemes since as the order of the schemes increases the region of absolute stability becomes smaller.

Adopting these method to solve systems of ordinary differential equations, a large amount of algebra was involved and it is time consuming.

Another very serious limitation of the schemes is non - acceptability of zero initial conditions. At zero value of y(x), the program signifies floating error.

## 6.3 RECOMMENDATIONS

Based on the above limitations we need to adopt some strategies which can help to find appropriate balance between the step size (h), the order P of accuracy of the methods in order to achieve a better, more accurate, efficient, and effective scheme.

These strategies include choice of initial step size (h), the parameter of the schemes and introduction of a flag which will signify when a zero value is being approached.

Implementation techniques which accommodate appropriate step size control can lead to significant improvement in convergence rate of the iteration.

For example it is advantageous to choose the initial step size to be of order of the smallest time constant or largest eigen value of the differential systems in order to minimize the amount of computations and storage requirement since too large initial step size may result into the schemes failing to start or terminate.

Another strategy is to allow more iterations per step in order to reduce the effect of error due to small step size (h) and error tolerance (tol) and use double precision arithmetic to minimize the effects of round off error.

It is also necessary that some strategies be adopted to make the schemes accept zero initial conditions. However, we intend to take up some of these challenges in future studies.

## 6.4 AREAS OF FURTHER STUDIES

In this thesis, we only consider the implicit aspect of the rational R - K schemes. The previous works have considered the explicit aspect of the rational R - K schemes.

Area of the schemes that need further studies is the semi - implicit aspect of the methods which could leads to viable numerical methods for solution of ODES.

Also, the previous and the present work have assumed that r and s are equal, (numbers of functions evaluations). Therefore further studies can consider areas of the schemes with r not equals to s.

## 6.5 CONTRIBUTION TO KNOWLEDGE

The new implicit rational R - K schemes for solving stiff and non - stiff ordinary differential equation is suitable enough to be developed into multipurpose code for solving problem arising from nuclear reaction processes, economic system, radioactive dynamic processes, and other dynamic process such as diffussion, heat and matter transfer and pharmaco kinectic theories.

The developed schemes are new creation in research efforts to find suitable techniques for solving these classes of problems.

By the new development, a new dimension to research effort in the areas of scientific computing in applied Mathematics and Engineering technology have been extended.

# REFERENCES


1. Addison, C.A. (1979), "Implementing a Stiff Method Based upon the Second Derivative Formular", Technical Report No. 130, Department of Computer Science, University of Toronto, Canada. Pg 60-68.

2. Ademuluyi, R. A. (1987), "New hybrid methods for systems of stiff Ordinary differential equations". PhD Dissertation, University of Benin, Benin City.

3. Ademiluyi, R.A. (1988), "The Design of a Class of Stiff ODEs Solvers". Computational Method II, Proceedings of first International conference on Numerical Analysis and its application held at the University of Benin, Benin City.

4. Alexander, R. (1977), "Diagonally implicit Runge-Kutta methods for stiff Ordinary differential equations". SIAM Journal of Numerical Analysis Vol. 14, Pg. 1006 - 1021.

5. Birkoff, G and Varga, R.S. (1965), "Discretization Errors for Well-set Cauchy Problems"; Int. Journals of Mathematics and Physics, Vol.44 Pg 1 - 23.

6. Blum, E.K. (1952), "A modification of Runge - Kutta Fourth - order Method", Maths Computation, Vol. 16 Pg 176 - 187.

7. Brice, C. (1969), "Applied Numerical Method". John Wiley and Son Inc. (New York). Pg 70-78

8. Burrage, K. And Butcher, J.C. (1973), "Stability Criteria for Implicit Runge - Kutta Methods". SIAM Journal on Numerical Analysis,

Vol.16, Pg. 46 - 57.

9. Burrage, K. (1978) "A Special Family of Runge - Kutta method for solving stiff Differential equations" B IT 18, Pg. 22 - 41.

10. Burdent and Faires, (1985) Initial Numberical analysis approach. Third Edition, Weber and Schmidt. Pg 100-110

11. Butcher, J.C. (1864), "Implicit Runge - Kutta processes" Math computation Vol. 18, Pg. 50 - 64.

12. Butcher, J.C. (1975), "A Stability property of Implicit Runge - Kutta methods", B.T. 16, Pg. 237 - 240.

13. Butcher, J.C., Burrage, K. And Chipman, F.H. (1979), "STRIDE stable Runge-Kutta Integrator for Differential equations." Computational Mathematics Report. No. 20.

14. Byrne, G.D. and Hindmarch, A.C. (1975), "A Polyalgorithm for Numerical Solution of ODE", ACM TOM 51 pg 71-96.

15. Cash, J.R. and A. Singhal, (1982) "Meno - Implicit Runge - Kutta formular for numerical integrator of stiff Differential equations" IMA Journal of Numerical Analysis; Vol 12 Pg. 211 - 226.

16. Chipman, F.H. (1973), "Implementation of Runge - Kutta processes", BIT 13, Pg. 391 - 393.

17. Copper, G.J. (1980), "On existence of algebraically stable Runge - Kutta method" IMA Journal on Numerical Analysis, Vol. 16, Pg. 325 - 331.

18. Curtis, C.F. and J. O. Hir schfelder (1952) "Integration of stiff equations". National academy of sciences, Vol. 38, pg. 235 - 243.

19. Dalquist, D. (1863), "A special stability problem for linear multistep methods" BIT 3 pg. 27 - 43.

20. Dormand, J.R. and P. Prince, (1985) "Global error estimation with Runge - Kutta methods" IMA Journal of Numerical Analysis Vol.5, Pg. 481 - 497.

21. Enright, W.H. (1974), "Second Derivative Multistep Method for Stiff ODEs" SIAM J Num Analysis 11 pg 321-331.

22. Enright, W.H. (1975), "Comparing numerical methods for stiff systems of Ordinary differential equations". BIT Vol.15, Pg. 19 - 48.

23. Fatunla S.O. (1980). "Numerical Integrator for stiff and highly oscillatory problem Differential equation". Math Computation Vol. 34, pg. 374 - 390.

24. Fatunla, S.O. (1986), "Recent advances in stiff Ordinary differential equations preceeding of the first International Conference on Numerical Analysis and Application, January 1986 Pg. 27-31.

25. Fatunla, S.O. (1986). Applied Numerical Methods for initial value problem in Ordinary differential equations. Academic press Cambridge, Manuscript. Pg 370-378.

26. Felhberg, E. (1964), "New higher order Runge - Kutta, Formulas with step size control for systems of first, and second order Differential equations", Angew Maths Mech. Vol. 44, pg. 83 - 88.

27. Gafney, P.W. (1984), "A Performance Evaluation of some FORTRAN Subroutines for Integration of stiff Oscillatory ODEs", ACM TOM 10, pg 58-72.

28. Gear, C.W. (1965), "Hybrid methods for initial value problems" in Ordinary differential equations". SIAM Journal on Numerical Analysis Vol. 2, Pg. 69 - 86.

29. Gear, C.W. (1971), "DIFSUB for solution of Ordinary differential equations" Communication of ACM, Vol. 14, Pg. 185 - 190.

30. Gill, S. (1951), "A Process for Step - by - Step Integration of Differential Equations in an Automatic Digital Computing Machine", Proc. Cambridge Philos. Soc. Vol.47 Pg 95 - 108.

31. Hammer, P.C. and Hollingsworth, J.W. (1955), "Trapezoidal Methods of Approximating Solutions of Differential Eqoations", M.T.A.C, Vol.9 Pg 92 - 96.

32. Hindmarch, A.C. (1974), "GEAR: ODE System Solver Revision 3 Report UCID - 30001, Lawrence Livermore Laboratory, University of California, Livermore.

33. Hong Yuan Fu (1980), "A Class of A - stable or A (∝) stable Explicit Schemes" Computational and assymptatic methods for Boundary and interior layer, Proceeding of BAILJI Conferences Trinity College, Dublin, Pg. 236 - 241.

34. Jain, M.K. (1979), Numerical Solution of Ordinary differential equations, Wiley Eastern Limited. Pg 200-210

35. Jain, M.K. (1982), "Numerical method for scientific and Engineering Computation Wiley Eastern Limited. Pg 250-260

36. Kaps P. And Rentrop. P. (1979), "Generalised Runger - Kutta method of order four with steps size control for stiff Ordinary differential

equation". Computational Mathematics Vol. 33, Pg. 55 - 56.

37. King, R (1966), "Runge - Kutta Methods with Constrained Minimum Error Bounds", Maths. Comp. Vol. 20, Pg 386 - 391.

38. Lambert, J.D., and Watson, J. A. (1976) "Symmetric multistep methods for periodic initial value problems". IMA Journal of Numerical Analysis Pg. (189 - 202).

39. Lawson, J.D. (1966), "An Order Five R-K Process with Extended Region of Absolute Stability". SINUM 4, pg 372-380.

40. Liniger,W and Willoughby, R. A. (1967), "Efficient Numerical Integration Methods for Stiff Systems of Differential Equations", IBM Research Report RC - 1970.

41. Norsett, S.P. (1974), "One Step Methods of Harmite Type for Numerical Integration of Stiff Systems BIT 14, pg 63-77.

42. Novikov, V.A. and Novikov E.A. (1986) "Explicit methods for stiff systems of Ordinary differential equations". Preceeding of first international conference on Numerical Analysis and its application Pg. (27 - 31).

43. Okunbor D.I. (1985), Explicit "Runge - Kutta Schemes for stiff systems of Ordinary differential Equations". M.Sc Thesis, University of Benin, Benin city.

44. Peter, H. (1962), Discrete variable methods in Ordinary differential equations, John Wiley and Inc. London. Pg 400-420

45. Press, W.H., Flanner, B.P., Teukolsky, S.A. and Veterling, W.T. (1986). Numerical Recipe, Scientific Computing, Cambridge

University Press, Cambridge.

46. Ralston, A. (1962), "Runge - Kutta with Minimum error bounds", Maths Computation Vol. 16, Pg. 431 - 438.

47. Ralston, A. (1965), A first course in Numerical Analysis, Mc Graw - Hill New York. Pg 300-320

48. Robert, D. (1983), "Computational error estimates for stiff Ordinary differential equations", Proceeding of first International Conference on Numerical Analysis and its applications.

49. Sidney, Y. and Ferenc, (1986). And Introduction to Numerical Computations, Macmillan Publishing Company (New York and London). Pg 250-260

50. Skeel, R.D. and Kong, A.D. (1972), "Blended Linear Multistep Methods". ACM Tom 3, Pg. 326 - 345.

51. Steeve C. and Raymond, P.C. (1988), Numerical Methods for Engineers, McGraw - Hill International Edition. Pg 400-450

## APPENDIX 1

```
C THIS PROGRAM SOLVES STIFF INITIAL VALUE PROBLEM Y'=λ (Y-E(X))+E'(X)
C WHERE λ IS A COMPLEX NUMBER WITH NEGATIVE REAL PART
C  COMPARING TWO-STAGE  IMPLICIT RATIONAL RUNGE-KUTTA OF ORDER
C FOUR AND CLASSICAL RUNGE KUTTA OF ORDER FOUR
    IMPLICIT REAL*8(A-H,T, X-Z)
C INITIALISE X,Y,H,TOL,XL
    OPEN(UNIT=3,FILE='PAT.OUT',STATUS='NEW')
    WRITE(3,1)
 1   FORMAT(12X,' RESULTS OF IMPLICIT RATIONAL RUNGE-KUTTA SCHEME

    1 OF ORDERS FOUR AND CLASSICAL RUNGE-KUTTA SCHEME OF ORDER
FOUR')
    WRITE(3,102)
102  FORMAT('|','-----------------------------------------------------
    1----------------------------------------------------------------
    1---','|')
    WRITE(3,7)
 7   FORMAT(6X,'H',24X,'YEXACT',13X,'ORDER FOUR',13X,'E2',13X,'CLASSICAL
    1RUNGE KUTTA OFORDER FOUR',  3X,'E3')
    WRITE(3,101)
101  FORMAT('|','.....................................................
    1...............................................................
    1...','|')
    HI=0.1D-1
8    HMIN = 0.1953125D-5
    TOL= 1.0D-6
    XL=1.0D0
    X0=0.0D0
    Y0=1.0D0
C INVOKE SUBROUTINE IMPRRK4AND IRUNKUT TO COMPUTE
C APPROXIMATE VALUE OF Y
    CALL IMPRRK4(X0,Y0,HI,TOL,YP,E2,YN4)
    CALL IRUNKUT(X0,Y0,HI,YP,E3,YP4)
C  CHANGE THE STEPSIZE TO HALF OF THE STEPSIZE
    HN=0.5D0*HI
C WRITE OUT RESULTS
    WRITE(3,10)HI,YP, Y,E2, YP4, E3
10   FORMAT(D14.8,7X,5(D16.8,4X))
    WRITE(3,100)
100  FORMAT('|',18('-'),'|',5(20('-'),'|'))
    IF(HI.LE.HMIN) THEN
    STOP
    ELSE
    HI=HN
    GO TO 8
```

```
      END IF
      END

C FUNCTION SUBPROGRAM TO COMPUTE THE FUNCTION OF F
      FUNCTION F(X,Y)
      IMPLICIT REAL*8(A-H,T,X-Z)
      F=-100D0*(Y-X**3D0)+3D0*X*X
      RETURN
      END

C FUNCTION SUBPROGRAM TO COMPUTE YEXACT
      FUNCTION YEXACT(X,Y)
      IMPLICIT REAL*8(A-H,T,X-Z)
      YEXACT=(X**3.0D0)+EXP(-100.0*X)
      RETURN
      END

C  SUBROUTINE  IMPRRK4 TO COMPUTE THE APPROXIMATE VALUE OF
C Y,USING STAGE TWO  IMPLICIT RATIONAL RUNGE KUTTA OF ORDER FOUR
      SUBROUTINE IMPRRK4(X, Y, H,TOL,YP,E,YN)
      IMPLICIT REAL*8(A-H,T,X-Z)
      AH1=0.0D0
      AH2=0.0D0
      B11=0.25D0
      B12=0.25D0+(SQRT(3.0D0)/6.0D0)
      B22=0.25D0
      B21=0.25D0-(SQRT(3.0D0)/6.0D0)
      D1=0.5D0 +(SQRT(3.0D0)/6.0D0)
      D2=0.5D0 -(SQRT(3.0D0)/6.0D0)
12    X1=X+D1*H
      X2=X+D2*H
      Z0= 1.0D0/Y
      Z1=Z0+(B11*AH1)+(B12*AH2)
      Z2=Z0+(B21*AH1)+(B22*AH2)
      Y1=1.0D0/Z1
      Y2=1.0D0/Z2
      GH1=-Z1*Z1*F(X1,Y1)*H
      GH2=-Z2*Z2*F(X2,Y2)*H
      A1=ABS(GH1-AH1)
      A2=ABS(GH2-AH2)
      IF(A1 .LE.TOL .AND. A2.LE.TOL) THEN
      TSUM=0.5D0*(GH1+GH2)
      YN=Y/(1.0D0+(Y*TSUM))
      XN=X+H
      YP=YEXACT(XN,YN)
      E=ABS(YP-YN)
      ELSE
```

```
      AH1=GH1
      AH2=GH2
      GO TO 12
      END IF
      RETURN
      END


C SUBROUTINE IRUNKUT COMPUTE APROXIMATE VALUES OF Y USING
C CLASSICAL RUNGE KUTTA  SCHEME OF ORDER FOUR
      SUBROUTINE IRUNKUT4(X,Y,H,YP,E,YN)
      IMPLICIT REAL*8(A-H,T,X-Z)
      AK1=H*F(X,Y)
      X2=X+0.50D0*H
      Y2=Y+(0.5D0*AK1)
      AK2=H*F(X2,Y2)
      Y3=Y+(0.5D0*AK2)
      AK3=H*F(X2,Y3)
      X3=X+H
      Y4=Y+AK3
      AK4=H*F(X3,Y4)
      TSUM=(AK1+(2,0D0*(AK2+AK3))+AK4)/6.0D0
      YN=Y+TSUM
      XN=X+H
      YP=YEXACT(XN,YN)
      E=ABS(YP-YN)
      RETURN
      END
```

## APPENDIX 2

```
C  THIS PROGRAM SOLVES STIFF SYSTEMS OF ORDINARY DIFFERENTIAL
C   EQUATIONS   USING TWO-STAGE IMPLICIT RATIONAL RUNGE KUTTA OF
C  ORDER FOUR.
C  USING ERROR ESTIMATES TO ADJUST THE STEPSIZE. THE STRATEGY IS
C  TO CONTROL   THE STEPSIZE TO GET MINIMUM ERROR.
C  THE CRITERION OF CONTROLLING THE STEPSIZE WAS SUGGESTED BY
C  PRESS ET AL  (1984)
      IMPLICIT REAL*8(A-H,T-Z)
      OPEN(UNIT=8,FILE='PAT3.OUT',STATUS='NEW')
      WRITE(8,*)'RESULTS OF TWO STAGE IMPLICIT RATIONAL RUNGE KUTTA'
      WRITE(8,*)'OF ORDER FOUR,ADOPTING RICHARDSON EXTRAPOLATION '
      WRITE(8,*)'METHOD TO ESTIMATE THE LOCAL TRUNCATION ERROR AND'
      WRITE(8,*)' CONTROL THE STEP SIZE'
      WRITE(8,120)
 120  FORMAT('|','-----------------------------------------------------
     1--------------------------------------------------------------
     1---','|')
      WRITE(8,11)
 11   FORMAT(3X,'STEP SIZE          VALUE OF X        YEXACT
     1 NUMERICAL VALUE      LOCAL ERROR        ERROR')
      WRITE(8,130)
 130  FORMAT('|','------------------------------------------------------
     1--------------------------------------------------------------
     1','|')
      X0=0.0D0
      Y0=1.0D0
      HI=0.5D-1
      TOL=0.1D-3
      XL=1.0D0
 2    XP=X0+HI
      CALL ADAPT(X0,Y0,TOL,HN,LTE)
      CALL IMPRRK4(X0,Y0,HI,TOL,YNP1)
      YN=YEXACT(XP,YNP1)
      E=ABS(YN-YNP1)
      WRITE(8,3) HN,XP,YN,YNP1,LTE,E
 3    FORMAT(3X,6(D16.8,4X))
      WRITE(8,100)
 100  FORMAT('|',6(20('-'),'|'))
      IF(XP .GE. XL) STOP
      X0=XP
      Y0=YNP1
      GO TO 2
      END
C  SUBROUTINE ADAPT TO CONTROL THE STEPSIZE
```

```
      SUBROUTINE ADAPT(X,Y,TOL,HN,LTE)
      IMPLICIT REAL*8(A-H,T-Z)
      H=0.5D-1
7     YPS=ABS(Y+(H*F(X,Y)))
      DN=TOL*YPS
      H2=0.5D0*H
      XP1=X+H2
      CALL IMPRRK4(X,Y,H,T,YP1)
      CALL IMPRRK4(X,Y,H2,T,YP2)
      CALL IMPRRK4(XP1,YP2,H2,T,YP3)
      DP=ABS(YP3-YP1)
      DNP=ABS(DN/DP)
      EA=(32.0D0*DP)/31.0D0
      IF(DP .LE. DN) THEN
      HN=H*ABS(DNP**0.2D0)
      ELSE
      HN=H*ABS(DNP**0.25D0)
      END IF
      IF(LTE .LE. ET)THEN
      GOTO 8
      ELSE
      H=HN
      GOTO 7
      ENDIF
8     RETURN
      END


C  SUBROUTINE IMPRRRK4 TO COMPUTE NUMERICAL VALUES OF Y
      SUBROUTINE IMPRRK4(X,Y,H,T,YP)
      IMPLICIT REAL*8(A-H,T-Z)
C  INITIALIZE THE VARIABLES
      AH1=0.0D0
      AH2=0.0D0
      B11=0.25D0
      B12= 0.25D0+SQRT(3.0D0)/6.0D0
      B21= 0.25D0-SQRT(3.0D0)/6.0D0
      B22=0.25D0
      D1= 0.5D0+(SQRT(3.0D0)/6.0D0)
      D2= 0.5D0-(SQRT(3.0D0)/6.0D0)
5     X1=X+(D1*H)
      X2=X+(D2*H)
      Z0=1.0D0/Y
      Z1=Z0+(B11*AH1)+(B12*AH2)
      Z2=Z0+(B21*AH1)+(B22*AH2)
      Y1=1.0D0/Z1
```

```
      Y2=1.0D0/Z2
      GH1=-Z1*Z1*F(X1,Y1)*H
      GH2=-Z2*Z2*F(X2,Y2)*H
      IF(ABS(GH1-AH1) .LE. T .AND.ABS(GH2-AH2).LE. T)THEN
      TSUM2=0.5D0*(GH1+GH2)
      YP=Y/(1.0D0+(Y*TSUM2))
      ELSE
      AH1=GH1
      AH2=GH2
      GO TO 5
      END IF
      RETURN
      END

C  FUNCTION SUBPROGRAM TO COMPUTE F
      FUNCTION F(X,Y)
      IMPLICIT REAL*8(A-H,T-Z)
      F=-10.0 *(Y-X**3) +3*X**2
      RETURN
      END

C  FUNCTION SUBPROGRAM TO COMPUTE YEXACT
      FUNCTION YEXACT(X,Y)
      IMPLICIT REAL*8(A-H,T-Z)
      YEXACT=X**3+EXP(-10.0*X)
      RETURN
      END
```

## APPENDIX 3

```
C THIS PROGRAM SOLVES STIFF SYSTEMS OF ORDINARY DIFFERENTIAL
C EQUATIONS. USING TWO STAGE IMPLICIT RUNGE KUTTA OF ORDER
C THREE AND ORDER FOUR
C THE ERROR IS TAKEN AS A RESULT OF DIFFERENCE BETWEEN RESULTS
C  OF ORDER FOUR AND ORDER THREE.
      IMPLICIT REAL*8(A-H,T-Z)
      CHARACTER AKL*1
      DIMENSION YN(10),A(10,10),E(10),YNEW(10)
      OPEN(UNIT=5,FILE='PAT8.OUT',STATUS='NEW')
      WRITE(5,1)
      WRITE(5,2)
      WRITE(5,15)
      WRITE(5,17)
      WRITE(5,3)
      WRITE(5,16)
1     FORMAT('RESULTS OF IMPLICIT RUNGE-KUTTA SCHEMES FOR SOLVING
STIFF
     1 SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS ADOPTING FEHLBERG
     1(1964) APPROACH FOR ESTIMATING THE LOCAL TRUNCATION ERROR AND
     1CONTROL THE STEP SIZES')
2     FORMAT('------------------------------------------------------------
     1-----------------------------------------------------------------')

15    FORMAT('                          Y1           Y2
     1      Y3  ')
17    FORMAT('---------------------------------------------------------
     1 -----------------------------------------------------')
3     FORMAT('      X    CONTROL STEP SIZE    E1              E2
     1           E3    ')
16    FORMAT('|','-------------------------------------------------------
     1-----------------------------------------------------------------
     1-------------','|')
      WRITE(*,*) 'NUMBER OF SYSTEM OF EQUATIONS'
      READ(*,*) N
      WRITE(*,*) 'INITIAL VALUE OF X '
      READ(*,*) XN
      WRITE(*,*) 'LAST VALUE OF X'
      READ(*,*) XL
      WRITE(*,*) 'STEP SIZE '
      READ(*,*) HI
      WRITE(*,*) 'ERROR TOLERANCE'
      READ(*,*) TOL
      WRITE(*,*)' ENTER DATA FOR A()'
      DO 20 I=1,N
      WRITE(*,*) I
```

```
      DO 20 J=1,N
      WRITE(*,*) J
20    READ(*,*) A(I,J)
      WRITE(*,*)'ENTER VALUE FOR Y()'
      DO 30 I=1,N
      WRITE(*,*) I
30    READ(*,*) YN(I)
C      WRITE(*,*)'ENTER VALUE FOR MIN STEPSIZE'
C      READ(*,*) HMIN
       KI=0
4      CALL IMPRK43(N,A,XN,YN,HI,TOL,HN,E,YNEW)
       XP=XN+HN
      KK=MOD(KI,10)
      WRITE(*,*)' THE VALUE OF KK IS ',KK
      WRITE(*,*)' PRESS ANY KEY TO CONTINUE'
      READ(*,33) AKL
33      FORMAT(A1)
      IF(KK.EQ.0)THEN
      WRITE(5,130)( YNEW(I),I=1,N)
130   FORMAT(39X,2(2X,D16.10,6X))
      WRITE(5,150)
150   FORMAT('|',39('-'),'|',2(22('-'),'|'))
      WRITE(5,6)XP,HN, (E(J),J=1,N)
6     FORMAT(1X,D16.10,3X,D16.10,3X,2(2X,D16.10,6X))
      WRITE(5,160)
160   FORMAT('|'19('-'),'|'19('-'),'|',2(22('-'),'|'))
      ELSE
      GO TO 11
      ENDIF
11     IF (XP.GE. XL) THEN
       STOP
       ELSE
       HI=HN
       XN=XP
       DO 7  L=1,N
7      YN(L)=YNEW(L)
       KI=KI+1
       GO TO 4
       ENDIF
       END
C THIS SUBROUTINE  ADOPT FELHBERG(1964)APPROACH TO ESTIMATE
C  LOCAL TRUNCATION ERROR AND ALSO CONTROL STEP SIZES
      SUBROUTINE IMPRK43(N,A,X,Y,H,TOL,HN,E,YNEW)
      IMPLICIT REAL*8(A-H,T-Z)
      DIMENSION A(10,10),Y(10),YPN3(10),E(10),YNEW(10)
88    CALL IMPRRK3(N,A,X,Y,H,TOL,YPN3)
      CALL RUNKUT4(N,A,X,Y,H,TOL,YNEW)
```

```
      H1=0.1D0*H
      DO 5 K=1,N
      E(K)=ABS(YNEW(K)-YPN3(K))
      IF(E(K).LT.TOL)THEN
      HN=3.0D0*H
      ELSE
      H=H1
      GO TO 88
      END IF
5     CONTINUE
      RETURN
      END

C SUBROUTINE IMPRRK3 COMPUTE APPROXIMATE VALUES Y ,ADOPTING C
TWO STAGE IMPLICIT RUNGE KUTTA METHOD OF ORDER THREE
      SUBROUTINE IMPRRK3(N,A,X,Y,H,TOL,YNP1)
      IMPLICIT REAL*8(A-H,T-Z)
      DIMENSION AH1(10),AH2(10),ZN(10),Z1(10),Y1(10),Y2(10),Z2(10)
      DIMENSION F1(10),F2(10),GH1(10),GH2(10),TSUM1(10),YNP1(10),Y(10)
      DIMENSION A(10,10)
C INITALISE THE VARIABLES
      WRITE(*,60)
60    FORMAT(14X,'RESULTS OF TWO STAGE OF ORDER THREE')
      D1=2.0D0/3.0D0
      B11=1.0D0/2.0D0
      B12=1.0D0/6.0D0
      B21=-1.0D0/2.0D0
      B22=1.0D0/2.0D0
      DO 40 I=1,N
      AH1(I)=0.0D0
40    AH2(I)=0.0D0
21    X1=X+(D1*H)
      X2=X
      DO 22 K=1,N
      ZN(K)=1.0D0/Y(K)
      Z1(K)=ZN(K)+(B11*AH1(K))+(B12*AH2(K))
      Y1(K)=1.0D0/Z1(K)
      Z2(K)=ZN(K)+(B21*AH1(K))+(B22*AH2(K))
      Y2(K)=1.0D0/Z2(K)
22    CONTINUE
      CALL FCN(N,A,X1,Y1,F1)
      CALL FCN(N,A,X2,Y2,F2)
      DO 24 J=1,N
      GH1(J)=-Z1(J)*Z1(J)*F1(J)*H
      GH2(J)=-Z2(J)*Z2(J)*F2(J)*H
      IF((ABS(GH1(J)-AH1(J)).LE.T) .AND.(ABS(GH2(J)-AH2(J)).LE.T)) THEN
      TSUM1(J)=0.25D0*((3.0D0*GH1(J))+GH2(J))
```

```
      YNP1(J)=Y(J)/(1.0D0+(Y(J)*TSUM1(J)))
      ELSE
      AH1(J)=GH1(J)
      AH2(J)=GH2(J)
      WRITE(*,*)'I AM COMPUTING PLEASE...'
      GOTO 21
      ENDIF
24    CONTINUE
      write(*,*)'the value of Y is',(Y(I),I=1,N)
      write(*,*)'the value of gh1 is',(GH1(I),I=1,N)
      WRITE(*,*)'THE VALUE OF GH2 IS',(GH2(I),I=1,N)
       WRITE(*,29)(YNP1(L),L=1,N)
29    FORMAT(3X,3(D16.10,3X))
      RETURN
      END


C SUBROUTINE IRUNKUT4 COMPUTE APPROXIMATE VALUES OF VECTOR Y
C ADOPTING IMPLICIT RATIONAL RUNGE KUTTA METHOD OF ORDER FOUR
     SUBROUTINE RUNKUT4(N,A,X,Y,H,T0L,YNP2)
      IMPLICIT REAL*8(A-H,T-Z)
      DIMENSION AH3(10),AH4(10),ZN(10),Z3(10),Z4(10),Y3(10),Y4(10)
      DIMENSION GH3(10),GH4(10),TSUM2(10),F3(10),F4(10),YNP2(10),Y(10)
      DIMENSION A(10,10)
C  INITIALISE THE VARIABLES
      WRITE(*,44)
44    FORMAT(13X,'RESULTS OF TWO STAGE FOURTH ORDER IRRK METHOD')
     D3=0.5D0+(SQRT(3.0D0)/6.0D0)
      D4=0.5-(SQRT(3.0D0)/6.0D0)
      B11=0.25D0
      B12=0.25D0+(SQRT(3.0D0)/6.0D0)
      B21=0.25D0-(SQRT(3.0D0)/6.0D0)
      B22=0.25D0
      DO 8 J=1,N
      AH3(J)=0.0D0
8     AH4(J)=0.0D0
9     X3=X+(D3*H)
      X4=X+(D4*H)
      DO 11 L=1,N
      ZN(L)=1.0D0/Y(L)
      Z3(L)=ZN(L)+(B11*AH3(L))+(B12*AH4(L))
      Y3(L)=1.0D0/Z3(L)
      Z4(L)=ZN(L)+(B21*AH3(L))+(B22*AH4(L))
      Y4(L)=1.0D0/Z4(L)
11    CONTINUE
      CALL FCN(N,A,X3,Y3,F3)
      CALL FCN(N,A,X4,Y4,F4)
```

```
      DO 12  I=1,N
      GH3(I)=-Z3(I)*Z3(I)*F3(I)*H
      GH4(I)=-Z4(I)*Z4(I)*F4(I)*H
      IF ((ABS(GH3(I)-AH3(I)).LE.T).AND.(ABS(GH4(I)-AH4(I)).LE.T)) THEN
      TSUM2(I)=0.5D0*(GH3(I)+GH4(I))
      YNP2(I)=Y(I)/(1.0D0+Y(I)*TSUM2(I))
      ELSE
      AH3(I)=GH3(I)
      AH4(I)=GH4(I)
      WRITE(*,*)'I AM COMPUTING PLEASE HERE.....'
      GOTO 9
      ENDIF
12    CONTINUE
      write(*,*)'the value of Y is',(Y(J), J=1,N)
      write(*,*)'the value of GH3 is',(GH3(J), J=1,N)
      WRITE(*,*)'THE VALUE OF GH4 IS',(GH4(J), J=1,N)
      WRITE(*,17)(YNP2(L), L=1,N)
17    FORMAT(3X,3(D16.10,3X))
      RETURN
      END

C EXTERNAL NAME SUBROUTINE COMPUTES THE SYSTEMS OF ORDINARY
C DIFFERENTIAL EQUATIONS. THIS CAN SOLVE UP TO TEN SYSTEMS OF
C ORDINARY DIFFERENTIAL EQUATIONS
      SUBROUTINE FCN(N,A,X,Y,F)
      IMPLICIT REAL*8(A-H,T-Z)
      DIMENSION X(10),Y(10),F(10),A(10,10)
       DO 14 J=1,N
       F(J)=0
       DO 14 I=1,N
14      F(J)=F(J)+A(I,J)*Y(J)
       RETURN
       END
```