K^{TH}–ORDER INVERSE POLYNOMIAL METHODS FOR THE INTEGRATION OF ORDINARY DIFFERENTIAL EQUATIONS WITH SINGULARITIES

BY

OKOSUN, KAZEEM OARE

IMC/89/2100

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

SEPTEMBER, 2002

CERTIFICATION

This is to certify that this project was carried out by Mr. OKOSUN KAZEEM OARE of the Department of Industrial Mathematics, Federal University of Technology, Akure.

17-03-03

Dr. R.A. Ademiluyi
Major Supervisor

Date

Prof. O.C. Akinyokun
Chief Examiner

Date

DEDICATION

This work is dedicated to the Almighty God, my late father Mr Ahmed Okosun , my mother Mrs. Philomina Okosun and my wife Mrs Olabimpe Okosun

ACKNOWLEDGEMENT

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

I must express my ineffable gratitude to my supervisor DR. R.A. ADEMILUYI who gave freely his books and ideas. He equally sacrificed his precious time to read through the project and made valuable corrections and suggestions. I also thank him for his patience and tolerance towards my shortcomings.

My special thanks goes to my past head of Department Dr. D.O Awoyemi whose fatherly guidance and instructions have contributed in no small measure to the results of this research work.

My appreciation goes to Prof. O.C. Akinyokun, the head of Department and other lecturers in the department for their support during the course of my programme.

I must not fail to mention the immense contribution of my dear brothers, Mr. B.K. Alese and Mr. B.J. Abiodun who prayerfully stood by me from the beginning to the end of this programme.

To my wife and friends who contributed to the success of this programme, I say thank you.

Finally, I thank the Almighty God who through his mercies made my dream come true.

TABLE OF CONTENTS

PAGES

CHAPTER THREE

CHAPTER FOUR

Properties of the New Schemes

CHAPTER FIVE

Computer Implementation and Numerical Results

CHAPTER SIX

General Conclusion

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

In this thesis, Inverse Polynomial Schemes are developed, analysed and computerized to solve ordinary differential equations with singularities.

The method is motivated by the variety of application areas of this class of ordinary differential equations. In its development and analysis, Taylor'sseries expansion, Binomial series expansion and Pade's approximation technique are used respectively. These schemes are convergent and A-Stable. Numerical results show that the schemes are effective and efficient.

CHAPTER ONE

INTRODUCTION

1.1. ORDINARY DIFFERENTIAL EQUATIONS

Numerical problems are encountered in the various branches of human activities such as science, engineering, management and technology. They include:

(i) The problem of determining the quantity of electric charge flowing across an electric circuit.

(ii) The study of chemicals reactions of two different elements.

(iii) The study of the rate of decomposition of a radioactive substance.

(iv) The rate of growth of the population of a given community.

(v) Impulse response problems

The mathematical formulation of these problems oftens lead to differential equations in the general form :

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

where y is the dependent variable and n is the highest order of its derivatives. This highest order derivative is also the order of the equation and its degree is the power to which the highest derivative is raised after rationalisation.

If no product of the dependent variable y(x) with itself or any of the derivatives occurs, the equation is said to be linear, otherwise, it is non-linear.

The differential equation (1.1) together with initial conditions stated in (1.2) below, is called initial value problem (IVP).

That is, an nth order initial value problem is of the form

$$\left.\begin{array}{ll} f(x, y, y', y'', \ldots\ldots\ldots\ldots, y^{(n)}) = 0 & \\ y'(x_o) = \alpha_i \quad ; & i = 1(1)\, n-1 \end{array}\right\} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.2)$$

In general any equation of type (1.2) can be reduced to vector equation of the form

$$y' = f(x,y) \; ; \qquad y(x) = \alpha_i \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.3)$$

where,

$\alpha_i = (\alpha_0, \quad \alpha_1, \quad \alpha_2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots, \alpha_{n-1})^T$

$f = (f_1, \quad f_2, \quad f_3, \ldots\ldots\ldots\ldots\ldots\ldots\ldots, f_n)^T$

$y = (y_1, \quad y_2, \quad y_3, \ldots\ldots\ldots\ldots\ldots\ldots, y_n)^T$ and $y_i = y^{(i-1)}$,(i=1,2,3,..n-1)

The subject of differential equations constitutes a large and very important branch of mathematics. From early days till the present day, the subject has been an area of great theoretical research and practical applications.

The problems that can arise in connection with equation (1.3) include;

(i) f(x,y) is a real valued function

(ii) f(x,y) contains discontinuities in the form of finite jumps in the components of f itself or some derivatives of f.

(iii) f(x,y) is non-linear

(iv) Eigen value λ of the Jacobian $J = \frac{df}{dy}$ of f is large (i.e. stiff problems).

(v) Lower order discontinuous derivatives in the solution.

There are several existing algorithms designed to solve non-linear and real valued problems. These includes;

(a) Runge – Kutta methods, Euler's methods and Taylor's series expansion as discussed in Lambert (1973).

(b) Hybrid methods by Ademiluyi (1987)

(c) Implicit BDM (Backward difference method) by Gear (1971)

Similarly, several researchers had carried out a lot of research on stiff equations. These includes: Lambert (1973), Fatunla (1982), Ademiluyi (1985, 1987, 1988, 1992) and Gear (1971) just to mention a few.

DEFINITION

A function f(x,y) is said to be discontinuous when the following conditions are satisfied:

(i) f(x,y) is infinite or unbounded

(ii) The partial derivatives f_x ,f_y are large and unbounded, at some points (x,y)

There are few literature available in this area.

The features of ordinary differential equation that attracts our interest in this work, is the Phenomenon of low order discontinuous derivatives in the solution. This class of ordinary differential equations arises in a large variety of application areas such as electrical networks, economy affected by inflation and other perturbation problems. The nature of ordinary differential equations with lower order discontinous derivatives will be considered in the next section.

1.2. NATURE OF ORDINARY DIFFERENTIAL EQUATIONS WHOSE SOLUTIONS POSSESS SINGULARITIES AND THE ASSOCIATED PROBLEMS

The mathematical formulation of physical situations in simulation, electrical engineering, control theory and economy often leads to first order initial value problem in ordinary differential equations of the form.

$$y' = f(x,y); \qquad y(o) = y_o \qquad \text{...............}(1.4)$$

,in which the solution contains discontinuous low order derivative or singularities (Fatunla, 1986). A simple example is the following

$$y' = y^2$$

$$y(o) = 1 \; ; \qquad 0 \le x \le 2 \qquad \ldots\ldots (1.5)$$

whose theoretical solution is

$$y(x) = \frac{1}{1-x} \qquad \ldots\ldots (1.6)$$

At the point x = 1, y(x) is undefined, and its first order derivative does not exist, hence the differential equation has solution with low order discontinuity at this value of x. A similar example is the non-linear first order equation.

$$y' = 1 + y^2 \; , \qquad y(o) = 1$$

$$0 \le x \le \pi/4 \qquad \ldots\ldots (1.7)$$

whose theoretical solution is given by

$$y(x) = \tan(x + \pi/4) \qquad \ldots\ldots (1.8)$$

This is clearly a problem with singularity at $x = \pi/4$. At this point, the solution does not have first order derivative. Incidentally, the conventional numerical integrators are in general formulated on the basis of polynomial interpolation with the assumption that the IVP (1.3) satisfies the uniqueness theorem.

Consequently, such algorithms perform poorly when they are applied to IVPs which violates this theorem. They are inefficient as they merely track the solution which "explodes" in the neighbourhood of this singularity, Carver(1978).

In this regard, the alternative strategies are based on non-polynomial interpolating functions such as the perturbed polynomial or rational functions.

The resultant algorithms often behave nicely in the neighbourhood of singularities provided the step size is chosen as to sand-wich the points of singularity, Lambert(1973), Ellison(1981).

1.3. EXISTING METHODS

As we have earlier mentioned, the discovery of singularity in many mathematical models of physical situation, have made modern researchers generate a lot of interest in this class of problems.

The existing methods designed for singular or discontinuous low order derivative IVPs includes:

(i) The switching function techniques by Evans and Fatunla (1975),

(ii) Fractional step method by Evans and Fatunla (1975),

(iii) The inverse interpolation method by Hay et al. (1974) and

(iv) The local error estimator technique by Gear and Qsterly (1984). The method detects and locates the points of discontinuities.

This was incorporated into an existing automatic code called GEAR by Hindmarch in 1974. The algorithm also provides estimates of the magnitude and order of the discontinuity and provided an efficient vehicle to restart the integration beyond the point of discontinuity. The presence of a discontinuity is identified by detector

$$d = \left|\frac{Tol}{t_n}\right| \frac{1}{(P+1)} \geq 1 \qquad \text{.................................(1.9)}$$

where Tol is the allowable error tolerance, P the order of the method, and t_n is an estimate of the local truncation error.

The discontinuity is normally located with a desired accuracy by repeatedly halving the step size. In another development, Lambert and Shaw (1965) proposed an algorithm in which the theoretical solution to (1.3) is represented by the following perturbed polynomials.

$$F(x) = P_L(x) + \begin{cases} b|A + x|^N & ; N \notin \{0, 1 \ldots\ldots L\} \\ b|A + x|^N \log|A + x| & ; N \in \{0, 1, \quad L\} \end{cases} \quad \ldots(1.10)$$

with $P_L(x)$ being a polynomial of degree L defined by

$$P_L(x) = \sum_{r=0}^{L} a_r x^r \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1.11)$$

and the second term on the right hand side(r.h.s) being the perturbed term. A and N are the singularity parameters with A controlling the location of the singularity while N determines the nature of singularity. However the unfortunate aspect of this scheme is that it can only handle IVPs whose singularity are restricted to those of (1.10), Bo, H-L et.al.(1983).

This perhaps led Lambert and Shaw (1966) to suggest an alternative procedure called Rational Functions approximation methods.

Rational functions are quotients of polynomials and so constitute a much richer class of functions. Their accurate representations of problems with singularities make them principal targets for function approximation, Alkinson(1978).

This property perhaps motivated Lambert and Shaw (1966) to present a procedure based on a local representation of the theoretical solution to (1.3) by specialised form of rational function that is,

$$F(x) = \frac{P_m(x)}{b + x} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1.12)$$

where $P_m(x)$ is a polynomial of degree m. However the resultant integration formulae from (1.12) can only cope with special and small class of singular initial value problems. The satisfactory performance of the formulae however, motivated Luke et. al (1975) to suggest a more general interpolating function.

$$R(x)=\frac{P_{\mu}(x)}{Q_{v}(x)} \qquad \ldots\ldots\ldots\ldots(1.13)$$

which eliminates the need to characterise the nature of singularities. The singularities are specified by the zeros of $Q_v(x)$. The complications and amount of computations involved in the generation of the parameters of the resultant formulae led Fatunla (1982a) to suggest the adoption of a special variant of (1.13) in the form.

$$y(x) = \frac{A}{1+\sum_{j=1}^{k} b_j x^j} \qquad \ldots\ldots\ldots\ldots(1.14)$$

$k \geq 1$, where the parameters A and b_j's are real coefficients as the form of the solution to the differential equation.

He considered the case $k = 1$ and came up with a first order computational method of the form

$$y_{n+1}= \frac{y_n^{2}}{y_n - hy'} \qquad \ldots\ldots\ldots\ldots(1.15)$$

which he christened as Inverse Euler. The performance evaluation of the method showed that it has wider region of absolute stability $(-\infty, 0)$ than the traditional Euler method.

$$y_{n+1}=y_n+ hf_n \qquad \ldots\ldots\ldots\ldots(1.16)$$

whose region of absolute stability is only $(-2,0)$.

In this work therefore, we would complement his work by extending the schemes to the cases $k = 2$ and $k = 3$ methods and implement them on a micro- computer to establish their applicability.

1.4 AIMS AND OBJECTIVES

The aims and objectives of this work are to:

(i) Develop a numerical integration algorithm based on rational approximation techniques (1.14) for the integration of ordinary differential equations(ODEs) with low order discontinuous derivatives.

(ii) Theoretically analyse its consistency, convergence and stability properties.

(iii) Code the algorithms in a computer programming language (FORTRAN)

(iv) Implement the programme with specific sample problems on a micro-computer with a view to establishing its applicability and suitability.

1.5 MOTIVATION

The large variety of application areas of this class of ordinary differential equations to human endeavours motivated the research work.

1.6 RESEARCH METHODOLOGY

To accomplished the above aims and objectives, we went into literature review, adopted Taylor and Binomial series expansion to generate the parameters of the new schemes.

The analysis of the error, consistency, convergence and stability properties were carried out by adopting Pade's approximation technique, Dahlquist stability theorems, and Richardson extrapolation methods respectively. The algorithm was coded in Fortran programming language and implemented on a micro-computer to confirm the workability and accuracy of the newly developed schemes with sample problems.

1.7 . ORGANIZATION OF WORK

The remaining chapters of the thesis are organized as described below.

In chapter two, the general principles of one-step schemes were examined because they are relevant to the proposed schemes.

Chapter three discusses the development of the new schemes. Particularly we considered the one-stage scheme, two-stage scheme, and three-stage schemes.

Their error, consistency, convergence and stability properties were all fully discussed in chapter four. While chapter five, considers the implementation of the schemes on a micro-computer using some sample problems. Finally, chapter six summarises the whole thesis and makes some appropriate recommendations.

CHAPTER TWO

PRELIMINARY CONCEPTS AND PRINCIPLES

2.1 PRINCIPLES OF ONE-STEP SCHEMES

We shall first discuss the principles of one –step schemes, since the proposed schemes are based on these principles.

The general one-step schemes for solution of the differential equation of type (1.3) is the method in which the approximation y_{n+1} to the solution at point x_{n+1} can be generated from the knowledge of y_n at x_n.

Generally, one-step schemes are written in the form

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

where $\phi\,(x_n, y_n, h)$ is a function of the arguments x,y,h and in addition depends on f as in equation (1.4). This function $\phi(x_n, y_n, h)$ is called the increment function.

THE FAMILIES OF ONE-STEP SCHEMES INCLUDES

(a) Euler scheme

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

(b) Taylor's series method of the form

$$y_{n+1} = y_n + hf\,(x_n, y_n,) + \frac{h^2}{2} f'(x_1, y_x) + \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.3)$$

(c) and Runge – Kutta formula

$$y_{n+1} = y_n + \sum_{i=1}^{R} C_i k_i \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.4)$$

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

and R- is the stage of the method.

2.2 PROPERTIES OF ONE-STEP SCHEMES

The ability of a numerical scheme to reliably control the global error is a major and significant aspect that needs to be considered. Thus, the global discretisation error associated with (2.1) is given by

$$\ell_{n+1} = y_{n+1} - y(x_{n+1}) \qquad \text{.......................(2.5)}$$

where y_{n+1} is the numerical solution at step x_{n+1} and $y(_{x+1})$ is the theoretical solution. It is required that the error be made as small as possible by making h sufficiently close to zero.

To make this concept clearer, the following definitions are given:-

Definition 1

The local truncation error T_{n+1} associated with one-step schemes (2.2) is defined as the amount by which the theoretical solution y(x) of the initial value problem (1.3) fails to satisfy the difference equation (2.1). That is,

$$T_{n+1} = y(x_{n+1}) - y(x_n) - h\,\phi(x_n\, y(x_n), h) \qquad \text{................(2.6)}$$

Obviously, there exists a relationship between the global error defined in (2.5) and the local truncation error (2.6). The relationship is

$$|\ell_{n+1}| \le C\,|T_{n+1}| \qquad \text{..................................(2.7)}$$

where C is a constant. From (2.6), it is clear that the local discretization error is directly proportional to the local truncation 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.

Definition 2

The integration formula (2.1) is said to be consistent with the IVP (1.3) provided the increment function ϕ (x,y,h) satisfies the following relationship:

$$\phi\,(x_{n,}\ y_{n,}\ o) = f(x,y) \qquad \text{.. (2.8)}$$

as h tends to zero

The consistency of a one-step numerical integration ensures that the scheme is at least of order one.

Definition 3

The one-step scheme (2.1) is considered convergent provided that for an arbitrary initial solution vector y_o and an arbitrary point $x_o \in (a,b)$, there exists a global error, e_n such that

$$\lim_{h \to 0} e_n = 0 \qquad \text{..............................(2.9)}$$

Definition 4

The integration formula (2.1) is said to be of order p if p is the largest positive integer such that the local truncation error (L.t.e) t_{n+1} satisfies

$$t_{n+1} = 0(\,h^{p+1}) \qquad \text{...(2.10)}$$

where $0(h^{p+1})$ implies the existence of finite constants C and $h_o > o$ such that

$$\|t_{n+1}\| \le Ch^{p+1} \qquad \text{...(2.11)}$$

for all $h \leq h_o$.

Consistency demands that order $P \geq 1$.

A numerical solution to an initial value problem in ordinary differential equation is said to be accurate, if its numerical results do not deviate significantly from the corresponding values of the exact solution, while a numerical solution that is not stable is said to have unbounded discretisation error, Dahlquist(1959).

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

Induced instability is the characteristics of the numerical methods while the inherent instability is a bye-product of mathematical transformation of the real situation into differential equation.

To further explain these concepts, we shall consider the following scalar initial value problem

$$y' = \lambda y \;;\quad a \leq x \leq b \;; y(x_o) = y_o \quad \text{......} (2.12)$$

with $\text{Re}(\lambda) < 0$.

Its theoretical solution is

$$y(x) = y_o \ell^{\lambda x} \quad \text{......} (2.13)$$

Supposing we slightly change the initial condition (2.12) by $\alpha > o$, so that the initial condition becomes

$$y(x_o) = y_o + \alpha \quad \text{......} (2.14)$$

Therefore, the solution of (2.12) together with condition (2.14) yields

$$y(x) = (y_o + \alpha)\ \ell^{\lambda x} \qquad \text{...} (2.15)$$

Consequently not minding the integration schemes used, and for h > o, no matter how small, it is seen that the second term $\alpha\ \ell^{\lambda x}$ in (2.15) eventually grow exponentially as the computation proceeds, if Re(λ) > o, thus the solution y(x). will become unstable for slight change in the initial condition.

This kind of instability is said to be inherent; A differential equation may be stable while the numerical scheme gives unstable solution due to truncation, round off and error propagation. This class of instability is termed induced instability.

It arises as a result of adopting

(i) finite interation steps instead of infinite Interation process during computer implementation.

(ii) truncation associated with the process of derivation of the scheme.

This can normally be detected by applying the integration scheme to solve the scalar stability test equation (2.12). Its instability will show in the form of spurious exponentials. The instability can be minimised by reducing the step size. The stability of one step schemes can be defined in terms of global error. The following definition further elucidates the concept.

Definition 6

One-step scheme is said to be stable, if for any initial error e_0, there exist a constant K and $h_o > o$ such that when (2.1) is applied to initial value problem (1.3) with step size $h \in (o, h_o)$, the ultimate error ℓ_n satisfies the following inequality.

$$\ell_n \leq K \ell_o , \quad 0<K<1, \qquad \text{…… (2.16)}$$

One-step schemes is said to be absolutely stable for a given step size h_0 and for initial value problem (1.3), if the errors tends to zero, as the step size decreases.

This absolute stability property of any numerical method of type (2.1) can be investigated by applying it to the scalar test problem (2.12) which yields the following first order difference equation.

$$y_{n+1} = \mu(z)y_n \qquad \text{……(2.17)}$$

where $z = \lambda h$

$\mu(z)$ is called the stability function of the schemes.It can either be polynomial or rational function with the latter being a more desirable property.

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

CHAPTER THREE

THE PROPOSED SCHEMES

Rational functions are quotients of polynomials. They are more adequate and capable of yielding better results compared with polynomial interpolation schemes (Lambert, 1973) in all cases where the function to be interpolated has a pole or singularities. Hence it constitutes a much richer class of functions than polynomials. This greatly increases their prospect for accurate approximation.

Since polynomials do not have singularities (Fatunla, 1978), it breeds endless generations of smooth derivatives, they are not really suited for problems with singularities, as such, poor results are usually obtained.

The principal target of rational approximation therefore, are the problems with singularities. Even with non-singular functions there are occasions when rational approximations may be preferred.

3.1 DERIVATION OF THE NEW SCHEMES

In this work , we assumed that the numerical approximation y_{n+k} at $x = x_{n+k}$ to the exact solution $y(x_{n+k})$ can be represented as

$$y_{n+k} = \frac{y_n}{1+\sum_{j=1}^{k} b_j x^j_n} \qquad \text{...}(3.1)$$

Where the parameter b_j's are to be determined from the non-linear systems of equations generated by adopting the following steps:

(i) obtained the Taylor series expansion of y_{n+k} about point (x_n, y_n) for $k = 1,2,3,\ldots\ldots$

(ii) obtained the Binomial series expansion of the right hand side of (3.1)

(iii) insert the series expansions into (3.1)

(iv) compare the coefficients of the powers of h in the final expression

These parameters bj; are then chosen as to ensure that one or more of the following conditions are satisfied.

(a) adequate high order of accuracy of the scheme

(b) minimum bound of local truncation error

(c) minimum storage requirement.

(d) maximum interval of absolute stability.

We illustrated the scheme with k=1,2 and 3.

3.1.1 ONE STAGE SCHEME

Setting k = 1 in equation (3.1), we obtained a general one step formula in the form

$$y_{n+1} = \frac{y_n}{1+b_1x_n} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.2)$$

Adopting binomial expansion theorem on the right hand side of (3.2)as stated above and ignoring terms of order higher than one, we have

$$y_{n+1} = y_n \left[1 - b_1 x_n\right] + 0(x^2) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.3)$$

The first order Taylor expansion of y_{n+1} about y_n gives

$$y_{n+1} = y'_n + h\,y'_n + 0\,(h^2) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.4)$$

Substituting (3.4) into (3.3) we obtained

$$y_n + h\,y'_n = y_n - b_1 y_n x_n \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.5)$$

The coefficient b_1 is evaluated by imposing the condition that equation (3.5) agrees term by term. Hence

$$-b_1 y_n x_n = h\,y'_n \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.6)$$

and

$$b_1 = \frac{-hy'_n}{y_n x_n} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.7)$$

Substituting (3.7) into (3.2), we obtained a one step integrator of order 1 in the form

$$y_{n+1} = \frac{y_n^{\,2}}{y_n - hy'_n} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.8)$$

which incidentally coincides with Fatunla's inverse Euler method in (1.15).

3.1.2 TWO STAGE SCHEME

Similarly, by setting $k = 2$, in (3.1), we obtained a two-stage method of the form.

$$y_{n+2} = \frac{y_n}{1 + b_1 x_n + b_2 x_n^{\ 2}} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.9)$$

Adopting binomial series expansion on the right hand side of (3.9) and ignoring terms of order higher than two, equation (3.9) modifies into

$$y_{n+2} = y_n - y_n\, b_1 x_n - y_n b_2 x_n^{\ 2} + y_n b_1^{\ 2} x_n^{\ 2} + 0(x^3) \qquad \ldots\ldots.(3.10)$$

By the second order Taylor series expansion of y_{n+2} about the point $x = x_n$ we have

$$y_{n+2} = y_n + h\, y'_n + \frac{h^2}{2}\ y''_n + 0(h^3) \qquad \ldots\ldots\ldots\ldots\ldots\ldots(3.11)$$

Inserting (3.11) into (3.10), equation (3.11) becomes

$$y_n + h\, y'_n + \frac{h^2}{2}\ y''_n + 0(h^3) = y_n - y_n\, b_1 x_n - y_n b_2 x_n^{\ 2} + y_n b_1^{\ 2} x_n^{\ 2} + 0(x^3) \quad ..(3.12)$$

The coefficient b_1 and b_2 are evaluated by ensuring that equation (3.12) agrees term by term. Consequently we obtained

$$-b_1\, y_n x_n = h\, y'_n$$

$$b_1 = \frac{-hy'_n}{y_n x_n}$$

$$-x_n^2 (y_n b_2 - y_n b_1^2) = \frac{h^2 y''_n}{2} \qquad \dots\dots\dots\dots\dots\dots\dots\dots(3.13)$$

$$y_n b_2 - y_n b_1^2 = \frac{h^2 y''_n}{2x_n^2} \qquad \dots\dots\dots\dots\dots\dots\dots\dots(3.14)$$

$$b_2 - b_1^2 = \frac{h^2 y''_n}{2x_n^2} \qquad \dots\dots\dots\dots\dots\dots\dots\dots(3.15)$$

$$b_2 = b_1^2 + \frac{h^2 y''_n}{2x_n^2} \qquad \dots\dots\dots\dots\dots\dots\dots\dots(3.16)$$

Solving these equations, we get

$$b_2 = \frac{h^2}{2y_n^2 x_n^2}\left[2(y'_n)^2 - y_n y''_n\right] \qquad \dots\dots\dots\dots\dots\dots\dots\dots(3.17)$$

Using these in (3.9), we obtained a two stage method

$$y_{n+2} = \frac{2y_n^3}{2y_n^2 - 2hy_n\, y'_n + h^2\,(2(y'_n)^2 - y_n y''_n)} \qquad \dots\dots\dots\dots\dots\dots(3.18)$$

of order 2

3.1.3 THREE STAGE SCHEME

Similarly, the general three stage scheme is of the form

$$y_{n+3} = \frac{y_n}{1 + b_1 x_n + b_2 x_n + b_3 x_n^3} \qquad \dots\dots\dots\dots\dots\dots\dots(3.19)$$

Adopting Binomial series expansion on the right hand side of (3.19) and also a Taylor series expansion of y_{n+3} on the left of (3.19) about $x = x_n$ we have

$$y_n + h\,y'_n + \frac{h^2}{2}y''_n + \frac{h^3 y'''_n}{3!} + 0(h^4) = y_n - b_1 - y_n x_n - y_n x_n^2 (b_1 - b_1^2) - x^3_n y_b \qquad \text{............ (3.20)}$$

and ensuring that the expansion in (3.20) agrees term by term. Thus we have

$$-b_1 y_n x_n = h\,y'_n$$

$$-x^2_n y_n (b_2 - b_1^2) = \frac{h^2 y''_n}{2}$$

$$-\ x^3_n y_n (b_3 - 2b_1 b_2 + b_1^3) = \frac{h^3 y''_n}{6} \qquad \text{................(3.21)}$$

Solving the above set of equations, we have

$$b_1 = \frac{-h^2 y'_n}{y_n x_n}$$

$$b_2 = \frac{h^2}{2x_n^2}\left[2(y'_n)^2 - y_n y''_n\right]$$

$$b_3 = \frac{-h^3 y_n^2 y'''_n + 6h^2 y_n y'_n y''_n - 6h^3 (y')^3}{6y^3_n x_n^3} \qquad \text{....................(3.22)}$$

Substituting the value of b_1, b_2 and b_3 into (3.19), we obtained the 3- stage integrator of order three, as

$$y_{n+3} = \frac{6y_n^4}{6y_n^3 - 6hy^2_n y'_n + 3h^2 y_n (2(y'_n)^2 - y_n y''_n) - h^3 (6(y'_n)^3 - 6yy'_n y''_n + y_n y'''_n)}$$

Higher order schemes can be obtained in this way.

A cursory look at the Sequence of the stages and orders of the methods, shows that, a one stage method is of order 1, two stage method is of order 2 and three stage method is of order 3, thence it can be concluded that a k- stage method is of order k .

The next chapter is devoted to the examination of the consistency, convergence and stability properties of the new schemes.

CHAPTER FOUR

PROPERTIES OF THE NEW SCHEMES

As a result of the mode of the development of the scheme, errors occur. Consequently, analysis of these errors and the consistency, convergence and stability properties of the new schemes are therefore very essential to enable us to know whether such methods are capable of solving ordinary differential equations of our interest (that is, differential equations whose solution contains low order discontinuous derivatives.)

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. Therefore, in this chapter, we shall examine these properties of the new schemes.

4.1 Error Analysis

As mentioned above, the errors associated with numerical approximation techniques, like the one under consideration for ordinary differential equations with singularities, can arise from sources that can be majorly classified into truncation, round-off and discretisation errors respectively.

Round-off (or computation) error is an error introduced by the computing devices or arithmetic operations. It can be mathematically expressed as

$$r_{n+1} = y_{n+1} - P_{n+1} \qquad \ldots\ldots\ldots\ldots\ldots\ldots(4.1)$$

where y_{n+1} is the expected solution of the finite difference equation (3.1) while P_{n+1} is the computer output at x_{n+1}. It is the amount by which the computed approximation P_{n+1} differs from the expected approximation y_{n+1} by the scheme at point x_{n+1}.

The way and manners machine operations are performed in terms of storage and manipulation of numbers determines the magnitude of the errors.

However, from the work of researchers like Fatunla (1987), Lambert (1973) and Alkinson (1978) we observe that the effects of this kind of error can be disastrous, because of the inevitable loss of accuracy it introduces. This class of error is not amendable to analysis but its size can be controlled by employing double precision arithmetic.

Truncation error on the other hand can occur when a limiting process is broken off before coming to the limiting value, as a result of stopping the iteration after a finite number of steps. This is often defined as the amount by which the true solution y (x_{n+1}) of the differential equation (1.3) fails to satisfy the difference equation of the scheme, and it is given by the expression

$$T_{n+k} = y(x_{n+1}) - \frac{y(x_n)}{1+\sum_{j=1}^{k} b_j x^j_n} \qquad \text{.............................(4.2)}$$

The bound of this local truncation error for the scheme (3.2) can be obtained by adopting Taylor series expansion for $y(x_{n+1})$ and expressing the results as

$$T_{n+1} = \psi(x_n, y(x_n))h^p + 0(h^{p+1}) \qquad \text{........................(4.3)}$$

where p is its order of accuracy while $\psi(x_n, y(x_n)$ refers to the principal truncation error associated with the scheme. It can serve as a measure of the largest or principal error introduced at each stage of the integration scheme. It measures the deviation of the numerical solution of the ordinary differential equations from the exact solution.

For example, the local truncation error for the family of one-stage methods of order one is

$$T_{n+1} = \left[\frac{2y'_n y_n - y_n^{\ 2}}{y_n - hy'_n}\right]h^2 \qquad \text{..........................(4.4)}$$

Discretisation errors are introduced as a result of replacing differential equation (1.3) by its difference equivalence (4.2). The discretisation error associated with this transformation is mathematically expressed as

$$\ell_{n+1} = y(x_{n+1}) - y_{n+1} \qquad \text{..........................(4.5)}$$

This is the difference between the exact solution $y(x_{n+1})$ and the numerical solution y_{n+1} generated by the scheme at point x_{n+1}.

Since numerical solution involves iteration process, there will be propagation of error from step to step because when iterating with a numerical scheme, we obtained a sequence of values $\{y_j\}$ j = 1(1)n. If y_1 has an error and since y_2 depends on y_1, y_2 will be subject to error. Similarly since y_3 depends on y_2 and so on, errors are accumulated and the final solutions may be in serious inaccuracy.

Propagated error can subsequently grow to the extent of distorting the numerical result. One of the main objective of the numerical solution is to be able to control the growth of such errors

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

Therefore, in order to guarantee the quality of the integration scheme. It is essential to have estimate of these errors.

The error estimation procedure for the schemes is based on Richardson extrapolation process, where by the local truncation error is estimated from the difference between two predictions using the same scheme with different step sizes h and h/2.

If y_{n+1} designate the solution by single step size h, the local discretisation error as defined by (4.5) can be rewritten as suggested by Lambert (1973) as

$$y(x_{n+1}) - y_{n+1} = \psi(x_n, y(x_n), h)h^p + 0(h^{p+1}) \qquad \text{....................(4.6)}$$

where p is refers to the order of accuracy and h as step size.

Similarly, suppose we compute another approximation L_{n+1} to y (x_{+1}) by applying scheme (3.3) with h/2 as step size. Then, it follows that

$$y(x_{n+1}) - L_{n+1} = \psi(x_n, y(x_n), \frac{h}{2})[\frac{h}{2}]^p + 0\,(h^{p+1}) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.7)$$

subtracting (4.6) from (4.7) and simplifying, we got

$$\psi(x_n, y(x_n)h^{p+1} = \left[y_{n+1} - L_{n+1}\right]\left[\frac{2^p}{1-2^p}\right] \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.8)$$

Therefore, the local discretisation error of the scheme can be estimated from

$$\ell_{n+1} = \left[y_{n+1} - L_{n+1}\right]\left[\frac{2^p}{1-2^p}\right] \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.9)$$

Thus, the approximation y_{n+1} from step x_n to x_{n+1} is accepted as a good approximation to the exact solution, if the global error is less than error tolerance (Tol), that is,

$$|\ell_{n+1}| < \text{Tol} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.10)$$

Experts like Lambert (1973) , Gear (1971), Hindmarch (1983) have found this form of error estimate to be adequate for ordinary differential equations.

This estimate can then be used to choose reasonable step size that will accelerate convergence of the scheme. However, it involves considerable amount of computational efforts.

4.2 THE CONSISTENCY PROPERTY

Recall that the conventional one-step numerical integrator for the IVP (1.3) was described as follows:

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

subtracting y_n from both sides, we obtained

$$y_{n+1} - y_n = h\phi\,(x_n, y_n, h) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.11)$$

dividing both sides by h, we get

$$\frac{y_{n+1} - y_n}{h} = \phi\,(x_n, y_n, h) \qquad \text{.............................. (4.12)}$$

By our definition 2, (page 12 chapter two) the integration formular is consistent with the IVP (1.3) provided the increment function $\phi\,(x,y,h)$ satisfies the following

$$\phi\,(x,y,o) = f(x,y).$$

We shall in this section demonstrate the consistency of our proposed schemes.

1. <u>One-Stage Scheme</u>

From equation (3.8), a one-stage scheme is

$$y_{n+1} = \frac{y_n^{\,2}}{y_n - hy'_n}$$

Subtracting y_n from both sides, we obtained

$$y_{n+1} - y_n = \frac{hy_n\ y'_n}{y_n - hy'_n} \qquad \text{....................(4.13)}$$

which on dividing both sides by h, we have

$$\frac{y_{n+1} - y_n}{h} = \frac{y_n y'_n}{y_n - hy'_n} \qquad \text{..................(4.14)}$$

As h tends to zero, equation (4.14) leads to

$$\frac{y_{n+1} - y_n}{h} \rightarrow y'_n = f(x_n, y_n) \qquad \text{..................(4.15)}$$

Suggesting that the one-stage integrator is consistent.

II. TWO STAGE SCHEME

Recall that the two-stage scheme in equation (3.18) is

$$y_{n+2} = \frac{2y_n^{3}}{2y^2_n - 2hy_n y'_n + h^2(2(y'_n)^2 - y_n y''_n)}$$

subtracting y_n from both sides we obtained

$$y_{n+2} - y_n = \frac{h(2y_n^{2} y'_n - hy'_n(2(y'_n)^2 - y_n y''_n)}{2y_n^2 - 2hy_n y'_n + h^2(2(y'_n)^2 - y_n y''_n)}$$

Dividing both sides by h, we get

$$\frac{y_{n+2} - y_n}{h} = \frac{2y_n^2 y'_n - hy'_n(2(y'_n)^2 - y_n y''_n)}{2y_n^2 - 2hy_n y'_n + h^2(2(y'_n)^2 - y_n y''_n)} \qquad \text{............ (4.16)}$$

As h → o, then

$$\frac{y_{n+2} - y_n}{h} \rightarrow y'_n$$

Implying that the two - stage scheme is consistent.

III. Three – Stage Scheme

From (3.27) a three stage scheme is

$$y_{n+3} = \frac{6y_n^{4}}{6y_n^{3} - 6hy_n^2 y'_n + 3h^2 y_n(2(y'_n)^2 - y_n y''_n) - h^3(6(y'_n)^3 - 6y_n y'_n y''_n + y_n y'''_n)}$$

following a similar procedure as before, we have that

$$\frac{y_{n+3}-y_n}{h}=\frac{6y_n^3y_n'-6hy_n^2(y_n')^2+3hy_n^3y_n''-h^2y_n^3y_n'''-2h^2y_n^2y_n'y_n''-4h^2y_n(y_n')^2}{6y_n^3-6hy_n^2y_n'+3h^2y_n(2(y_n')^2-y_ny_n'')-h^3(6(y_n')^3-6y_ny_n'y_n''+y_ny_n''')}$$

which as h tends to zero, leads to

$$\frac{y_{n+3}-y_n}{h}\rightarrow y_n' \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.19)$$

By induction, it can be shown that the general scheme is consistent and hence convergent (Lambert, 1973).

4.3. CONVERGENCE PROPERTIES

As mentioned earlier that any error introduced at any stage of the computation which is not bounded can produce un-stable numerical results.

Therefore, in this section, we examine the behavior of the error generated by the proposed schemes, with a view to accessing its suitability.

Here, we intend to check, whether the schemes satisfies our definition 6 in chapter two. This shall be established for the one – stage scheme and the general case.

I. One stage scheme

Recall that the one stage scheme discussed in chapter three is

$$y_{n+1}=\frac{y^2_n}{y_n-hy_n'} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.20)$$

The theoretical solution y(x) to the equation (1.3) is seen to satisfy

$$y(x_{n+1})=\frac{y^2(x_n)}{y(x_n)-hy'(x_n)}+T_{n+1} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.21)$$

where T_{n+1} refers to the truncation error. Subtracting (4.20) from (4.21), we have

$$y(x_{n+1}) - y_{n+1} = \frac{y^2(x_n)}{y(x_n) - hy'(x_n)} - \frac{y_n^{\;2}}{y_n - hy'_n} + T_{n+1} \qquad \text{............(4.22)}$$

which by adopting the definition

$y(x_{n+1}) - y_{n+1} = \ell_{n+1}$ and simplifying, we obtain

$$\ell_{n+1} = \frac{y_n^{\;2}\ell_n}{[y(x_n) - hy'(x_n)][y_n - hy'_n]} + T_{n+1} \qquad \text{..................(4.23)}$$

By a first order Taylor series approximation, we have

$y_{n-1} = y_n - h\,y'_n$ and

$y(x_{n-1}) = y(x_n) - h\,y'(x_n)$

Therefore equation (4.23) approximately reduces to

$$\ell_{n+1} = \frac{y_n^{\;2}\ell_n}{(y_{n-1})(y_{n-1})} + T_{n+1} \qquad \text{........................(4.24)}$$

Taking modulus on both sides of (4.24), we obtained

$$|\ell_{n+1}| \le \left|\frac{y_n^{\;2}}{y^2_{\;n-1}}\right| \ell_n + T_{n+1} \qquad \text{....................................(4.25)}$$

Setting

$$\left|\frac{y_n{}^2}{y^2{}_{n-1}}\right| = \left|\left(\frac{y_n}{y_{n-1}}\right)^2\right| = K \qquad \text{..................................(4.26)}$$

Then the inequality (4.25) becomes

$$|e_{n+1}| \le K|e_n| + T_{n+1} \qquad \text{..................................(4.27)}$$

Let T = $\operatorname{Sup}|T_{n+1}|$ and $E_{n+1} = \underset{0\le n\le\infty}{\operatorname{Sup}}|e_{n+1}|$

Therefore

$E_{n+1} < K\,E_n + T$, $\quad n = 0, 1, 2$ ....(4.28)

$E_1 < K\,E_0 + T$

$E_2 < KE_1 + T$

Substituting the value of E_1, we got

$E_2 < KE_1 + T = K(KE_0 + T) + T$

$E_2 < K^2E_0 + KT + T$

Also $E_3 < KE_2 + T$

And substituting the value of E_2, we got

$E_3 < K^3E_0 + K^2T + KT + T$

Going in this way, it is not difficult to see that ,

$$E_n < K^nE_0 + \sum_{r=1}^{n} K^{r-1}T \qquad \text{..................................(4.29)}$$

Provided

$K < 1$

Then as $n \longrightarrow \infty$

$\lim_{n\to\infty} E_n \longrightarrow 0$

- Showing that the scheme converges. According to Lambert (1963), a consistent, and convergent method is stable. Therefore the method is stable for the one stage scheme.

II. **General Case**

Recall that the general scheme is

$$y_{n+k} = \frac{y_{n+k-1}}{1+\sum_{j=1}^{k} b_j x^i_{n+k}} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.30)$$

- The theoretical solution $y(x)$ is seen to satisfy equation

$$y\,(x_{n+k}) = \frac{y(x_{n+k-1})}{1+\sum_{j=1}^{k} b_j x^i_{n+k}} + T_{n+k} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.31)$$

Subtracting (4.30) from (4.31), we obtained

$$y(x_{n+k}) - y_{n+k} = \frac{y(x_{n+k-1})}{1+\sum_{j=1}^{k} b_j x^i{}_{n+k}} - \frac{y_{n+k-1}}{1+\sum_{j=1}^{k} b_j x^j{}_{n+k}} + T_{n+k} \qquad \ldots\ldots\ldots(4.32)$$

adopting the definition

$y(x_{n+k}) - y_{n+k} = \ell_{n+k}$, and simplifying, we got,

$$y(x_{n+k}) - y_{n+k} = \frac{y(x_{n+k-1}) - y_{n+k-1}}{1+\sum_{j=1}^{k} b_j x^j_{n+k}} + T_{n+k} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.33)$$

Simplifying and adopting 4.5, we obtained

$$\ell_{n+k} = \frac{\ell_{n+k-1}}{1+\sum_{j=1}^{k} b_j x^j_{n+k}} + T_{n+k} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.34)$$

taking modulus on both sides of (4.34), we got

$$\left|\ell_{n+k}\right| \le \left|\frac{\ell_{n+k-1}}{1+\sum_{j=1}^{k} b_j x^j{}_{n+k}}\right| + \left|T_{n+k}\right| \qquad \text{..............................(4.35)}$$

$$\left|\frac{1}{1+\sum_{j=1}^{k} b_j x_{n+k}^j}\right| = \frac{1}{1+\sum_{j=1}^{k} b_j x_{n+k}^j} \le \frac{1}{\left|1+\sum_{j=1}^{k} b_j x_{n+k}^j\right|}$$

setting $Q=\left|1+\sum_{j=1}^{k} b_j x_{n+k}^j\right|$

Then, $\left|\frac{1}{1+\sum_{j=1}^{k} b_j x_{n+k}^j}\right| = \frac{1}{Q} = M \qquad ; Q>0$

Then

$$\left|\ell_{n+k}\right| \le M\left|\ell_{n+k-1}\right| + \left|T_{n+k}\right| \qquad \text{.....................................(4.36)}$$

Let $T = \text{Sup}\ (T_{n+k})$ and $M<1$

Similarly setting

$E_{n+k} =$ Sup. ℓ_{n+1}, then the inequality (4.36) modifies into

$o < n < \infty$

$E_{n+k} \le M\ E_{n+k-1} + T$

Hence, for

$k = 1$

We have

$E_{n+1} \le ME_n + T$

$E_{n+2} \le M^2E_n + MT + T$

$E_{n+3} \le M^3E_n + M^2T + MT + T$

Following this trend, we see that,

$$E_{n+k} \le M^k E_n + \sum_{r=0}^{k-1} M^r T \qquad \ldots\ldots(4.37)$$

Now, $$|\ell_{n+k}| < E_{n+k} < M^k E_n + \sum_{r=0}^{k-1} M^r T \qquad \ldots\ldots(4.38)$$

Since $M < 1$, then

as $n \to \infty$

$E_{n+k} \to 0$

Hence the general method is convergent.

4.4 STABILITY PROPERTIES OF THE SCHEMES

To access the suitability of our schemes, we apply the schemes to the Dalhquist (1963) stability scalar test initial value problem.

$$y' = \lambda y$$

$$y(x_0) = y_0 \qquad \ldots\ldots(4.39)$$

under the assumption that Re $(\lambda) \le 0$. (λ is a complex constant with negative real part). A one – step application of the method to (4.39) yields the following first order difference equation.

$$y_{n+1} = \mu(z) y_n, \quad z = \lambda h \qquad \ldots\ldots(4.40)$$

where the stability function $\mu(z)$ is either a polynomial or a rational function in z, with the latter being a more desirable property. (Fatunla, 1986).

The parameters in (2.1) are chosen to ensure that $\mu(z)$ is a Pade's approximation to ℓ^z. To understand the concept of Pade's approximation, we consider the followings:-

Definition 1

Let P_r (z) denote a polynomial of degree r in z as specified by

$$P_r(z) = 1 + \frac{r}{2r!} + \frac{r(r-1)}{(2r)(2r-1)}\frac{z^2}{r!} + \ldots\ldots\ldots$$

$$+ \ldots + \frac{r(r-1)\ldots\ldots 1}{(2r)(2r-1)(r-1)} \quad \frac{z^r}{r!} \qquad \ldots\ldots\ldots\ldots\ldots\ldots(4.41)$$

The function

$$R_{r,s}(z) = \frac{P_r(z)}{P_s(-z)} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.42)$$

Can be shown to be an (r,s) Pade approximation to ℓ^z or equivalently, we say, $R_{r,s}$ (z) is a rational Pade's approximation to ℓ^z.

This $R_{r,s}$ (z) is a rational approximation of order r+s, if

$$R_{r,s}(z) = \ell^z + 0\,(z^{r+S+1}) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.43)$$

Which can be achieved by expressing ℓ^z as a power series in z^i and equating the coefficient of z^i, i = 1(1) r+s in equation

$$\sum_{i=1}^{n} a_i z^i = \left(\sum_{i=0}^{s} b_i z^i\right)\left(\frac{z^i}{i!}\right) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.44)$$

to define uniquely the coefficient a_is, b_is appearing in $R_{r,s}$ (z).

Definition 2

The scheme (3.1) is said to be absolutely stable if a point (z, $\mu(z)$) lies in the complex u-plane, that is, if the stability function satisfies.

$$|\mu(z)| < 1 \qquad \text{...(4.45)}$$

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

$$R = \left\{ z : |\mu(z)| < 1 \right\} \qquad \text{..............................(4.46)}$$

Definition 3

The numerical scheme (3.1) is said to be A- stable if the region of absolute stability, AS, specified in (4.46) includes the entire half of the complex plane denoted by

$$AS = \{ z \in C / \mathrm{Re}(z) < o \} \qquad \text{......................................(4.47)}$$

A stability property is one of the desirable properties for any numerical schemes for stiff and non stiff ordinary differential equations as suggested by Dalhquist (1963), Other stability criterion that qualify numerical scheme to perform effectively and efficiently includes L – stability and A(α) stability which normally are associated with one step schemes. These concept are explained briefly in the following definitions.

Definition 4

The integration scheme (3.1) is said to be A(α) stable, $\alpha \in [0, \pi/2]$, If its region of absolute stability contains infinite wedge Sα define by

$$S\alpha = \{ z / -\alpha < \text{Arg}(-z) < \alpha, z \neq 0 \} \qquad (4.48)$$

The largest α, is called the angle of absolute stability. The region of Sα is shown in Fig. 4.2 . In order to apply A(α) stability conditions, we verify whether the eigenvalues of the systems lie within a certain wedge Sα. It is said to be A(0) stable if it is A(α) stable for some $\alpha \in [0, \pi/2]$.

Definition 5

The numerical schemes (3.1) is said to be L-stable if it is A-stable, in addition, it's stability function $\mu(z)$ must satisfy the condition.

$$\underset{z \to -\infty}{\text{Limit}} / \mu(z) / = o \qquad (4.49)$$

Thus an L- stable method is necessarily A-stable and A(α) – stable.

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

Definition 6

A Pade's approximation to ℓ^{z} is said to be

(i) A – acceptable if $|P_{r,s}(z)| < 1$ whenever Re(z) < o

(ii) A(o) – acceptable if $|P_{r,s}(z)| < 1$ whenever z is real and negative.

(iii) L- acceptable, if it is A-acceptable and in addition satisfies Limit $|P_{r,s}(z)| \rightarrow o$ as Re (z) tends to negative infinity.

Birkoff and Virga (1965) gave further theorems on $P_{r,s}(z)$ as

(i) if r = s $P_{r,s}(z)$ is A – acceptable

(ii) if r = s + 1 or r = s + 2 $P_{r,s}(z)$ is L - acceptable

Liniger and Willoughby (1967) considered stages one and two of Pade's approximation to ℓ^{z}

$$R_{1,1}(z,\gamma) = \frac{1+\frac{1}{2}(1-\gamma)z}{1-\frac{1}{2}(1+\gamma)z} \qquad \text{.........................(4.50)}$$

and

$$R_{2,2}(z,\gamma) = \frac{1+\frac{1}{2}(1-\gamma)z+\frac{1}{4}(r-\gamma)^{z}}{1-\frac{1}{2}(1+\gamma)z+\frac{1}{4}(r+\gamma)z^{2}} \qquad \text{......................(4.51)}$$

With the following conditions.

(i) $P_{1,1}(Z,\gamma)$ is A- acceptable, if and only if γ > o,

L – acceptable, if and only if γ = 1

(ii) $P_{2,2}(Z,Y,r)$ is A – acceptable if and only if γ > o, and r ≥ o and L- acceptable, if and only α = r > o.

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

4.4.1 ONE-STAGE SCHEMES

Recall that the one stage scheme is of this form

$$y_{n+1} = \frac{y_n^2}{y_n - hy'_n} \qquad \text{.............................(4.52)}$$

To analyse the stability properties, we shall adopt the definitions, we earlier discussed on pages 35-36 . Applying (4.20) to the stability test equation (4.39), we obtained the recurrent relation.

$$y_{n+1} = \left[\frac{1}{1-\lambda h}\right] y_n \qquad \text{..........................(4.53)}$$

as the approximation to the solution, where $\frac{1}{1-\lambda h}$ is the stability function. For the convergence of the solution, we consider the behaviour of the stability function

$$\mu(z) = \frac{1}{1-\lambda h} = \frac{1}{1-z} \qquad \text{......................(4.54)}$$

where $z = \lambda h$

Now , a Binomial series expansion of (4.54), we obtain

$$\mu(z) = 1 + \lambda h - \lambda^2 h^2 + \ldots\ldots..$$

That is, $\mu(z) = 1 + z - z^2 + \ldots\ldots 0(z^3)$ .... (4.55)

Which by a pade's approximation, it is equivalent to $\mu(Z) = e^z + 0(h^3)$

However, the stability function (4.54) satisfies (4.45) with $(-\infty, 0)$ as the corresponding interval of absolute stability. This implies that the scheme is A-stable (see definition 3, equation (4.47)).

4.4.2 TWO – STAGE SCHEME

The two-stage scheme is of the form

$$y_{n+2} = \frac{2y_n^3}{2y_n^2 - 2hy_ny'_n + h^2(2(y'_n)^2 - y_ny''_n)}$$

Applying this formular to the test equation (4.39), we obtained

$$y_{n+2} = \left[\frac{1}{1 - \lambda h + \frac{\lambda^2 h^2}{2}}\right] y_n \qquad \text{.........................(4.56)}$$

whose stability function is

$$\mu(z) = \frac{1}{1 - z + \frac{z^2}{2}} \qquad \text{.........................(4.57)}$$

The scheme is A- stable by our definition 3, since it can be expressed as

$$\mu(z) = 1 + z - \frac{z^2}{2} + \ldots = e^z + 0(z^3) \qquad \text{..............(4.58)}$$

and the fact that it satisfies $\lim_{z \to -\infty} /\mu(z)/ = 0$ .... (4.59)

Its interval of absolute stability is $(-\infty, 0)$, showing that the scheme is also A – acceptable, since

$/ R_{2,2}(z)/ < 1$

with $Re(z) < 0$ (see fig.4.1).

It is also A(0) acceptable since $/ R_{2,2}(z)/ < 1$ with negative real part.

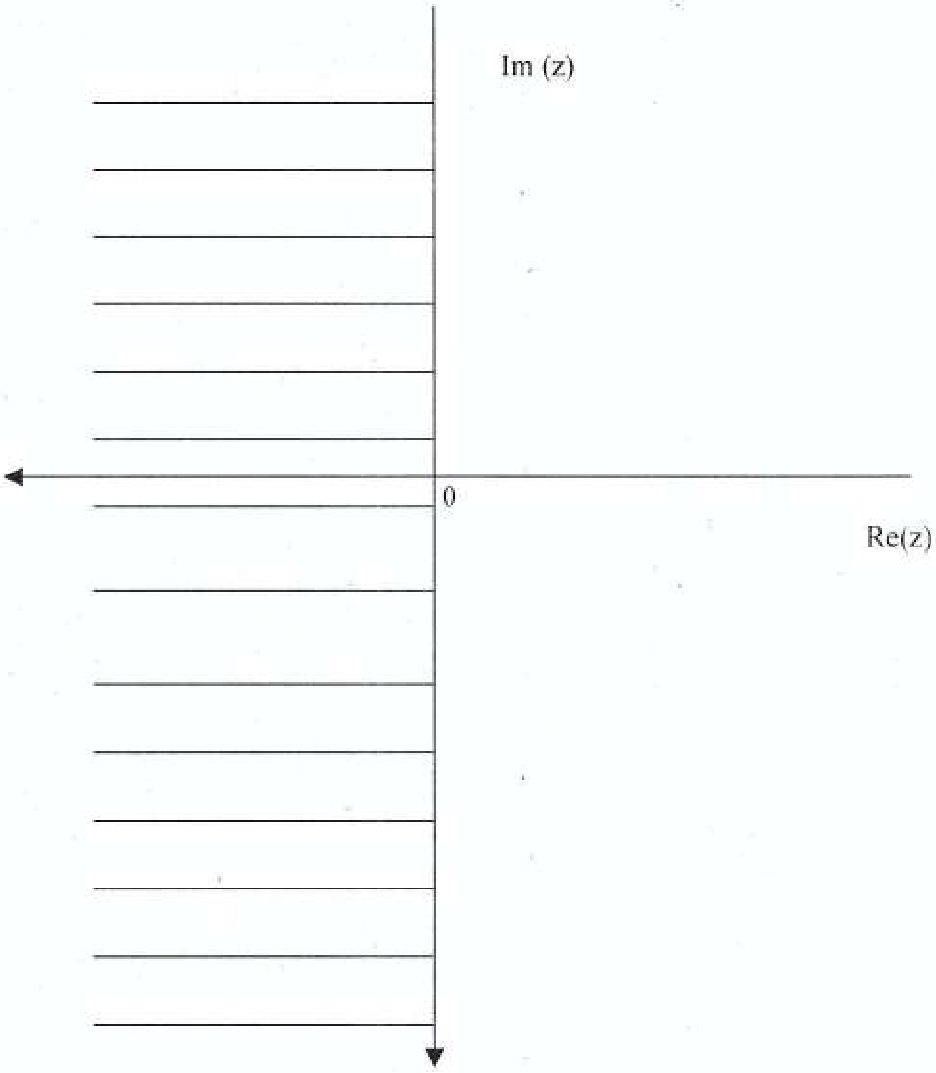


Fig. 4.1: A – Stability Region (shaded portion)

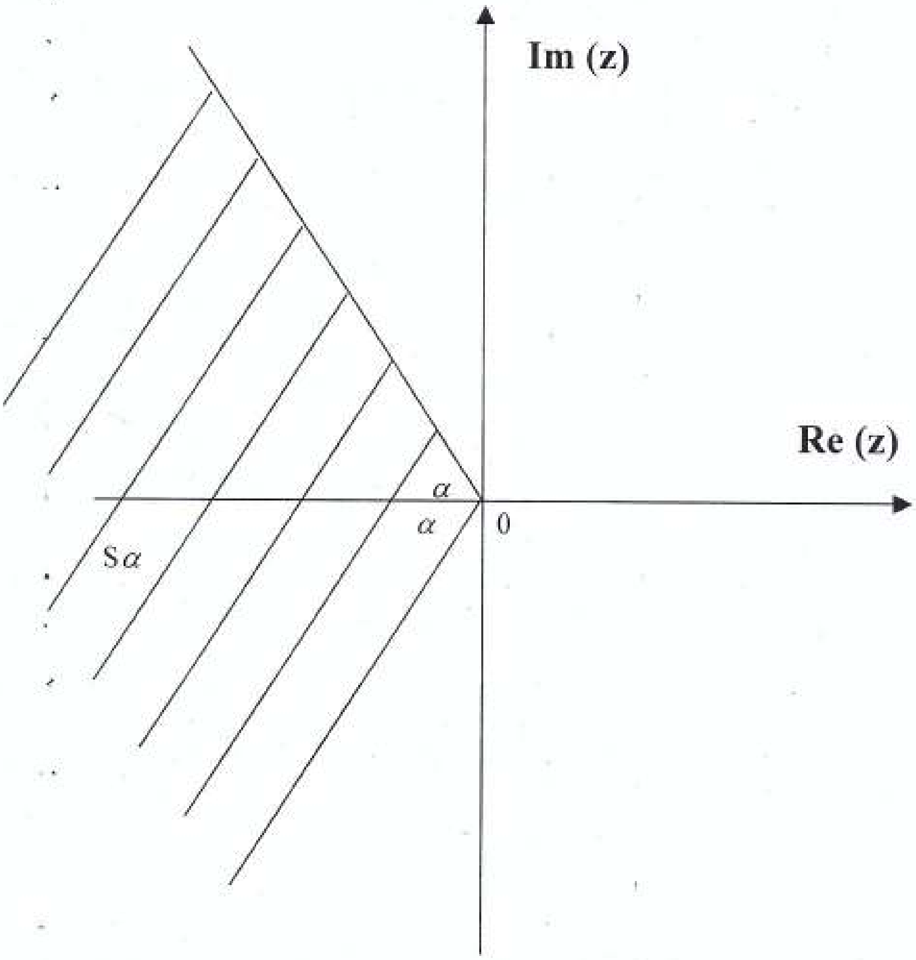


Fig. 4.2: Region of A (α) Stability (shaded portion) Widlund(1967)

CHAPTER FIVE

COMPUTER IMPLEMENTATION AND NUMERICAL RESULTS

In order to demonstrate the applicability and suitability of the new scheme, there is the need to translate the new numerical formula(3.1) into computer codes. This will involve the writing of the formula (3.1) in computer algorithm called pseudo-code.

Some of the computer languages that are available are FORTRAN, BASIC, PASCAL, and so on.

In this work, we considered the fortran programming language as the mode of implementation of the schemes.

To achieve this, we adopted the following steps.

(i) Re-write the formular in an algorithmic form
(ii) Translate the algorithm into a computer flow chart.
(iii) Translate the flow chart into computer code
(iv) Implement the code with sample problems on a digital computer
(v) Discuss the results.

5.1 COMPUTATIONAL ALGORITHM

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

In this section, we develop the numerical algorithm for implementing the inverse polynomial methods described in chapter three, most especially formula (3.1) and adopt the error estimation discussed in chapter four and the step size control measures.

The algorithm is given below:-

```
STEP 1:  Declaration of variables
STEP 2:  Define function
             F(x,y), y exact
STEP 3:  Selection of input values
             $X_o$ , $x_{last}$, $y_o$, h, tol
STEP 4:  Initialise variables by setting
             n =0 (counter)
             $x = x_o$
             $y = y_o$
             $H = h_{old}$
             P = O
STEP 5:  Compute the approximate values of y(x), adopting routine
    IMPRRK (x,y,h , tol, $y_{n+1}$)
For i=2,N
    $X_i = x_{i-1} + h_{old}$
$Y_{N(i)} = (y(i-1) **2) / y(i-1) - h_{old} * y_p (i-1))$
$P_{error} = ABS( ( Y_N^T - y_i ) )$

Write (2,10)     $X_i$, $Y_N$, $y_T$ , $P_{error}$
STEP 6:   ESTIMATE the local Truncation Error (L.T.E.) using subroutine
ADAPT (x,y,h,tol, L te, $h_{NEW}$ , $y_{N+1}$)
SET   $y_L = y^{**} 2/(y - \frac{H}{2} * y_p )$
$D_{NEW}$ = Abs (tol * $y_{new}$)
Call IMPRRK ( $x_o$, $y_o$, h, tol, $y_L$)
SET $h_{NEW} = 0.5 * h_{old}$
```

```
SET  $Xi = x_o + h_{NEW}$
SET
L.T.E  =  $\left[\frac{2^P}{1-2^P}\right]\left[y_N - y_L\right]$
$D_{OLD}$ =  Abs  $\left[y_N - y_L\right]$
$D_{NP}$  =  $(D_{New}/D_{old})$
While  $(D_{old} < D_{new})$
Then
SET $h_{new} = h_{old} * (D_{np})**0.25$
Else
SET
$H_{new} = h_{old} * (D_{np})**0.2$
STEP 7 :  While (L.T.E < 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
$X_n = x_{n+1}$
$y_n = y_{n+1}$
$h_{old} = h_{new}$
```

```
N =  n+ 1
and repeat step 5 - 10
Else
STEP 11            STOP.
```

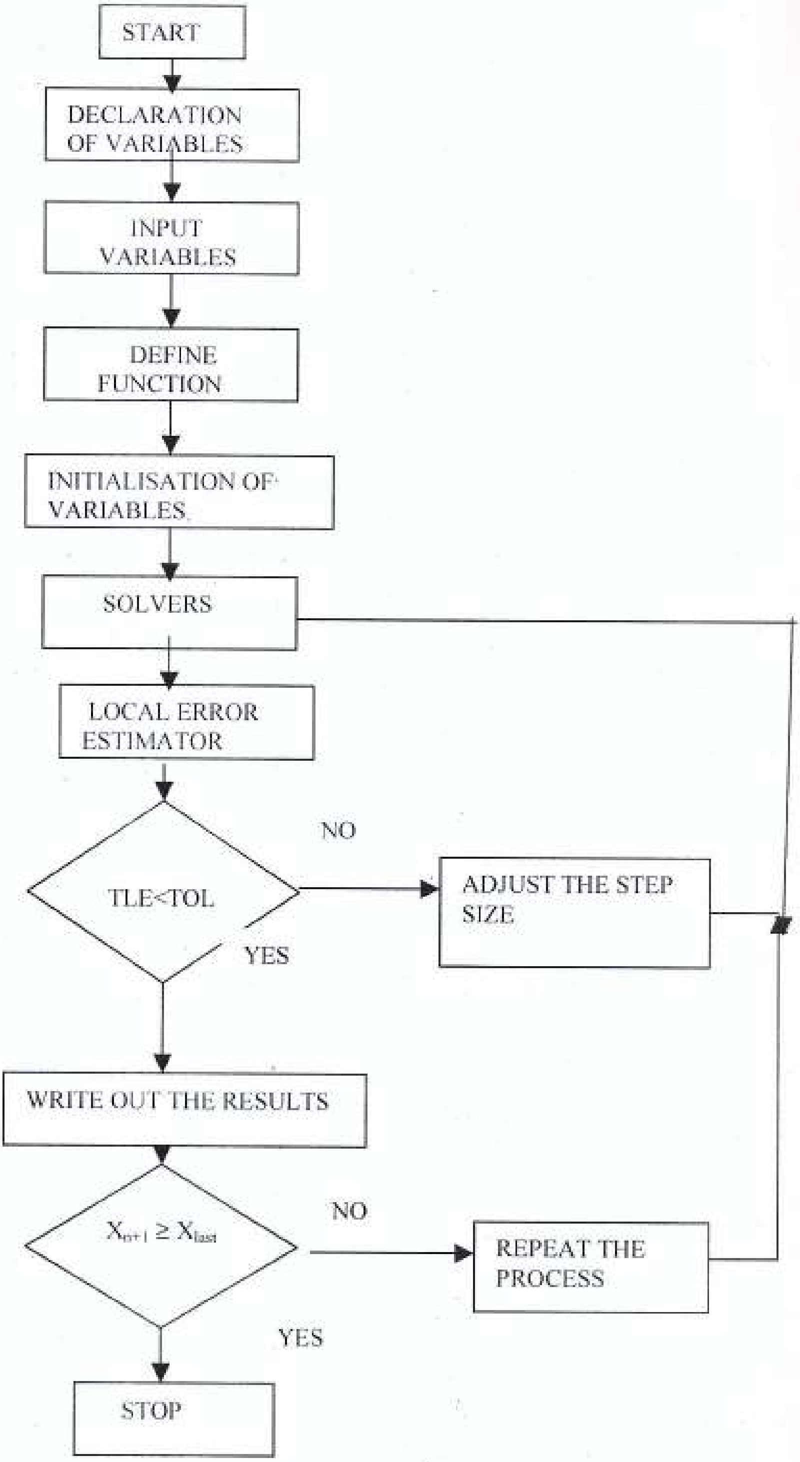
5.2 PROGRAM FLOW CHART

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 at the a point 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 process and so on.

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

Fig. 5.1 : **FLOW CHART OF THE IMPLEMENTATION**



5.3 PROGRAMMING IMPLEMENTATION

This section considers the computer implementation of the above alogorithm. The implementation is done in a variable step size fixed order method. The program consists of three modules called FUNC, IMPRRK, and ADAPT respectively.

The program starts by declaring the value of 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 which will call on subroute FUNC to supply 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 values of the error (LTE) associated with the computation. To control the step size which will increase the rate of convergence.

On the receipt of the error estimate from the subrountines 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 programm will ask whether the upper point is being reached, if yes, it will output result. Otherwise IMPRRK will go to the next step to generate the next round of approximation to the solution. The process will be repeated and continued until the upper end point is reached. When the program reached the end point (x_{last}) it will stop the process.

5.4. NUMERICAL COMPUTATIONS AND RESULTS

To demonstrate the applicability of the various theories discussed in previous chapter, we consider the following sample problems.

Problem 1

Consider initial value problem

$$y' = 1 + y^2; \; y(0) = 1$$

$$0 \leq x \leq 1$$

whose theoretical solution is

$y(x) = \tan(x + \pi/4)$, has singularity at $x = \pi/4$

The numerical results are returns in the variables $y(x_n)$, y_n, e_n, L.T. E as shown in table 1.

Where $y(X_n)$ y_n, e_n, L.T.E representing the exact solution, numerical solution, local discretisation error and local truncation error respectively, as shown in table 5.

PROBLEM 2

Another initial value problem considered is

$$y' = \frac{y + 5x^2 \exp\left[\frac{y}{5x}\right]}{x}; \qquad y(1) = 0$$

$$0 \leq x \leq 1$$

Its theoretical solution is

$y(x) = -5x\log(2 - x)$, with singularity at $x = 2$

The problem was solved numerically with formular (3.8 and 3.18) and the results are shown in table 2.

PROBLEM 3

Consider the initial value problem

$$y' = \frac{y \log y}{1 - x} \; ; \qquad y(o) = \exp(0.2)$$

The theoretical solution is

$$y(x) = \exp \lambda (1 - x)$$

With singularity at x=1.

The results are shown in table 3.

PROBLEM 4

Consider

$$y' = y^2 \; ; \quad y(o) = 1$$

$$0 < X \leq 2$$

The theoretical solution is

$$y(x) = \frac{1}{1 - x}$$

- with singularity at x = 1.

The results are shown in table 4.

Problem 5

Consider this non stiff equation

$$y' = 5y \; ; \; y(0) = 1$$

the theoretical solution is

$$y(x) = \ell^{5x}$$

the results is as shown in table 7

Problem 6

Consider

$$y' = \lambda(y - x^3) + 3x^2 \, ; \, y(0) = 1$$
$$0 \le x \le 2$$

The theoretical solution is
$y(x) = \ell^{\lambda x} + x^3$
where λ is a complex constant with negative real part. That is $\mathrm{Re}(\lambda) < 0$

from the form of the theoretical solution, the transcent part $\ell^{\lambda x}$ varies rapidly with respect to x while the 2nd part x^3 varies slowly. This behaviour is characteristic of stiff equations

the results is shown in table 8

5.5 DISCUSSION OF RESULTS

From the results in tables 1,2,3,4, 7 and 8 we observed that the discretisation error obtained from the solutions are sufficiently small, this shows that the schemes are very accurate, stable and convergent.

As shown from the results in tables 5a and 5b these schemes compared favourably with the existing Euler's method .

Table 1a

```
     ==================================================

       Results of One stage scheme for Equation 1

     ==================================================


h =    .01000

    x                 Y(Xn)                  Yn                en

  .01000          1.0208480201          1.0204081628      .0004398573
  .02000          1.0414806419          1.0412371554      .0002434864
  .03000          1.0625475709          1.0625045999      .0000429710
  .04000          1.0840670155          1.0842290474      .0001620320
  .05000          1.1060581556          1.1064300457      .0003718901
  .06000          1.1285412225          1.1291282059      .0005869833
  .07000          1.1515375604          1.1523452801      .0008077198
  .08000          1.1750697089          1.1761042410      .0010345321
  .09000          1.1991614904          1.2004293688      .0012678784
  .10000          1.2238381030          1.2253463537      .0015082507
  .11000          1.2491262234          1.2508823933      .0017561699
  .12000          1.2750541176          1.2770663144      .0020121968
  .13000          1.3016517630          1.3039286934      .0022769304
  .14000          1.3289510018          1.3315019934      .0025509916
  .15000          1.3569856254          1.3598207145      .0028350891
  .16000          1.3857915938          1.3889215609      .0031299671
  .17000          1.4154071907          1.4188436193      .0034364285
  .18000          1.4458732161          1.4496285538      .0037553377
  .19000          1.4772331974          1.4813208270      .0040876296
  .20000          1.5095336232          1.5139679405      .0044343173
  .21000          1.5428242008          1.5476207037      .0047965028
  .22000          1.5771581412          1.5823335229      .0051753816
```

.23000	1.6125924749	1.6181647307	.0055722559
.24000	1.6491884024	1.6551769490	.0059885465
.25000	1.6870116273	1.6934374913	.0064258640
.26000	1.7261329577	1.7330188152	.0068858575
.27000	1.7666286279	1.7739990167	.0073703888
.28000	1.8085808942	1.8164624043	.0078815101
.29000	1.8520786431	1.8605001262	.0084214831
.30000	1.8972180755	1.9062108702	.0089927947
.31000	1.9441034762	1.9537016794	.0095982032
.32000	1.9928480834	2.0030888471	.0102407637
.33000	2.0435750726	2.0544989385	.0109238659
.34000	2.0964186731	2.1080699618	.0116512887
.35000	2.1515254385	2.1639526845	.0124272460
.36000	2.2090556966	2.2223121547	.0132564581
.37000	2.2691852087	2.2833294252	.0141442165
.38000	2.3321070749	2.3472035567	.0150964818
.39000	2.3980339285	2.4141538965	.0161199680
.40000	2.4672004725	2.4844227518	.0172222793
.41000	2.5398664218	2.5582784800	.0184120582
.42000	2.6163199287	2.6360190658	.0196991371
.43000	2.6968815877	2.7179763515	.0210947638
.44000	2.7819091362	2.8045209868	.0226118506
.45000	2.8718029969	2.8960682580	.0242652611
.46000	2.9670128435	2.9930850410	.0260721975
.47000	3.0680454165	3.0960980568	.0280526403
.48000	3.1754738757	3.2057037986	.0302299229
.49000	3.2899490551	3.3225804753	.0326314201

Table 1b

```
        ==========================================

         Results of two stage scheme for Equation 1

        ==========================================


  h =   .01000

   X              Y(Xn)                Yn                 en

.01000         1.0208480201         1.0201999587        .0006480613

.02000         1.0414806419         1.0408078280        .0006728139

.03000         1.0625475709         1.0618487866        .0006987842

.04000         1.0840670155         1.0833409525        .0007260630

.05000         1.1060581556         1.1053034157        .0007547399

.06000         1.1285412225         1.1277563040        .0007849185

.07000         1.1515375604         1.1507208529        .0008167074

.08000         1.1750697089         1.1742194820        .0008502270

.09000         1.1991614904         1.1982758860        .0008856044

.10000         1.2238381030         1.2229151250        .0009229780

.11000         1.2491262234         1.2481637206        .0009625028

.12000         1.2750541176         1.2740497746        .0010043430

.13000         1.3016517630         1.3006030838        .0010486792

.14000         1.3289510018         1.3278552714        .0010957304

.15000         1.3569856254         1.3558399307        .0011456947

.16000         1.3857915938         1.3845927848        .0011988090

.17000         1.4154071907         1.4141518550        .0012553357

.18000         1.4458732161         1.4445576555        .0013155606

.19000         1.4772331974         1.4758533975        .0013797999

.20000         1.5095336232         1.5080852180        .0014484052

.21000         1.5428242008         1.5413024397        .0015217611

.22000         1.5771581412         1.5755578450        .0016002962
```

.23000	1.6125924749	1.6109079907	.0016844841
.24000	1.6491884024	1.6474135506	.0017748518
.25000	1.6870116273	1.6851396977	.0018719295
.26000	1.7261329577	1.7241565321	.0019764256
.27000	1.7666286279	1.7645395595	.0020890684
.28000	1.8085808942	1.8063702128	.0022106814
.29000	1.8520786431	1.8497364609	.0023421822
.30000	1.8972180755	1.8947334695	.0024846060
.31000	1.9441034762	1.9414643553	.0026391210
.32000	1.9928480834	1.9900410354	.0028070481
.33000	2.0435750726	2.0405851896	.0029898830
.34000	2.0964186731	2.0932293490	.0031893241
.35000	2.1515254385	2.1481181225	.0034073160
.36000	2.2090556966	2.2054096253	.0036460712
.37000	2.2691852087	2.2652770779	.0039081308
.38000	2.3321070749	2.3279106585	.0041964164
.39000	2.3980339285	2.3935196272	.0045143013
.40000	2.4672004725	2.4623347822	.0048656903
.41000	2.5398664218	2.5346113007	.0052551211
.42000	2.6163199287	2.6106320369	.0056878918
.43000	2.6968815877	2.6907113838	.0061702039
.44000	2.7819091362	2.7751997768	.0067093593
.45000	2.8718029969	2.8644890223	.0073139745
.46000	2.9670128435	2.9590185572	.0079942863
.47000	3.0680454165	3.0592829001	.0087625163
.48000	3.1754738757	3.1658405601	.0096333155
.49000	3.2899490551	3.2793246931	.0106243620

Table 2a

===

Results of one stage scheme for Equation 2

==

h = .01000

x	Y(Xn)	Yn	en
.01000	1.2238727362	1.2238504628	.0000222734
.02000	1.2263982671	1.2263527427	.0000455244
.03000	1.2289812448	1.2289114431	.0000698017
.04000	1.2316236459	1.2315284889	.0000951571
.05000	1.2343275382	1.2342058937	.0001216445
.06000	1.2370950870	1.2369457644	.0001493225
.07000	1.2399285599	1.2397503069	.0001782530
.08000	1.2428303332	1.2426218321	.0002085012
.09000	1.2458028983	1.2455627618	.0002401366
.10000	1.2488488690	1.2485756358	.0002732332
.11000	1.2519709888	1.2516631187	.0003078702
.12000	1.2551721395	1.2548280086	.0003441309
.13000	1.2584553493	1.2580732445	.0003821049
.14000	1.2618238056	1.2614019155	.0004218900
.15000	1.2652808567	1.2648172712	.0004635855
.16000	1.2688300312	1.2683227310	.0005073002
.17000	1.2724750466	1.2719218965	.0005531501
.18000	1.2762198225	1.2756185631	.0006012594
.19000	1.2800684944	1.2794167332	.0006517612
.20000	1.2840254286	1.2833206310	.0007047977
.21000	1.2880952389	1.2873347171	.0007605218
.22000	1.2922828036	1.2914637060	.0008190976

.23000	1.2965932853	1.2957125839	.0008807014
.24000	1.3010321517	1.3000866293	.0009455224
.25000	1.3056051911	1.3045914335	.0010137576
.26000	1.3103185580	1.3092329245	.0010856335
.27000	1.3151787801	1.3140173935	.0011613866
.28000	1.3201927945	1.3189515215	.0012412730
.29000	1.3253679801	1.3240424115	.0013255687
.30000	1.3307121934	1.3292976206	.0014145728
.31000	1.3362338073	1.3347251984	.0015086089
.32000	1.3419417543	1.3403337263	.0016080280
.33000	1.3478455738	1.3461323621	.0017132117
.34000	1.3539554641	1.3521308899	.0018245742
.35000	1.3602823401	1.3583397735	.0019425666
.36000	1.3668378974	1.3647702172	.0020676802
.37000	1.3736346824	1.3714342310	.0022004514
.38000	1.3806861710	1.3783447052	.0023414659
.39000	1.3880068561	1.3855154905	.0024913656
.40000	1.3956123442	1.3929614909	.0026508533
.41000	1.4035194646	1.4006987639	.0028207008
.42000	1.4117463903	1.4087446333	.0030017570
.43000	1.4203127742	1.4171178181	.0031949561
.44000	1.4292399020	1.4258385720	.0034013300
.45000	1.4385508643	1.4349288462	.0036220181
.46000	1.4482707506	1.4444124687	.0038582820
.47000	1.4584268693	1.4543153466	.0041115226
.48000	1.4690489963	1.4646656997	.0043832966
.49000	1.4801696588	1.4754943210	.0046753377

Table 2b

```
        ==========================================

        Results of two stage scheme for Equation 2

        ==========================================


h =     .01000

 X             Y(Xn)                 Yn                    en

.01000      1.2238727362         1.2238725366         .0000001996

.02000      1.2263982671         1.2263977306         .0000005365

.03000      1.2289812448         1.2289802237         .0000010211

.04000      1.2316236459         1.2316219809         .0000016651

.05000      1.2343275382         1.2343250581         .0000024801

.06000      1.2370950870         1.2370916075         .0000034795

.07000      1.2399285599         1.2399238824         .0000046775

.08000      1.2428303332         1.2428242437         .0000060895

.09000      1.2458028983         1.2457951664         .0000077319

.10000      1.2488488690         1.2488392470         .0000096220

.11000      1.2519709888         1.2519592096         .0000117792

.12000      1.2551721395         1.2551579156         .0000142239

.13000      1.2584553493         1.2584383712         .0000169782

.14000      1.2618238056         1.2618037370         .0000200686

.15000      1.2652808567         1.2652573384         .0000235184

.16000      1.2688300312         1.2688026754         .0000273559

.17000      1.2724750466         1.2724434354         .0000316112

.18000      1.2762198225         1.2761835055         .0000363170

.19000      1.2800684944         1.2800269853         .0000415091

.20000      1.2840254286         1.2839782029         .0000472257

.21000      1.2880952389         1.2880417298         .0000535090

.22000      1.2922828036         1.2922223992         .0000604044
```

.23000	1.2965932853	1.2965253242	.0000679611
.24000	1.3010321517	1.3009559185	.0000762332
.25000	1.3056051911	1.3055199187	.0000852724
.26000	1.3103185580	1.3102234090	.0000951490
.27000	1.3151787801	1.3150728474	.0001059328
.28000	1.3201927945	1.3200750942	.0001177003
.29000	1.3253679801	1.3252374454	.0001305348
.30000	1.3307121934	1.3305676663	.0001445271
.31000	1.3362338073	1.3360740298	.0001597775
.32000	1.3419417543	1.3417653585	.0001763958
.33000	1.3478455738	1.3476510711	.0001945027
.34000	1.3539554641	1.3537412330	.0002142311
.35000	1.3602823401	1.3600466137	.0002357265
.36000	1.3668378974	1.3665787471	.0002591503
.37000	1.3736346824	1.3733500016	.0002846808
.38000	1.3806861710	1.3803736563	.0003125147
.39000	1.3880068561	1.3876639854	.0003428707
.40000	1.3956123442	1.3952363541	.0003759901
.41000	1.4035194646	1.4031073229	.0004121417
.42000	1.4117463903	1.4112947652	.0004516251
.43000	1.4203127742	1.4198179996	.0004947746
.44000	1.4292399020	1.4286979400	.0005419620
.45000	1.4385508643	1.4379572591	.0005936052
.46000	1.4482707506	1.4476205788	.0006501718
.47000	1.4584268693	1.4577146820	.0007121873
.48000	1.4690489963	1.4682687525	.0007802438
.49000	1.4801696588	1.4793146499	.0008550088

Table 3a

Results of one stage scheme for Equation 3

h = .01000

x	Y(Xn)	Yn	en
.01000	1.0101010099	1.0101010099	.0000000000
.02000	1.0204081628	1.0204081631	.0000000003
.03000	1.0309278343	1.0309278347	.0000000004
.04000	1.0416666657	1.0416666662	.0000000005
.05000	1.0526315756	1.0526315780	.0000000023
.06000	1.0638297815	1.0638297854	.0000000039
.07000	1.0752688089	1.0752688147	.0000000057
.08000	1.0869565108	1.0869565190	.0000000081
.09000	1.0989010852	1.0989010955	.0000000103
.10000	1.1111110946	1.1111111079	.0000000133
.11000	1.1235954861	1.1235955019	.0000000158
.12000	1.1363636137	1.1363636328	.0000000192
.13000	1.1494252614	1.1494252839	.0000000225
.14000	1.1627906783	1.1627906933	.0000000150
.15000	1.1764705759	1.1764705841	.0000000082
.16000	1.1904761854	1.1904761859	.0000000005
.17000	1.2048192797	1.2048192724	.0000000073
.18000	1.2195122058	1.2195121899	.0000000159
.19000	1.2345679203	1.2345678961	.0000000243
.20000	1.2500000279	1.2499999943	.0000000337
.21000	1.2658228221	1.2658227787	.0000000433
.22000	1.2820513291	1.2820512756	.0000000535

.23000	1.2987013560	1.2987012911	.0000000649
.24000	1.3157895418	1.3157894651	.0000000767
.25000	1.3333333863	1.3333333240	.0000000623
.26000	1.3513513884	1.3513513417	.0000000466
.27000	1.3698630338	1.3698630031	.0000000308
.28000	1.3888888912	1.3888888779	.0000000132
.29000	1.4084506877	1.4084506924	.0000000047
.30000	1.4285713921	1.4285714153	.0000000232
.31000	1.4492753047	1.4492753484	.0000000437
.32000	1.4705881554	1.4705882203	.0000000649
.33000	1.4925372099	1.4925372964	.0000000865
.34000	1.5151513865	1.5151514969	.0000001104
.35000	1.5384613833	1.5384615196	.0000001363
.36000	1.5624998166	1.5624999812	.0000001645
.37000	1.5873013741	1.5873015679	.0000001939
.38000	1.6129029808	1.6129032050	.0000002242
.39000	1.6393439836	1.6393442391	.0000002555
.40000	1.6666663521	1.6666666412	.0000002891
.41000	1.6949149015	1.6949152271	.0000003256
.42000	1.7241375377	1.7241379025	.0000003648
.43000	1.7543855283	1.7543859344	.0000004061
.44000	1.7857138029	1.7857142522	.0000004492
.45000	1.8181812862	1.8181817817	.0000004956
.46000	1.8518512673	1.8518518128	.0000005455
.47000	1.8867918120	1.8867924115	.0000005995
.48000	1.9230762221	1.9230768797	.0000006576
.49000	1.9607835483	1.9607842688	.0000007205

Table 3b

```
     ====================================================

          Results of two stage scheme for  Equation 3

     ====================================================


h =   .01000

      x            Y(Xn)                   Yn                  en
.01000        1.0101010099          1.0101010099          .0000000000
.02000        1.0204081628          1.0204081631          .0000000003
.03000        1.0309278343          1.0309278347          .0000000004
.04000        1.0416666657          1.0416666662          .0000000005
.05000        1.0526315756          1.0526315779          .0000000023
.06000        1.0638297815          1.0638297854          .0000000039
.07000        1.0752688089          1.0752688147          .0000000057
.08000        1.0869565108          1.0869565190          .0000000081
.09000        1.0989010852          1.0989010955          .0000000103
.10000        1.1111110946          1.1111111079          .0000000133
.11000        1.1235954861          1.1235955019          .0000000158
.12000        1.1363636137          1.1363636328          .0000000191
.13000        1.1494252614          1.1494252839          .0000000225
.14000        1.1627906783          1.1627906933          .0000000150
.15000        1.1764705759          1.1764705840          .0000000082
.16000        1.1904761854          1.1904761858          .0000000004
.17000        1.2048192797          1.2048192724          .0000000073
.18000        1.2195122058          1.2195121898          .0000000159
.19000        1.2345679203          1.2345678960          .0000000243
.20000        1.2500000279          1.2499999942          .0000000337
.21000        1.2658228221          1.2658227787          .0000000434
.22000        1.2820513291          1.2820512755          .0000000535
```

.23000	1.2987013560	1.2987012911	.0000000649
.24000	1.3157895418	1.3157894651	.0000000767
.25000	1.3333333863	1.3333333240	.0000000623
.26000	1.3513513884	1.3513513417	.0000000466
.27000	1.3698630338	1.3698630031	.0000000308
.28000	1.3888888912	1.3888888779	.0000000133
.29000	1.4084506877	1.4084506923	.0000000047
.30000	1.4285713921	1.4285714153	.0000000232
.31000	1.4492753047	1.4492753484	.0000000437
.32000	1.4705881554	1.4705882203	.0000000649
.33000	1.4925372099	1.4925372964	.0000000865
.34000	1.5151513865	1.5151514970	.0000001104
.35000	1.5384613833	1.5384615196	.0000001363
.36000	1.5624998166	1.5624999812	.0000001645
.37000	1.5873013741	1.5873015679	.0000001938
.38000	1.6129029808	1.6129032050	.0000002241
.39000	1.6393439836	1.6393442391	.0000002555
.40000	1.6666663521	1.6666666413	.0000002892
.41000	1.6949149015	1.6949152272	.0000003256
.42000	1.7241375377	1.7241379025	.0000003648
.43000	1.7543855283	1.7543859345	.0000004062
.44000	1.7857138029	1.7857142523	.0000004494
.45000	1.8181812862	1.8181817819	.0000004957
.46000	1.8518512673	1.8518518129	.0000005457
.47000	1.8867918120	1.8867924117	.0000005997
.48000	1.9230762221	1.9230768798	.0000006577
.49000	1.9607835483	1.9607842689	.0000007206

Table 4a

===

Results of one stage scheme for Equation 4

===

h = .01000

x	Y(Xn)	Yn	en
.11000	-.3501172543	-.3533659522	.0032486979
.12000	-.3787630585	-.3852333606	.0064703021
.13000	-.4068599668	-.4165304731	.0096705063
.14000	-.4344035430	-.4472576742	.0128541312
.15000	-.4613892452	-.4774146157	.0160253705
.16000	-.4878124869	-.5070003663	.0191878794
.17000	-.5136686145	-.5360135157	.0223449012
.18000	-.5389529061	-.5644522464	.0254993403
.19000	-.5636605700	-.5923144073	.0286538373
.20000	-.5877867430	-.6195975450	.0318108019
.21000	-.6113264895	-.6462989568	.0349724673
.22000	-.6342747993	-.6724157079	.0381409086
.23000	-.6566265864	-.6979446582	.0413180718
.24000	-.6783766872	-.7228824843	.0445057971
.25000	-.6995198591	-.7472256950	.0477058359
.26000	-.7200507484	-.7709706394	.0509198910
.27000	-.7399639809	-.7941135240	.0541495430
.28000	-.7592540660	-.8166504153	.0573963493
.29000	-.7779154266	-.8385772502	.0606618236
.30000	-.7959423977	-.8598898395	.0639474418
.31000	-.8133292239	-.8805838762	.0672546522
.32000	-.8300700577	-.9006549389	.0705848812

.33000	-.8461589573	-.9200984942	.0739395369
.34000	-.8615898844	-.9389098999	.0773200154
.35000	-.8763567022	-.9570844095	.0807277072
.36000	-.8904531730	-.9746171710	.0841639980
.37000	-.9038729557	-.9915032293	.0876302736
.38000	-.9166096038	-1.0077375307	.0911279269
.39000	-.9286565626	-1.0233149182	.0946583557
.40000	-.9400071667	-1.0382301393	.0982229726
.41000	-.9506546376	-1.0524778396	.1018232020
.42000	-.9605920806	-1.0660525668	.1054604863
.43000	-.9698124823	-1.0789487733	.1091362910
.44000	-.9783087079	-1.0911608109	.1128521030
.45000	-.9860734976	-1.1026829345	.1166094369
.46000	-.9930994642	-1.1135093016	.1204098375
.47000	-.9993790895	-1.1236339718	.1242548823
.48000	-1.0049047215	-1.1330509062	.1281461847
.49000	-1.0096685704	-1.1417539686	.1320853982
.50000	-1.0136627058	-1.1497369238	.1360742180
.51000	-1.0168790527	-1.1569934383	.1401143856
.52000	-1.0193093880	-1.1635170806	.1442076927
.53000	-1.0209453365	-1.1693013196	.1483559832
.54000	-1.0217783671	-1.1743395250	.1525611579
.55000	-1.0217997890	-1.1786249688	.1568251799
.56000	-1.0210007467	-1.1821508238	.1611500771
.57000	-1.0193722164	-1.1849101639	.1655379475
.58000	-1.0169050014	-1.1868959654	.1699909640
.59000	-1.0135897270	-1.1881011066	.1745113796

Table 4b

Results of two stage scheme for Equation 4

h = .01000

x	Y(Xn)	Yn	en
.11000	-.3501172543	-.3495609343	.0005563200
.12000	-.3787630585	-.3776656279	.0010974306
.13000	-.4068599668	-.4052296914	.0016302754
.14000	-.4344035430	-.4322440570	.0021594861
.15000	-.4613892452	-.4587010539	.0026881914
.16000	-.4878124869	-.4845939030	.0032185839
.17000	-.5136686145	-.5099164039	.0037522106
.18000	-.5389529061	-.5346627272	.0042901789
.19000	-.5636605700	-.5588272823	.0048332877
.20000	-.5877867430	-.5824046250	.0053821180
.21000	-.6113264895	-.6053893990	.0059370905
.22000	-.6342747993	-.6277762881	.0064985112
.23000	-.6566265864	-.6495599855	.0070666009
.24000	-.6783766872	-.6707351694	.0076415178
.25000	-.6995198591	-.6912964862	.0082233730
.26000	-.7200507484	-.7112385357	.0088122128
.27000	-.7399639809	-.7305558662	.0094081148
.28000	-.7592540660	-.7492429558	.0100111101
.29000	-.7779154266	-.7672942133	.0106212133
.30000	-.7959423977	-.7847039688	.0112384290
.31000	-.8133292239	-.8014664702	.0118627537
.32000	-.8300700577	-.8175758802	.0124941776

.33000	-.8461589573	-.8330262688	.0131326885
.34000	-.8615898844	-.8478116144	.0137782700
.35000	-.8763567022	-.8619257966	.0144309056
.36000	-.8904531730	-.8753625955	.0150905775
.37000	-.9038729557	-.8881156886	.0157572671
.38000	-.9166096038	-.9001786479	.0164309559
.39000	-.9286565626	-.9115449348	.0171116278
.40000	-.9400071667	-.9222079014	.0177992653
.41000	-.9506546376	-.9321607830	.0184938546
.42000	-.9605920806	-.9413967004	.0191953802
.43000	-.9698124823	-.9499086531	.0199038292
.44000	-.9783087079	-.9576895179	.0206191900
.45000	-.9860734976	-.9647320457	.0213414519
.46000	-.9930994642	-.9710288592	.0220706050
.47000	-.9993790895	-.9765724491	.0228066404
.48000	-1.0049047215	-.9813551709	.0235495506
.49000	-1.0096685704	-.9853692428	.0242993275
.50000	-1.0136627058	-.9886067411	.0250559647
.51000	-1.0168790527	-.9910595974	.0258194552
.52000	-1.0193093880	-.9927195955	.0265897924
.53000	-1.0209453365	-.9935783676	.0273669689
.54000	-1.0217783671	-.9936273907	.0281509764
.55000	-1.0217997890	-.9928579832	.0289418058
.56000	-1.0210007467	-.9912613007	.0297394460
.57000	-1.0193722164	-.9888283327	.0305438838
.58000	-1.0169050014	-.9855498987	.0313551027
.59000	-1.0135897270	-.9814166437	.0321730833

Table 5a

THE RESULTS OF THE COMPARISON OF THE NEW SCHEME WITH EULER METHOD

Results of One stage scheme for Equation 1				Results of Euler formula for Equation 1	
h = .01000					
x	Y(Xn)	Yn	E1	eYn	E2
.01000	1.0208480201	1.0204081628	.0004398573	1.0199999996	.0008480205
.02000	1.0414806419	1.0412371554	.0002434864	1.0404039994	.0010766425
.03000	1.0625475709	1.0625045999	.0000429710	1.0612284027	.0013191682
.04000	1.0840670155	1.0842290474	.0001620320	1.0824904590	.0015765564
.05000	1.1060581556	1.1064300457	.0003718901	1.1042083145	.0018498411
.06000	1.1285412225	1.1291282059	.0005869833	1.1264010735	.0021401491
.07000	1.1515375604	1.1523452801	.0008077198	1.1490888658	.0024486946
.08000	1.1750697089	1.1761042410	.0010345321	1.1722929177	.0027767912
.09000	1.1991614904	1.2004293688	.0012678784	1.1960356239	.0031258665
.10000	1.2238381030	1.2253463537	.0015082507	1.2203406356	.0034974674
.11000	1.2491262234	1.2508823933	.0017561699	1.2452329485	.0038932748
.12000	1.2750541176	1.2770663144	.0020121968	1.2707389986	.0043151190
.13000	1.3016517630	1.3039286934	.0022769304	1.2968867736	.0047649894
.14000	1.3289510018	1.3315019934	.0025509916	1.3237059259	.0052450759
.15000	1.3569856254	1.3598207145	.0028350891	1.3512278979	.0057577275
.16000	1.3857915938	1.3889215609	.0031299671	1.3794860660	.0063055278

.17000	1.4154071907	1.4188436193	.0034364285	1.4085158829	.0068913078
.18000	1.4458732161	1.4496285538	.0037553377	1.4383550522	.0075181639
.19000	1.4772331974	1.4813208270	.0040876296	1.4690437045	.0081894929
.20000	1.5095336232	1.5139679405	.0044343173	1.5006245978	.0089090254
.21000	1.5428242008	1.5476207037	.0047965028	1.5331433392	.0096808617
.22000	1.5771581412	1.5823335229	.0051753816	1.5666486232	.0105095180
.23000	1.6125924749	1.6181647307	.0055722559	1.6011925015	.0113999734
.24000	1.6491884024	1.6551769490	.0059885465	1.6368306757	.0123577268
.25000	1.6870116273	1.6934374913	.0064258640	1.6736228220	.0133888053
.26000	1.7261329577	1.7330188152	.0068858575	1.7116329535	.0145000042
.27000	1.7666286279	1.7739990167	.0073703888	1.7509298252	.0156988027
.28000	1.8085808942	1.8164624043	.0078815101	1.7915873780	.0169935162
.29000	1.8520786431	1.8605001262	.0084214831	1.8336852316	.0183934115
.30000	1.8972180755	1.9062108702	.0089927947	1.8773092455	.0199088300
.31000	1.9441034762	1.9537016794	.0095982032	1.9225521454	.0215513308
.32000	1.9928480834	2.0030888471	.0102407637	1.9695142100	.0233338734
.33000	2.0435750726	2.0544989385	.0109238659	2.0183040725	.0252710001
.34000	2.0964186731	2.1080699618	.0116512887	2.0690395832	.0273790900
.35000	2.1515254385	2.1639526845	.0124272460	2.1218488299	.0296766086
.36000	2.2090556966	2.2223121547	.0132564581	2.1768712544	.0321844422
.37000	2.2691852087	2.2833294252	.0141442165	2.2342589389	.0349262698
.38000	2.3321070749	2.3472035567	.0150964818	2.2941780659	.0379290090
.39000	2.3980339285	2.4141538965	.0161199680	2.3568105967	.0412233318
.40000	2.4672004725	2.4844227518	.0172222793	2.4223561589	.0448443136
.41000	2.5398664218	2.5582784800	.0184120582	2.4910342503	.0488321715

.42000	2.6163199287	2.6360190658	.0196991371	2.5630867656	.0532331631
.43000	2.6968815877	2.7179763515	.0210947638	2.6387809006	.0581006871
.44000	2.7819091362	2.8045209868	.0226118506	2.7184125468	.0634965893
.45000	2.8718029969	2.8960682580	.0242652611	2.8023102119	.0694927849
.46000	2.9670128435	2.9930850410	.0260721975	2.8908396346	.0761732089
.47000	3.0680454165	3.0960980568	.0280526403	2.9844091687	.0836362477
.48000	3.1754738757	3.2057037986	.0302299229	3.0834761488	.0919977269
.49000	3.2899490551	3.3225804753	.0326314201	3.1885543954	.1013946597

Table 5b

THE RESULTS OF THE COMPARISON OF THE NEW SCHEME WITH EULER METHOD

h = .01000

x	Y(Xn)	Yn	E1	eYn	E2
	Results of two stage scheme for Equation 2			Results of Euler formula for Equation 2	
.01000	1.2238727362	1.2238725366	.0000001996	1.2238455674	.0000271688
.02000	1.2263982671	1.2263977306	.0000005365	1.2263426821	.0000555850
.03000	1.2289812448	1.2289802237	.0000010211	1.2288959314	.0000853134
.04000	1.2316236459	1.2316219809	.0000016651	1.2315072230	.0001164229
.05000	1.2343275382	1.2343250581	.0000024801	1.2341785521	.0001489861
.06000	1.2370950870	1.2370916075	.0000034795	1.2369120055	.0001830815
.07000	1.2399285599	1.2399238824	.0000046775	1.2397097682	.0002187917
.08000	1.2428303332	1.2428242437	.0000060895	1.2425741280	.0002562052
.09000	1.2458028983	1.2457951664	.0000077319	1.2455074827	.0002954157
.10000	1.2488488690	1.2488392470	.0000096220	1.2485123456	.0003365234
.11000	1.2519709888	1.2519592096	.0000117792	1.2515913534	.0003796355
.12000	1.2551721395	1.2551579156	.0000142239	1.2547472735	.0004248660
.13000	1.2584553493	1.2584383712	.0000169782	1.2579830123	.0004723371
.14000	1.2618238056	1.2618037370	.0000200686	1.2613016242	.0005221814
.15000	1.2652808567	1.2652573384	.0000235184	1.2647063204	.0005745363
.16000	1.2688300312	1.2688026754	.0000273559	1.2682004796	.0006295516

.17000	1.2724750466	1.2724434354	.0000316112	1.2717876593	.0006873873
.18000	1.2762198225	1.2761835055	.0000363170	1.2754716068	.0007482157
.19000	1.2800684944	1.2800269853	.0000415091	1.2792562731	.0008122213
.20000	1.2840254286	1.2839782029	.0000472257	1.2831458260	.0008796027
.21000	1.2880952389	1.2880417298	.0000535090	1.2871446655	.0009505734
.22000	1.2922828036	1.2922223992	.0000604044	1.2912574401	.0010253635
.23000	1.2965932853	1.2965253242	.0000679611	1.2954890646	.0011042207
.24000	1.3010321517	1.3009559185	.0000762332	1.2998447392	.0011874125
.25000	1.3056051911	1.3055199187	.0000852724	1.3043299704	.0012752207
.26000	1.3103185580	1.3102234090	.0000951490	1.3089505939	.0013679641
.27000	1.3151787801	1.3150728474	.0001059328	1.3137127993	.0014659808
.28000	1.3201927945	1.3200750942	.0001177003	1.3186231577	.0015696368
.29000	1.3253679801	1.3252374454	.0001305348	1.3236886505	.0016793296
.30000	1.3307121934	1.3305676663	.0001445271	1.3289167038	.0017954896
.31000	1.3362338073	1.3360740298	.0001597775	1.3343152209	.0019185864
.32000	1.3419417543	1.3417653585	.0001763958	1.3398926235	.0020491308
.33000	1.3478455738	1.3476510711	.0001945027	1.3456578951	.0021876787
.34000	1.3539554641	1.3537412330	.0002142311	1.3516206256	.0023348384
.35000	1.3602823401	1.3600466137	.0002357265	1.3577910670	.0024912731
.36000	1.3668378974	1.3665787471	.0002591503	1.3641801871	.0026577103
.37000	1.3736346824	1.3733500016	.0002846808	1.3707997351	.0028349472
.38000	1.3806861710	1.3803736563	.0003125147	1.3776623128	.0030238582
.39000	1.3880068561	1.3876639854	.0003428707	1.3847814503	.0032254058
.40000	1.3956123442	1.3952363541	.0003759901	1.3921716941	.0034406501
.41000	1.4035194646	1.4031073229	.0004121417	1.3998487048	.0036707598

.42000	1.4117463903	1.4112947652	.0004516251	1.4078293631	.0039170272
.43000	1.4203127742	1.4198179996	.0004947746	1.4161318905	.0041808837
.44000	1.4292399320	1.4286979400	.0005419620	1.4247759854	.0044639166
.45000	1.4385508643	1.4379572591	.0005936052	1.4337829736	.0047678907
.46000	1.4482707506	1.4476205788	.0006501718	1.4431759800	.0050947707
.47000	1.4584268693	1.4577146820	.0007121873	1.4529801183	.0054467510
.48000	1.4690489963	1.4682687525	.0007802438	1.4632227114	.0058262850
.49000	1.4801696588	1.4793146499	.0008550088	1.4739335347	.0062361241

THE RESULTS OF ONE STAGE SCHEME ADOPTING RICHARDSON METHOD TO ESTIMATE THE LOCAL TRUNCATION ERROR FOR EQUATION 1

Table 6a

h= 0.01

X	Y(Xn)	Yn	En	L..T.E.
0.01000	1.0208480201	1.0204081628	0.0004398573	-1.0204081628
0.02000	1.0414806419	1.0412371554	0.0002434864	-1.0412371555
0.03000	1.0625475709	1.0625045999	0.0000429710	-1.0625045999
0.04000	1.0840670155	1.0842290474	0.0001620320	-1.0839049835
0.05000	1.1060581556	1.1064300457	0.0003718901	-1.1056862655
0.06000	1.1285412225	1.1291282059	0.0005869833	-1.1279542392
0.07000	1.1515375604	1.1523452801	0.0008077198	-1.1507298406
0.08000	1.1750697089	1.1761042410	0.0010345321	-1.1740351768
0.09000	1.1991614904	1.2004293688	0.0012678784	-1.1978936120
0.10000	1.2238381030	1.2253463537	0.0015082507	-1.2220819331
0.11000	1.2491262234	1.2508823933	0.0017561699	-1.2508823933
0.12000	1.2750541176	1.2770663144	0.0020121968	-1.2730419208
0.13000	1.3016517630	1.3039286934	0.0022769304	-1.2993748326
0.14000	1.3289510018	1.3315019934	0.0025509916	-1.3264000102
0.15000	1.3569856254	1.3598207145	0.0028350891	-1.3829565047
0.16000	1.3857915938	1.3889215609	0.0031299671	-1.3889215609
0.17000	1.4154071907	1.4188436193	0.0034364285	-1.4119707622
0.18000	1.4458732161	1.4496285538	0.0037553377	-1.4421178784
0.19000	1.4772331974	1.4813208270	0.0040876296	-1.4731455678
0.20000	1.5095336232	1.5139679405	0.0044343173	-1.5050993059
0.21000	1.5428242008	1.5476207037	0.0047965028	-1.5380276980

0.22000	1.5771581412	1.5823335229	0.0051753816	-1.5719827596
0.23000	1.6125924749	1.6181647307	0.0055722559	-1.6070202190
0.24000	1.6491884024	1.6551769490	0.0059885465	-1.6431998559

THE RESULTS OF TWO STAGE SCHEME ADOPTING RICHARDSON METHOD TO ESTIMATE THE LOCAL TRUNCATION ERROR FOR EQUATION 1

Table 6b

h =0.01

X	Y(Xn)	Yn	En	L.T.E
0.01000	1.0208480201	1.0201999587	0.0006480613	1.37386666665407E-06
0.02000	1.0414806419	1.0408078280	0.0006728139	1.26356000000882E-05
0.03000	1.0625475709	1.0618487866	0.0006987842	2.45492000002419E-05
0.04000	1.0840670155	1.0833409525	0.0007260630	3.71766666666199E-05
0.05000	1.1060581556	1.1053034157	0.0007547399	5.05776000000087E-05
0.06000	1.1285412225	1.1277563040	0.0007849185	6.48189333333586E-05
0.07000	1.1515375604	1.1507208529	0.0008167074	7.99708000001805E-05
0.08000	1.1750697089	1.1742194820	0.0008502270	9.61145333334320E-05
0.09000	1.1991614904	1.1982758860	0.0008856044	1.13333599999876E-04
0.10000	1.2238381030	1.2229151250	0.0009229780	1.31721066666479E-04
0.11000	1.2491262234	1.2481637206	0.0009625028	1.51384799999950E-04
0.12000	1.2750541176	1.2740497746	0.0010043430	1.72433200000081E-04
0.13000	1.3016517630	1.3006030838	0.0010486792	1.94991199999883E-04
0.14000	1.3289510018	1.3278552714	0.0010957304	2.19195733333362E-04
0.15000	1.3569856254	1.3558399307	0.0011456947	2.45196933333247E-04
0.16000	1.3857915938	1.3845927848	0.0011988090	2.73160000000180E-04
0.17000	1.4154071907	1.4141518550	0.0012553357	3.03268133333331E-04
0.18000	1.4458732161	1.4445576555	0.0013155606	3.35722933333393E-04
0.19000	1.4772331974	1.4758533975	0.0013797999	3.70748799999987E-04
0.20000	1.5095336232	1.5080852180	0.0014484052	4.08597066666812E-04
0.21000	1.5428242008	1.5413024397	0.0015217611	4.49544666666727E-04
0.22000	1.5771581412	1.5755578450	0.0016002962	4.93899333333362E-04
0.23000	1.6125924749	1.6109079907	0.0016844841	5.42005733333480E-04

```
0.24000              1.6491884024    1.6474135506    0.0017748518   5.94249733333404E-04
```

Table 7a

Results of one stage scheme for Equation 5

h = .00100

x	Y(Xn)	Yn	en
.00100	1.0050125211	1.0050251259	.0000126048
.00200	1.0100501676	1.0100755035	.0000253359
.00300	1.0151130647	1.0151512599	.0000381951
.00400	1.0202013410	1.0202525225	.0000511815
.00500	1.0253151223	1.0253794198	.0000642975
.00600	1.0304545366	1.0305320807	.0000775441
.00700	1.0356197123	1.0357106341	.0000909218
.00800	1.0408107762	1.0409152103	.0001044341
.00900	1.0460278628	1.0461459402	.0001180775
.01000	1.0512711001	1.0514029551	.0001318550
.01100	1.0565406193	1.0566863873	.0001457680
.01200	1.0618365520	1.0619963695	.0001598174
.01300	1.0671590308	1.0673330349	.0001740041
.01400	1.0725081886	1.0726965177	.0001883291
.01500	1.0778841591	1.0780869526	.0002027935
.01600	1.0832870718	1.0835044752	.0002174034
.01700	1.0887170717	1.0889492216	.0002321499
.01800	1.0941742897	1.0944213284	.0002470388
.01900	1.0996588620	1.0999209332	.0002620712
.02000	1.1051709259	1.1054481744	.0002772485
.02100	1.1107106191	1.1110031905	.0002925713
.02200	1.1162780802	1.1165861214	.0003080412
.02300	1.1218734483	1.1221971072	.0003236589
.02400	1.1274968633	1.1278362891	.0003394259
.02500	1.1331484657	1.1335038086	.0003553429
.02600	1.1388283970	1.1391998079	.0003714110
.02700	1.1445367990	1.1449244305	.0003876315
.02800	1.1502738145	1.1506778199	.0004040054
.02900	1.1560395870	1.1564601206	.0004205336
.03000	1.1618342605	1.1622714781	.0004372177
.03100	1.1676579799	1.1681120387	.0004540588

.03200	1.1735108799	1.1739819485	.0004710686
.03300	1.1793931177	1.1798813556	.0004882379
.03400	1.1853048402	1.1858104080	.0005055678
.03500	1.1912461953	1.1917692544	.0005230591
.03600	1.1972173315	1.1977580448	.0005407132
.03700	1.2032183982	1.2037769295	.0005585313
.03800	1.2092495452	1.2098260601	.0005765149
.03900	1.2153109235	1.2159055885	.0005946650
.04000	1.2214026844	1.2220156670	.0006129826
.04100	1.2275249805	1.2281564496	.0006314691
.04200	1.2336779645	1.2343280904	.0006501259
.04300	1.2398617905	1.2405307444	.0006689538
.04400	1.2460766131	1.2467645677	.0006879546
.04500	1.2523225874	1,2530297165	.0007071291
.04600	1.2585998698	1.2593263486	.0007264788
.04700	1.2649086172	1.2656546221	.0007460049
.04800	1.2712489872	1.2720146961	.0007657089
.04900	1.2776211384	1.2784067301	.0007855917

Table 7b

Results of two stage scheme for Equation 5

h = .00100

X	Y(Xn)	Yn	en
.00100	1.0050125211	1.0050125001	.0000000210
.00200	1.0100501676	1.0100501254	.0000000422
.00300	1.0151130647	1.0151130016	.0000000631
.00400	1.0202013410	1.0202012555	.0000000855
.00500	1.0253151223	1.0253150146	.0000001078
.00600	1.0304545366	1.0304544061	.0000001305
.00700	1.0356197123	1.0356195591	.0000001532
.00800	1.0408107762	1.0408106021	.0000001740
.00900	1.0460278628	1.0460276653	.0000001975
.01000	1.0512711001	1.0512708790	.0000002211
.01100	1.0565406193	1.0565403743	.0000002450
.01200	1.0618365520	1.0618362831	.0000002689
.01300	1.0671590308	1.0671587374	.0000002934
.01400	1.0725081886	1.0725078704	.0000003181
.01500	1.0778841591	1.0778838162	.0000003429
.01600	1.0832870718	1.0832867089	.0000003629
.01700	1.0887170717	1.0887166834	.0000003883
.01800	1.0941742897	1.0941738759	.0000004137
.01900	1.0996588620	1.0996584227	.0000004393
.02000	1.1051709259	1.1051704605	.0000004654
.02100	1.1107106191	1.1107101274	.0000004917
.02200	1.1162780802	1.1162775621	.0000005181
.02300	1.1218734483	1.1218729037	.0000005446
.02400	1.1274968633	1.1274962917	.0000005715
.02500	1.1331484657	1.1331478672	.0000005985
.02600	1.1388283970	1.1388277712	.0000006258
.02700	1.1445367990	1.1445361456	.0000006534
.02800	1.1502738145	1.1502731330	.0000006816
.02900	1.1560395870	1.1560388771	.0000007098
.03000	1.1618342605	1.1618335220	.0000007385

.03100	1.1676579799	1.1676572124	.0000007675
.03200	1.1735108799	1.1735100941	.0000007858
.03300	1.1793931177	1.1793923135	.0000008042
.03400	1.1853048402	1.1853040177	.0000008225
.03500	1.1912461953	1.1912453539	.0000008414
.03600	1.1972173315	1.1972164714	.0000008601
.03700	1.2032183982	1.2032175193	.0000008789
.03800	1.2092495452	1.2092486471	.0000008981
.03900	1.2153109235	1.2153100060	.0000009174
.04000	1.2214026844	1.2214017475	.0000009369
.04100	1.2275249805	1.2275240241	.0000009564
.04200	1.2336779645	1.2336769882	.0000009764
.04300	1.2398617905	1.2398607943	.0000009963
.04400	1.2460766131	1.2460755968	.0000010163
.04500	1.2523225874	1.2523215510	.0000010365
.04600	1.2585998698	1.2585988127	.0000010571
.04700	1.2649086172	1.2649075393	.0000010779
.04800	1.2712489872	1.2712478885	.0000010987
.04900	1.2776211384	1.2776200186	.0000011199

Table 7c

Results of three stage scheme for Equation 5

h = .00100

x	Y(Xn)	Yn	en
.00100	1.0050125211	1.0050125211	.0000000000
.00200	1.0100501676	1.0100756095	.0000254419
.00300	1.0151130647	1.0151900341	.0000769693
.00400	1.0202013410	1.0203565799	.0001552389
.00500	1.0253151223	1.0255760476	.0002609253
.00600	1.0304545366	1.0308492546	.0003947180
.00700	1.0356197123	1.0361770352	.0005573229
.00800	1.0408107762	1.0415602411	.0007494649
.00900	1.0460278628	1.0469997415	.0009718787
.01000	1.0512711001	1.0524964239	.0012253238
.01100	1.0565406193	1.0580511950	.0015105757
.01200	1.0618365520	1.0636649804	.0018284284
.01300	1.0671590308	1.0693387259	.0021796951
.01400	1.0725081886	1.0750733966	.0025652080
.01500	1.0778841591	1.0808699800	.0029858208
.01600	1.0832870718	1.0867294833	.0034424115
.01700	1.0887170717	1.0926529374	.0039358657
.01800	1.0941742897	1.0986413941	.0044671045
.01900	1.0996588620	1.1046959301	.0050370681
.02000	1.1051709259	1.1108176445	.0056467186
.02100	1.1107106191	1.1170076622	.0062970430
.02200	1.1162780802	1.1232671324	.0069890522
.02300	1.1218734483	1.1295972311	.0077237829
.02400	1.1274968633	1.1359991597	.0085022964
.02500	1.1331484657	1.1424741486	.0093256829
.02600	1.1388283970	1.1490234552	.0101950582
.02700	1.1445367990	1.1556483662	.0111115671
.02800	1.1502738145	1.1623501984	.0120763838
.02900	1.1560395870	1.1691302996	.0130907126
.03000	1.1618342605	1.1759900491	.0141557886

.03100	1.1676579799	1.1829308581	.0152728782
.03200	1.1735108799	1.1899541725	.0164432926
.03300	1.1793931177	1.1970614721	.0176683544
.03400	1.1853048402	1.2042542720	.0189494318
.03500	1.1912461953	1.2115341247	.0202879294
.03600	1.1972173315	1.2189026203	.0216852887
.03700	1.2032183982	1.2263613874	.0231429892
.03800	1.2092495452	1.2339120955	.0246625502
.03900	1.2153109235	1.2415564549	.0262455314
.04000	1.2214026844	1.2492962189	.0278935345
.04100	1.2275249805	1.2571331846	.0296082041
.04200	1.2336779645	1.2650691950	.0313912304
.04300	1.2398617905	1.2731061393	.0332443488
.04400	1.2460766131	1.2812459553	.0351693422
.04500	1.2523225874	1.2894906316	.0371680441
.04600	1.2585998698	1.2978422065	.0392423367
.04700	1.2649086172	1.3063027736	.0413941564
.04800	1.2712489872	1.3148744797	.0436254925
.04900	1.2776211384	1.3235595297	.0459383912

```
Table 8a
=====================================================================

          Results of one stage scheme for Equation 6


=====================================================================

h =    .00100
      X                  Y(Xn)                  Yn                 en
    .00100           .8869204327           .8928571383        .0059367057
    .00200           .7866278601           .7971938725        .0105660124
    .00300           .6976763509           .7117802484        .0141038975
    .00400           .6187834417           .6355181017        .0167346600
    .00500           .5488117378           .5674269189        .0186151811
    .00600           .4867524417           .5066312464        .0198788046
    .00700           .4317108311           .4523494312        .0206386000
    .00800           .3828933805           .4038835698        .0209901893
    .00900           .3395962325           .3606105307        .0210142982
    .01000           .3011951863           .3219739493        .0207787630
    .01100           .2671366050           .2874770726        .0203404675
    .01200           .2369294572           .2566763722        .0197469150
    .01300           .2101382380           .2291758406        .0190376027
    .01400           .1863766895           .2046218998        .0182452102
    .01500           .1653022329           .1826988549        .0173966220
    .01600           .1466110448           .1631248375        .0165137928
    .01700           .1300336095           .1456481797        .0156145703
    .01800           .1153309380           .1300441789        .0147132409
    .01900           .1022910504           .1161122058        .0138211555
    .02000           .0907259379           .1036731314        .0129471935
    .02100           .0804688525           .0925670174        .0120981648
    .02200           .0713719026           .0826510627        .0112791601
    .02300           .0633039209           .0737977676        .0104938468
    .02400           .0561485729           .0658932929        .0097447200
    .02500           .0498026800           .0588359972        .0090333171
    .02600           .0441747317           .0525351298        .0083603980
    .02700           .0391835661           .0469096640        .0077260979
    .02800           .0347571996           .0418872549        .0071300553
    .02900           .0308317894           .0374033100        .0065715206
    .03000           .0273507124           .0334001574        .0060494450
```

.03100	.0242637495	.0298263050	.0055625555
.03200	.0215263654	.0266357786	.0051094132
.03300	.0190990517	.0237875300	.0046884783
.03400	.0169467735	.0212449100	.0042981366
.03500	.0150384582	.0189751965	.0039367382
.03600	.0133465479	.0169491738	.0036026259
.03700	.0118466014	.0151407575	.0032941561
.03800	.0105169418	.0135266579	.0030097161
.03900	.0093383444	.0120860821	.0027477377
.04000	.0082937589	.0108004649	.0025067060
.04100	.0073680638	.0096532315	.0022851676
.04200	.0065478483	.0086295832	.0020817349
.04300	.0058212184	.0077163078	.0018950894
.04400	.0051776262	.0069016094	.0017239832
.04500	.0046077170	.0061749569	.0015672399
.04600	.0041031945	.0055269480	.0014237535
.04700	.0036567015	.0049491891	.0012924876
.04800	.0032617131	.0044341864	.0011724733
.04900	.0029124433	.0039752504	.0010628071

Table 8b

Results of two stage scheme for Equation 6

h = .00100

x	Y(Xn)	Yn	en
.00100	.8869204327	.8871540049	.0002335722
.00200	.7866278601	.7870422305	.0004143704
.00300	.6976763509	.6982276856	.0005513347
.00400	.6187834417	.6194355236	.0006520819
.00500	.5488117378	.5495347693	.0007230316
.00600	.4867524417	.4875220687	.0007696270
.00700	.4317108311	.4325072997	.0007964686
.00800	.3828933805	.3837007809	.0008074004
.00900	.3395962325	.3404019496	.0008057171
.01000	.3011951863	.3019892918	.0007941055
.01100	.2671366050	.2679114360	.0007748310
.01200	.2369294572	.2376792321	.0007497749
.01300	.2101382380	.2108587254	.0007204874
.01400	.1863766895	.1870649329	.0006882433
.01500	.1653022329	.1659563171	.0006540842
.01600	.1466110448	.1472298825	.0006188377
.01700	.1300336095	.1306168235	.0005832140
.01800	.1153309380	.1158786715	.0005477335
.01900	.1022910504	.1028038688	.0005128184
.02000	.0907259379	.0912047298	.0004787919
.02100	.0804688525	.0809147512	.0004458987
.02200	.0713719026	.0717862188	.0004143162
.02300	.0633039209	.0636880875	.0003841667
.02400	.0561485729	.0565041010	.0003555281
.02500	.0498026800	.0501311231	.0003284431
.02600	.0441747317	.0444776569	.0003029252
.02700	.0391835661	.0394625308	.0002789647
.02800	.0347571996	.0350137341	.0002565345
.02900	.0308317894	.0310673825	.0002355932
.03000	.0273507124	.0275668018	.0002160894

.03100	.0242637495	.0244617140	.0001979645
.03200	.0215263654	.0217075149	.0001811495
.03300	.0190990517	.0192646349	.0001655832
.03400	.0169467735	.0170979700	.0001511965
.03500	.0150384582	.0151763785	.0001379203
.03600	.0133465479	.0134722337	.0001256858
.03700	.0118466014	.0119610267	.0001144254
.03800	.0105169418	.0106210154	.0001040736
.03900	.0093383444	.0094329117	.0000945673
.04000	.0082937589	.0083796050	.0000858461
.04100	.0073680638	.0074459159	.0000778521
.04200	.0065478483	.0066183792	.0000705309
.04300	.0058212184	.0058850493	.0000638308
.04400	.0051776262	.0052353298	.0000577036
.04500	.0046077170	.0046598207	.0000521037
.04600	.0041031945	.0041501835	.0000469890
.04700	.0036567015	.0036990214	.0000423199
.04800	.0032617131	.0032997732	.0000380601
.04900	.0029124433	.0029466190	.0000341758

Table 8c

Results of three stage scheme for Equation 6

h = .00100

x	Y(Xn)	Yn	en
.00100	.8869204327	.8869273954	.0000069627
.00200	.7866278601	.7961960775	.0095682174
.00300	.6976763509	.7217770860	.0241007351
.00400	.6187834417	.6596317222	.0408482805
.00500	.5488117378	.6069528739	.0581411361
.00600	.4867524417	.5617288253	.0749763835
.00700	.4317108311	.5224801530	.0907693219
.00800	.3828933805	.4880946169	.1052012364
.00900	.3395962325	.4577198858	.1181236533
.01000	.3011951863	.4306918180	.1294966317
.01100	.2671366050	.4064852329	.1393486279
.01200	.2369294572	.3846793758	.1477499186
.01300	.2101382380	.3649331947	.1547949567
.01400	.1863766895	.3469673157	.1605906262
.01500	.1653022329	.3305507096	.1652484767
.01600	.1466110448	.3154906660	.1688796213
.01700	.1300336095	.3016251627	.1715915533
.01800	.1153309380	.2888169950	.1734860570
.01900	.1022910504	.2769491950	.1746581446
.02000	.0907259379	.2659214381	.1751955002
.02100	.0804688525	.2556471820	.1751783295
.02200	.0713719026	.2460513785	.1746794759
.02300	.0633039209	.2370686278	.1737647069
.02400	.0561485729	.2286416749	.1724931020
.02500	.0498026800	.2207201814	.1709175014
.02600	.0441747317	.2132597117	.1690849800
.02700	.0391835661	.2062208912	.1670373251
.02800	.0347571996	.1995687071	.1648115075
.02900	.0308317894	.1932719209	.1624401315
.03000	.0273507124	.1873025714	.1599518590

.03100	.0242637495	.1816355585	.1573718090
.03200	.0215263654	.1762482855	.1547219200
.03300	.0190990517	.1711203530	.1520213013
.03400	.0169467735	.1662333004	.1492865270
.03500	.0150384582	.1615703805	.1465319223
.03600	.0133465479	.1571163650	.1437698171
.03700	.0118466014	.1528573760	.1410107746
.03800	.0105169418	.1487807400	.1382637982
.03900	.0093383444	.1448748596	.1355365152
.04000	.0082937589	.1411291020	.1328353431
.04100	.0073680638	.1375337020	.1301656381
.04200	.0065478483	.1340796731	.1275318249
.04300	.0058212184	.1307587328	.1249375144
.04400	.0051776262	.1275632353	.1223856090
.04500	.0046077170	.1244861109	.1198783940
.04600	.0041031945	.1215208129	.1174176184
.04700	.0036567015	.1186612705	.1150045690
.04800	.0032617131	.1159018461	.1126401330
.04900	.0029124433	.1132372972	.1103248539

CHAPTER SIX
GENERAL CONCLUSION

6.1 SUMMARY

In this thesis, we have developed a k^{th} - order inverse polynomial method for solving ordinary differential equations with singularities.

This is motivated by the rational approximation technique suggested by Fatunla (1982a). It was analysed, computerised and implemented with sample problems on a micro computer.

The results show that the schemes are capable of solving ordinary differential equations with poles or low order discontinuous differentials.

6.2 LIMITATIONS

Since the method is based on Taylor and Binomial series expansions, it is subject to point to point error and possible error propagation. Besides these, the followings are possible problems:

(i) There is difficulty in the selection of starting stepsize, that is, the stepsize that will over step the points of singularities.

(i) Step size restriction by accuracy in the neighbourhood of singularity.

(ii) Step size restriction dictated by stability

(iii) Complexicity of the determination of the k parameters of the methods which increases with the stage k of the scheme.

6.3. Recommendation

Based on the limitations, we need to adopt some strategies which can help to find appropriate balance between the step size (h), the order of accuracy, and stability of the methods in order to achieve a better, more accurate, efficient and effective scheme.

These strategies include choice of step size (h) as to satisfy h = $\frac{y_n}{y'_n} \neq 0$ and the use of double precision arithmetic to minimise the effects of round off error.

As a result of the high accuracy of these schemes, we are recommending that a higher order stage k schemes be exploited as a general purpose code for solution of ordinary differential equations with singularities.

6.4 CONTRIBUTION TO KNOWLEDGE

The new schemes will be adequate for solution of differential equation of Electrical networks, and model equation of economy affected by inflation. Other areas of application include control problem and computer aided designs from which ordinary differential equations with singularity can arise. By the stability properties of the schemes, it will be capable of solving stiff and non stiff oscillating differential equations.

REFERENCES

1. Ademiluyi, R.A (1985): "Polynomial representation of hybrid methods for numerical solution of stiff ODEs" A paper presented at the 9th annual conference of mathematical society of Nigeria, University of Ife, Ile- Ife.
2. Ademiluyi, R.A (1987): "New hybrid methods for systems of stiff ordinary differential equations" Ph.D Dissertation, University of Bnin, 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 applications, held at the University of Benin, Benin City.
4. Ademiluyi, R.A (1992): "Finite difference approximation of differential equations as a case study of scientific computating" International conference on computational mathematics held at the University of Benin, Benin City.
5. Alkinson K.E(1978): "An introduction to Numerical Analysis" John Wiley & Sons
6. Birkoff, G and Virga, R.S (1965): "Discretisation Errors for well-set Cauchy problems" International Journals of mathematics and physics, vol. 44, pp. 1-23
7. Bo, H-L Yans, W-H and Xin, A-Y (1983): "An Effective method for a system of QDE with RHS functions containing Discontinuities". Proceedings with IMACS World congress on system simulation and scientific computation, pp. 6-9
8. Carver M.B(1978): "Efficient integration over Discontinuities in ODE simulations" Math. Comp. simulation 20, pp. 190-196.

9. Dahlquist G.(1956): "Convergence and stability in the Numerical Integration of ΘDES" Math, Scand 4. pp. 33 – 53.

10. Dahliquist G.(1959): "Stability and Error Bounds in the Numerical Integration of ODES", Transcript 130, Royal Institute of Technology, Stockholm.

11. Dahlquist G.(1963): "A special stability problem for linear multistep methods" BIT 3, pp. 27-43

12. Ellison D.(1981): "Efficient Automatic Integration of ODES with discontinuities" Math. Comp. Sm. 23, pp. 12-20.

13. Evans, D.J and Fatunla S.O (1975): "Accurate Numerical determination of the intersection point of the solution of a DE with given algebraic relation" , J. IMA 16, pp. 355-359.

14. Fatunla S.O.(1978): "A variable order one step scheme for Numerical solution of ODES" Comp. Math with applications 4, pp. 33 – 41.

15. Fatunla S.O (1982): "Numerical Treatment of Special IVPS", Proceedings BAIL II conference (J.J. H . Miller, C.C), pp. 28 – 45. Trinity college, Dublin

16. Fatunla S.O (1985): "Numerical Treatment of Singular IVPS" Computer and math. With Aplli.

17. Fatunla S.O.(1986): "Applied Numerical Methods for Initial value problems in ODES", Academic press, San Diego.

18. Gear, C.W (1971): "Algorithm 407, DIFUB for solution of ordinary differential equations" communication of ACM, vol. 14, pp. 185-190

19. Gear, C.W (1980): " Runge-Kutta starters for Multistep methods", ACM TOMS 6, pp. 263-279

20. Gear, C.W and Qsterby, O. (1984): "Solving ODES with Discontinuities. "ACM TUMS 10, pp. 23 – 44.

21. Hay, J.L , Gosbie, R.E and Chaplin, R.I (1974): " Integration routines for systems with discontinuities", Computer J. 17, pp. 275-278.

22. Hindmarsh, A.C (1974): "GEAR : ODE system solvers", revision 3, report UCID-30001, Lawrence, Livermore laboratory, University of California, Livermore.

23. Hindmarsh, A.C (1983): "Toward a systematized collection of ODE solvers", World congress on systems simlation and scientific computation, pp. 427-429

24. Iserles A.(1981): "Rational Interpolation to exp(-x) with application to certain stiff systems." SIAM J. Numer Anal. 18, pp. 1-12."

25. Lambert: J.D " Computational methods in ordinary Differential Equations", John Wiley, New York (1973).

26. Lambert J.D and Shaw B (1965): "On the Numerical solution of $y' = f(x,y)$ by a class of formulae based on Rational approximations" Math. Comp. 19, pp. 456 – 462.

27. Lambert J.D and Shaw B (1966): "A method for the numerical solution of $y' = f(x,y)$ based on a self- adjusting non polynomial-interpolant", Math. Comp. 20, pp. 11-20

28. Liniger, W and Willonghby, R.A (1967): "Efficient Numerical integration methods for stif systems of differential equations", IBM research report RC-170

29. Luke, Y.L Fair, W and Wimp J. (1975): "Predicator Corrector Formulae based on Rational Interpolants" J. Computers and maths. With applications 1, pp. 3-12.

30. Widlund, O. (1967): "A note on unconditional stable LMM", BIT 7, pp. 65-70.

APPENDIX 1

```
C       THIS PROGRAM SOLVES ORDINARY DIFFERENTIAL
C       EQUATIONS WITH SINGULARITIES
C       Initialise x , y, h, Tol, X1
        OPEN (UNIT=2, FILE = ' MATHS.OUT', STATUS = 'NEW'
        OPEN (UNIT=1, FILE= ' maths.dat',STATUS = 'NEW'
1       FORMAT (12X',RESULTS OF ONE STAGE SCHEME')
        WRITE (3,100)
100     FORMAT ('/',..............................................'./')
        WRITE (2,6)
6       FORMAT (7X,'X', 19X, 'Y(Xn)', 20X, 'Yn' 19X,' ℓn ')
        WRITE (3,101)
101     FORMAT ('/',...........................................'/')
       HI = 0.01
8       HMIN = 0.01D-5
        TOL =1.0D-6
```

```
      $X_1$=1.0D0
      $X_0$=0.0D0
      $Y_0$=1.0D0
C     INVOKE SUBROUTINE IMPRRK TO COMPUTE APPROXIMATE
C     VALUE OF Y
      CALL IMPRRK (X, Y, HI, TOL, $Y_{n+1}$ , $\ell_n$ )
C     CHANGE THE STEP SIZE TO HALF OF THE STEP SIZE
      FOR I =2,N ; $X_i$  = $X_{i-1}$ + $h_{old}$
      HN = 0.5D0*HI
C     WRITE OUT RESULTS
      WRITE (2,10) HI, Y(X), $Y_{n+1}$ , $\ell_n$
10    FORMAT (f10.5,2x,3(f20.10,2x))
      WRITE (2,99)
99    FORMAT ('/', 18('-'), '/',5(20('-'),'/'))
      IF (HI.LE.HMIN) THEN
      STOP
```

```
      ELSE
      HI=HN
      GOTO 8
      END IF
      END
C     FUNCTION SUBPROGRAM TO COMPUTE THE FUNCTION OF F
      FUNCTION F(x,y)
      REAL*8 YI, Y, Perror
      F= (Yi-1 **2)/(Yi-1  – H* Y'i-1)
      Perror =ABS((Yn – Y))
      RETURN
      END
C     FUNCTION SUBPROGRAM TO COMPUTE EXACT
      REAL *8  Yn, Y, Perror
      YEXACT = tan(Xi  + pi/4)
      RETURN
```

```
      END

C     SUBROUTINE IMPRRK TO COMPUTE THE APPROXIMATE

C     VALUE OF Y, USING EULER METHOD

      SUBROUTINE, (X,Y,H, Yn, $\ell_n$)

      REAL *8 (Yn, Y, Perror)

      $Y_{n+1} = Y_n + H* Y'_n$
```

APPENDIX 2

```
C       THIS PROGRAM SOLVES ORDINARY DIFFERENTIAL EQUATIONS

C       WITH SINGULARITIES USING TWO STAGE SCHEME

C       USING ERROR ESTIMATES TO ADJUST THE STEP SIZE. THE STRATEGY

C       IS TO CONTROL THE STEP SIZE TO GET MINIMUM ERROR

        REAL*8 (A-H, T-Z)

        OPEN (UNIT=2, FILE= ‘maths2.out’, STATUS =’NEW’)

        WRITE (8,*) ‘ RESULTS OF TWO STAGE SCHEME, ADOPTING
        RICHARDSON EXTRAPOLATION METHOD TO ESTIMATE THE

        LOCAL TRUNCATION ERROR AND CONTROL THE STEP SIZE

        WRITE (8,100)

100     FORMAT (‘/’,..........................................................,’/’)

        WRITE(8,11)

11      FORMAT (2X, ‘STEP SIZE’ , VALUE OF X, YEXACT, NUMERICAL
        VALUE, LOCAL ERROR, ERROR

        WRITE (8,120)

   120     FORMAT(‘/’......................................................’/’)
```

```
       X0 =0.0D0
       Y0 =1.0D0
       HI=0.01D-1
       TOL=0.1D-2
       X1 =1.0D0
2      Xp = X0 + HI
       CALL ADAPT (X,Y,TOL, HN, L.T.E, Yn+1 )
       CALL IMPRRK (X0 , Y0 , HI, TOL, Y n)
       Yn = YEXACT (Xp, Ynp)
       E = ABS(Yn+1 – Ynp)
       WRITE (8,2) , HN, Yn, Yn+1 , L.T.E, E)
2      FORMAT(2X, 6(f10.5,3X))
       WRITE (8,101)
101    FORMAT ('/', 6(20('-'), '/'))
       IF (Xp. GE. X1 ) STOP
       X0 = Xp
       Y0= Ynp
```

```
      GOTO 2

      END

C     SUBROUTINE ADAPT TO CONTROL THE STEP SIZE

      SUBROUTINE ADAPT (X, Y, TOL, HN, L.T.E)

      REAL *8 (A-H, T-Z)

      H= 0.01D-1

      Dn=TOL*Ynp

      H2 = 0.01D0*H

      Xp = X + H2

      CALL IMPRRK (X,Y,H,L.T.E, Ynp1)

      CALL IMPRRK (X,Y,H2,L.T.E,Ynp2)

      CALL IMPRRK (Xp1, Yp2, H2, L.T.E, Ynp3)

      SET L.T.E = (2^P /1-2^P )(Yn-Y_L )

      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 = ABS(DNP**0.25D0)
      END IF
      IF (L.T.E. LE . E) THEN
      GOTO 8
      ELSE
      H=HN
      GOTO 7
7     RETURN
      END
C     SUBROUTINE IMPRRK TO COMPUTE NUMERICAL VALUES OF Y
      SUBROUTINE IMPRRK (X, Y, H, L.T.E, Ynp)
C     INITIALISE THE VARIABLES
      AH 1= 0.0D0
      AH 2=0.0D0
C     FUNCTION SUBPROGRAM TO COMPUTE F
```

```
FUNCTION F (x,y)

REAL* 8 (A-H,T-Z)

SET

$Y_{n+2}$ = Y**3/(Y**2-H*Y* Y' + (H**2/2)*(2*( Y' )**2- Y * Y" )

Perror = ABS($Y_{N+1}$- Y(X))

RETURN

END
```