Journal of the Nigerian Association of Mathematical Physics
Volume33,(January, 2016), pp 61 – 70
© J. of NAMP

On the Rooted Tree and Component Analysis of an Explicit Fourth-Stage Fourth-Order Runge-Kutta Method

[1]Agbeboh Goddy Ujagbe,[2]Esekhaigbe Aigbedion Christopher and [3]Akhanolu Aziegbe Gilbert

[1]Department of Mathematics, Ambrose Alli University, P.M.B 14, Ekpoma, Edo State, Nigeria.
[2]Department of Statistics, Auchi Polytechnic, P.M.B 13, Auchi, Edo State, Nigeria.
[3]Department of Mathematics, Ambrose Alli University, P.M.B 14, Ekpoma, Edo State, Nigeria.

Abstract

The object of this work is to concentrate on the f(x,y) functional derivatives after using Taylor Series expansion to expand the fourth-stage fourth-order explicit Runge-Kutta method. Efforts will be made to vary the parameters with the aim of getting a fourth order formula that can improve results when implemented on initial-value problems. Efforts will also be made to represent the derived equations and their individual f(x,y) functional derivatives on Butcher's rooted trees. This idea is derivable from general graphs and combinatorics.

Keywords: Rooted tree diagram, Components, explicit, f(x,y) functional derivatives, Runge-Kutta Methods, Linear and non- linear equations, Taylor series, Graphs, Parameters, Initial-value Problems, Combinatorics

1.0 Introduction

The essence of this paper is toconcentrate on the f(x,y) functional derivatives after using Taylor Series to expand an explicit fourth- stage fourth- order Runge–Kutta method.The aim is to derive a fourth-stage fourth – order Runge-kutta formula that can improve performance. It also involvesrepresenting the derived equations and their individual f(x,y) functional derivatives on Butcher's rooted trees. Scientific implementation of the formula on initial-value problems in ordinary differential equations of the form:

$y^1 = f(x, y), y(x_0) = y_0,\ a \le x \le b,$ h given, with a view to finding out its consistency and accuracy, is also carried out.

Explicit Runge – kutta (ERK) formulas are among the oldest and best – understood schemes in the numerical analysis tool kit. However, according to [1]; "despite the evolutions of a vast and comprehensive body of knowledge, ERK algorithms continue to be sources of active research". The history of ERK methods began almost a century ago. Classic references are [2],[3] and [4]. According to [5], "the Runge –Kutta methods represent an important family of implicit and explicit iterative methods for approximation of ordinary differential equations in numerical analysis".

Because of their elegance and simplicity, ERK methods are usually among the first to be taught in the ODE section of a numerical methods course. Thankfully, good quality introductory texts no longer dismiss "the Runge – kutta method" as a fixed step size implementation of the classic 4^{th} order ERK formula. However, significant advancements in the state – of – the – art which post – date the work of [6], even in the fundamental area of deriving ERK formulas, tend to be ignored. According to [7], another side effect of the simple nature of the ERK formula is that a generation of non – experts have been tempted to write their own "quick and dirty" codes. It is widely acknowledged that "squeaky – clean" codes require a great deal of expertise and programming effort. A high – level discussion of some of the issues involved in ERK is found in [8]. Recent work on Runge-Kutta analysis can be found in [9 - 13]. More recent works are in [14 - 16].

The work of Butcher in [17,18,19,20,21] revealed much successes in the analysis of explicit Runge-Kutta methods and their transformation to rooted tree diagrams. This was because the continuation of the process of Taylor Series gives rise to very complicated formulas. It was therefore, advantages to use a graphical representation for a convenient analysis of the order of a Runge – kutta method; hence, the basic tree theory was introduced. A tree is a rooted graph which contains no circuits.

Corresponding author: Agbeboh Goddy Ujagbe, E-mail:ujagbegoddy@yahoo.com, Tel.: +2348036587219

The symbol T is used to represent the tree with only one vertex. All rooted trees can be represented using T and the operation [$t_1, t_2, \ldots, t_m$].
Hence, it is the differentials and equations derived that are represented on trees so as to enable us compare the order condition with their differentials for varying parameters.
References [22,23] pointed out that the Butcher group is the group of characters that had risen independently in their own work on rooted trees analysis in Runge – kutta methods for solving initial-value problems in ordinary differential equations.
Recent works on rooted tree analysis are found in [24 - 28].
Conclusively, despite the fact that good, reliable explicit Runge-Kutta formulas exist, there is still need for their transformation to rooted tree diagrams. Traditionally, Runge – kutta methods are all explicit, although recently, implicit Runge – kutta methods, which have improved weak stability characteristics have been considered. However, the transformation of implicit Runge-Kutta methods to rooted tree diagrams can also be explored.

2.0 Methods of Derivation

i. From the general Runge-Kutta method, get a Fourth Stage-Fourth order method
ii. Obtain the Taylor series expansion of $k_{i's}^{'}$ about the point (x_n, y_n), i=2,3,4,
iii. Carry out substitution to ensure that all the $k_{i's}^{'}$ are in terms of k_1 only.
iv. Insert the $k_{i's}^{'}$ in terms of k_1 only into $b_1k_1 + b_2k_2 + b_3k_3 + b_4k_4$
v. Separate all f(y) functional derivatives with their coefficients from all f(x,y) functional derivatives and their coefficients.
vi. Discard all f(y) functional derivatives and their coefficients.
vii. Equate the coefficients of all f(x,y) functional derivatives with the coefficients Taylor series expansion involving only f(x,y) functional derivatives of the form:

$$\phi(x,y,h) = f + \frac{h}{2!}f_x + \frac{h^2}{3!}\left(f_{xx} + 2ff_{xy} + f_xf_y\right) + \frac{h^3}{4!}(f_{xxx} + 3ff_{xxy} + 3f^2f_{xyy} + 3f_xf_{xy} + 5ff_yf_{xy} + 3ff_xf_{yy} + f_{xx}f_y + f_xf_y^2) \quad (1)$$

As a result, a set of linear/non-linear equations will be generated. Represent those equations and their f(x, y) functional derivatives on Butcher's rooted tree.
viii. Vary the set of equations to derive a new fourth-stage fourth-order explicit Runge-Kutta formula.

3.0 Derivation of The Fourth – Order Fourth Stage ERK Method

According to Lambert (1991), the general R – Stage Runge – Kutta method is:

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

$$\phi(x_n, y_n, h) = \sum_{r=1}^{R} b_r k_r$$

$$k_1 = f(x,y)$$

$$k_r = f(x + hc_r, y + h\sum_{s=1}^{r-1} a_{rs}k_s), r = 2, 3, \ldots R \quad (2)$$

The formula is defined by the number of stages s, the nodes$[c_r]^s{}_{r=1}$, the internal weights $[a_{rs}]^{r-1,s}{}_{s=1,r=2}$ and the external weights$[b_r]^s{}_{r=1}$.
From the above scheme, the fourth stage fourth – order method is:

$$y_{n+1} = y_n + h(b_1k_1 + b_2k_2 + b_3k_3 + b_4k_4)$$
$$k_1 = f(x_n, y_n)$$
$$k_2 = f(x_n + c_2, y_n + ha_{21}k_1)$$
$$k_3 = f(x_n + c_3h, y_n + h(a_{31}k_1 + a_{32}k_2))$$
$$k_4 = f(x_n + c_4h, y_n + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)) \quad (3)$$

Using Taylor's series expansion for $k_i^{'}s$, we have:

$$k_1 = f(x_n, y_n)$$
$$k_2 = \sum_{r=o}^{\infty}\frac{1}{r!}(c_2h\frac{d}{dx} + ha_{21}k_1\frac{d}{dy})^r f(x_n, y_n)$$
$$k_3 = \sum_{r=o}^{\infty}\frac{1}{r!}(c_3h\frac{d}{dx} + h(a_{31}k_1 + a_{32}k_2)\frac{d}{dy})^r f(x_n, y_n)$$
$$k_4 = \sum_{r=o}^{\infty}\frac{1}{r!}(c_4h\frac{d}{dx} + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)\frac{d}{dy})^r f(x_n, y_n) \quad (4)$$

Hence, we have:

$k_1 = f$

$k_2 = f + \left(c_2hf_x + ha_{21}k_1f_y\right) + \frac{1}{2!}\left(c_2hf_x + ha_{21}k_1f_y\right)^2$
$+ \frac{1}{3!}\left(c_2hf_x + ha_{21}k_1f_y\right)^3 + \frac{1}{4!}\left(c_2hf_x + ha_{21}k_1f_y\right)^4 + 0(h^5)$

$k_3 = f + \left(c_3hf_x + h(a_{31}k_1 + a_{32}k_2)\, f_y\right) + \frac{1}{2!}\left(c_3hf_x + h(a_{31}k_1 + a_{32}k_2)\, f_y\right)^2$
$+ \frac{1}{3!}\left(c_3hf_x + h(a_{31}k_1 + a_{32}k_2)\, f_y\right)^3 + \frac{1}{4!}\left(c_3hf_x + h(a_{31}k_1 + a_{32}k_2)\, f_y\right)^4 + 0(h^5)$

$k_4 = f + \left(c_4hf_x + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)\, f_y\right) + \frac{1}{2!}\left(c_4hf_x + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)\, f_y\right)^2$
$+ \frac{1}{3!}(c_4hf_x + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)\, f_y)^3$
$+ \frac{1}{4!}\left(c_4hf_x + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)\, f_y\right)^4 + 0(h^5)$ (5)

Expanding fully and substituting the various k_i' s, i = 2, 3, 4 into their various positions in terms of k_1 only and collecting like terms, in terms of f(x, y) functional derivatives only, and discarding everything that has to do with f(y) functional derivatives, we have:

$k_1 = f$

$k_2 = f + hc_2f_x + \frac{h^2}{2!}c_2^2f_{xx} + h^2c_2a_{21}ff_{xy}\ \frac{h^3}{3!}c_2^3f_{xxx} +$
$\frac{h^3}{2!}c_2^2a_{21}ff_{xxy} + \frac{h^3}{2!}c_2a_{21}^2f^2f_{xyy} + \frac{h^4}{4!}c_2^4f_{xxxx} +$
$\frac{h^4}{3!}c_2^3a_{21}ff_{xxxy}\ \frac{h^4}{2!\,2!}c_2^2a_{21}^2f^2f_{xxyy} + \frac{h^4}{3!}c_2a_{21}^3f^3f_{xyyy} + 0\,(h^5)$

$k_3 = f + hc_3f_x$
$+ \frac{h^2}{2!}c_3^2f_{xx} + h^2c_3(a_{31} + a_{32})ff_{xy} + h^2c_2a_{32}f_xf_y + \frac{h^3}{3!}c_3^3f_{xxx}$
$+ \frac{h^3}{2!}c_3^2(a_{31} + a_{32})ff_{xxy} + \frac{h^3}{2!}c_3(a_{31}^2 + 2a_{31}a_{32} + a_{32}^2)f^2f_{xyy}$
$+ h^3c_2a_{32}(a_{31} + a_{32})ff_xf_{yy} + h^3a_{21}a_{32}(c_2 + c_3)ff_yf_{xy} + \frac{h^3}{2!}c_2^2a_{32}f_yf_{xx}$
$+ h^3c_2c_3a_{32}f_xf_{xy} + \frac{h^4}{4!}c_3^4f_{xxxx} + \frac{h^4}{3!}c_2^3a_{32}f_{xxx}f_y + \frac{h^4}{2!}c_3^2c_2a_{32}f_xf_{xxy}$
$+ \frac{h^4}{2!}a_{21}a_{32}(c_2^2 + c_3^2)ff_yf_{xxy} + \frac{h^4}{3!}a_{21}a_{32}(2c_2a_{31} + 3c_2a_{21} + 6c_3a_{31})f^2f_yf_{xyy}$
$+ \frac{h^4}{2!}c_3a_{32}c_2^2f_{xx}f_{xy} + \frac{h^4}{2!}c_2^2a_{32}(a_{31} + a_{32})ff_{xx}f_{yy} +$
$\frac{h^4}{2!}a_{21}a_{32}(2c_2a_{31} + 2c_2a_{32} + c_3a_{21})f^2f_{xy}f_{yy} + h^4c_3a_{32}c_2a_{21}ff_{xy}^2 + \frac{h^4}{2!}a_{32}^2c_2^2f_x^2f_{yy}$
$+h^4a_{32}^2a_{21}c_2ff_xf_yf_{yy} + \frac{h^4}{2!}c_3c_2a_{32}(6a_{31} + 2a_{32})ff_xf_{xyy}$
$+ \frac{h^4}{2!}c_2a_{32}(a_{31}^2 + 2a_{31}a_{32} + a_{32}^2)f^2f_xf_{yyy} + \frac{h^4}{3!}c_3^3(a_{31} + a_{32})ff_{xxxy}$
$+ \frac{h^4}{2!\,2!}c_3^2(a_{31}^2 + 2a_{31}a_{32} + a_{32}^2)f^2f_{xxyy} + \frac{h^4}{3!}c_3(a_{31}^3 + 3a_{31}^2a_{32} + 3a_{31}a_{32}^2 + a_{32}^3)f^3f_{xyyy} + 0\,(h^5)$

$k_4 = f ++ \mathrm{h}c_4f_x + h^2(c_4a_{42} + c_3a_{43})f_xf_y +$
$\frac{h^2}{2!}c_4^2f_{xx} + h^2c_4(a_{41} + a_{42} + a_{43})ff_{xy} + \frac{h^3}{2!}\ (c_2^2a_{42} + c_3^2a_{43})f_{xx}f_y + h^2(c_4a_{21}a_{42} +$
$c_3a_{31}a_{43} + c_3a_{32}a_{43}))ff_{xy}f_y + h^3c_2a_{32}a_{43}f_xf_y^2 + h^3(c_2c_4a_{42} + c_3c_4a_{43})\ f_xf_{xy}$
$+h^3c_2a_{32}a_{43}f_xf_y^2 + h^3(c_2c_4a_{42} + c_3c_4a_{43})\ f_xf_{xy} + h^3(c_2a_{21}a_{42} + c_4a_{31}a_{43} +$
$c_4a_{32}a_{43})ff_yf_{xy} + h^3(c_2a_{41}a_{42} + c_3a_{41}a_{43} + c_3a_{42}a_{43} + c_2a_{42}a_{43} + c_2a_{42}^2 +$

$c_3a_{43}^2)ff_xf_{yy} + \frac{h^3}{3!}c_4^3f_{xxx} + \frac{h^3}{2!}(c_4^2a_{41} + c_4^2a_{42} + c_4^2a_{43})ff_{xxy} + \frac{h^3}{2!}c_4(a_{41}^2 + 2a_{41}a_{42} +$
$2a_{41}a_{43} + a_{42}^2 + 2a_{42}a_{43} + a_{43}^2)f^2f_{xyy} + \frac{h^4}{3!}(c_2^3a_{42} +$
$c_3^3a_{43})f_{xxx}f_y + \frac{4}{3!}(3c_2^2a_{21}a_{42} + c_3^2a_{31}a_{43} + 3c_3^2a_{32}a_{43} + 3c_4^2a_{21}a_{42} + 3c_4^2a_{31}a_{43} +$
$3c_4^2a_{32}a_{43})ff_{xxy}f_y + \frac{h^4}{2!}(c_2a_{21}^2a_{42} + 2c_3a_{31}a_{32}a_{43} + c_3a_{31}^2a_{43} + c_3a_{32}^2a_{43} +$
$2c_4a_{21}a_{41}a_{42} + 2c_4a_{31}a_{41}a_{43} + 2c_4a_{32}a_{41}a_{43} + 2c_4a_{21}a_{42}^2 + 2c_4a_{31}a_{42}a_{43} +$
$2c_4a_{32}a_{42}a_{43} + 2c_4a_{21}a_{42}a_{43} + 2c_4a_{31}a_{43}^2 + 2c_4a_{32}a_{43}^2)f^2f_yf_{xyy} +$
$\frac{h^4}{3!}(c_2^2a_{32}a_{43})f_{xx}f_y^2 + h^4(c_2a_{21}a_{32}a_{43} + c_3a_{21}a_{32}a_{43} + c_4a_{21}a_{32}a_{43})ff_{xy}f_y^2 +$
$\frac{h^4}{2!}(2c_2a_{31}a_{32}a_{43} + 2c_2a_{32}^2a_{43} + 2c_2a_{32}a_{41}a_{43} + 2c_2a_{32}a_{42}a_{43} + 2c_2a_{31}a_{42}a_{43} +$
$2c_2a_{32}a_{42}a_{43} + 2c_3a_{21}a_{42}a_{43} + 2c_2a_{21}a_{42}^2 + c_2a_{32}a_{43}^2 + c_3a_{31}a_{43}^2 + c_3a_{32}a_{43}^2 +$
$c_3a_{31}a_{43}^2 + c_3a_{32}a_{43}^2)ff_xf_yf_{yy} + h^4(c_2c_3a_{32}a_{43} + c_2c_4a_{32}a_{43})f_xf_yf_{xy} +$
$\frac{h^4}{2!}(c_2^2c_4a_{42} + c_3^2c_4a_{43})f_{xx}f_{xy} + h^4(c_2c_4a_{21}a_{42} + c_3c_4a_{31}a_{43} + c_3c_4a_{32}a_{43})$
$ff_{xy}^2 + \frac{h^4}{2!}(c_4a_{21}^2a_{42} + c_4a_{31}^2a_{43} + 2c_4a_{31}a_{32}a_{43} + c_4a_{32}^2a_{43} + 2c_2a_{21}a_{41}a_{42} +$
$2c_3a_{31}a_{41}a_{43} + 2c_3a_{32}a_{41}a_{43} + 2c_3a_{31}a_{42}a_{43} + 2c_3a_{32}a_{42}a_{43} + c_2a_{21}a_{42}^2 + c_3a_{31}a_{43}^2 +$
$c_3a_{32}a_{43}^2)\,f^2f_{xy}f_{yy} + \frac{h^4}{2!}(2c_2c_3a_{42}a_{43} + c_2^2a_{42}^2 + c_3^2a_{43}^2)f_x^2f_{yy} + \frac{h^4}{2!}(c_2c_4^2a_{42} +$
$c_3c_4^2a_{43})f_xf_{xxy} + h^4(c_2c_4a_{41}a_{42} + c_3c_4a_{41}a_{43} + c_2c_4a_{42}^2 + c_3c_4a_{42}a_{43}) +$
$c_2c_4a_{42}a_{43} + c_3c_4a_{43}^2)ff_xf_{xyy} + \frac{h^4}{2!}(c_2c_{41}^2a_{42} + c_3c_{41}^2a_{43} + 2c_2a_{41}a_{42}^2 + 2c_3a_{41}a_{42}a_{43} +$
$2c_2a_{41}a_{42}a_{43} + c_3c_{42}^2a_{43} + 2c_2c_{42}^2a_{43} + 2c_3a_{41}a_{43}^2 + 2c_3a_{42}a_{43}^2 + c_2a_{42}a_{43}^2 + c_2c_{42}^3 +$
$c_3a_{43}^3)f^2f_xf_{yyy} + \frac{h^4}{4!}c_4^4f_{xxxx} + \frac{h^4}{3!}(c_4^3a_{41} + c_4^3a_{42} + c_4^3a_{43})ff_{xxxy} + \frac{h^4}{2!2!}c_4^2(a_{41}^2 +$
$2a_{41}a_{42} + 2a_{41}a_{43} + a_{42}^2 + 2a_{42}a_{43} + a_{43}^2)f^2f_{xxyy} + \frac{h^4}{3!}c_4(a_{41}^3 + 3a_{41}^2a_{42} + 3a_{41}^2a_{43} +$
$3a_{41}a_{42}^2 + 6a_{41}a_{42}a_{43} + 3a_{42}^2a_{43} + 3a_{42}a_{43}^2 + a_{42}^3 + a_{43}^3)\,f^3f_{xyyy} + \frac{h^4}{2!2!}(2c_2^2a_{41}a_{42} +$
$2c_3^2a_{41}a_{43} + 2c_3^2a_{42}a_{43} + c_2^2a_{42}^2 + c_3^2a_{43}^2)ff_{xx}f_{yy.} + 0(h^5).$ (6)

Putting the $k_{i's}^{;}$ (f(x, y) functional derivatives only) into $y_{n+1} = y_n + h(b_1k_1 + b_2k_2 + b_3k_3 + b_4k_4)$ where $\phi(x,y,h) = b_1k_1 + b_2k_2 + b_3k_3 + b_4k_4$ and equating coefficients with the Taylor series expansion, we have the equations below:

$b_1 + b_2 + b_3 + b_4 = 1$ (7)

$b_2c_2 + b_3c_3 + b_4c_4 = ½$ (8)

$b_2c_2^2 + b_3c_3^2 + b_4c_4^2 = 1/3$ (9)

$b_2c_2a_{21} + b_3c_3(a_{31} + a_{32}) + b_4c_4(a_{41} + a_{42} + a_{43}) = \frac{1}{3}$ (10)

$b_3c_2a_{32} + b_4(c_2a_{42} + c_3a_{43}) = \frac{1}{6}$ (11)

$b_2c_2^3 + b_3c_3^3 + b_4c_4^3 = \frac{1}{4}$ (12)

$b_2c_2^3a_{21} + b_3c_3^2(a_{31} + a_{32}) + b_4c_4^2(a_{41} + a_{42} + a_{43}) = \frac{1}{4}$ (13)

$b_2c_2a_{21}^2 + b_3c_3(a_{31}^2 + 2a_{21}a_{32} + a_{32}^2) + b_4c_4(a_{41}^2 + 2a_{41}a_{42} +$
$(a_{41}^2 + 2a_{41}a_{42} + 2a_{41}a_{43} + a_{42}^2 + 2a_{41}a_{43} + a_{43}^2) = \frac{1}{4}$ (14)

$b_3c_2a_{32}(a_{31} + a_{32}) + b_4(c_2a_{42}(a_{41} + a_{42} + a_{43}) + c_3a_{43}(a_{41} + a_{42} + a_{43})) = 1/8$ (15)

$b_3c_{21}a_{32}(c_2 + c_3) + b_4(c_2a_{21}a_{42} + c_3a_{43}(a_{31} + a_{43}) + c_4a_{21}a_{42}$
$+ c_4a_{43}(a_{31} + a_{32}) = \frac{5}{24}$ (16)

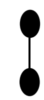
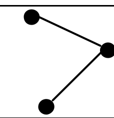
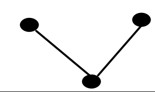
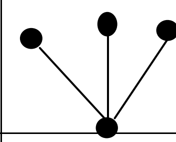
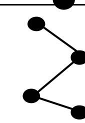
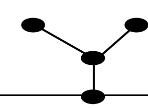
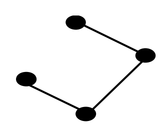
$b_3c_2^2a_{32} + b_4(c_2^2a_{42} + c_3^2a_{43}) = \frac{1}{12}$ (17)

$b_3c_2c_3a_{32} + b_4(c_2c_4a_{42} + c_3c_4a_{43}) = \frac{1}{8}$ (18)

$b_4c_2a_{32}a_{43} = \frac{1}{24}$ (19)

Below is Table 1 Showing the above thirteen equations (equations 7 to 19), their f(x,y)functional derivatives and rooted trees:

TABLE 1:

EQUATIONS	DERIVATIVES	R(T)	TREE	T	$\emptyset(t) = \frac{1}{r(t)}$
$b_1 + b_2 + b_3 + b_4 = 1$	f	1		t	$\sum_{i=1}^{4} b_i = 1$
$b_2c_2 + b_3c_3 + b_4c_4 = 1/2$	f_x	2		[t]	$\sum_{i=2}^{4} b_i\, c_i = 1/2$
$b_3a_{32}c_2 + b_4a_{42}c_2 + b_4a_{43}c_3 = 1/6$	f_xf_y	3		$[_2t]_2$	$\sum_{i=3,j=2}^{4,3} b_i\, a_{ij}c_j = 1/6$
$b_2c_2^2 + b_3c_3^2 + b_4c_4^2 = 1/3$	f_{xx}	3		$[t^2]$	$\sum_{i=2}^{4} b_i\, c_i^2 = 1/3$
$b_2c_2^2 + b_3c_3^2 + b_4c_4^2 = 1/3$	$f f_{xy}$				
$b_2c_2^3 + b_3c_3^3 + b_4c_4^3 = 1/4$	f_{xxx}	4		$[t^3]$	$\sum_{i=2}^{4} b_i\, c_i^3 = 1/4$
$b_2c_2^3 + b_3c_3^3 + b_4c_4^3 = 1/4$	$f f_{xxy}$				
$b_2c_2^3 + b_3c_3^3 + b_4c_4^3 = 1/4$	$f^2 f_{xyy}$				
$b_4a_{43}a_{32}c_2 = 1/24$	$f_xf_y^2$	4		$[3t]_3$	$\sum_{i=4,j=3,k=2}^{4,3,2} b_i\, a_{ij}a_{jk}c_k = 1/24$
$b_3a_{32}c_2^2 + b_4a_{42}c_2^2 + b_4a_{43}c_3^2 = 1/12$	$f_{xx}f_y$	4		$[_2t^2]_2$	$\sum_{i=3,j=2}^{4,3} b_i\, a_{ij}c_j^2 = 1/12$
$b_3a_{32}c_2^2 + b_4a_{42}c_2^2 + b_4a_{43}c_3^2 = 1/12$	$f f_y f_{xy}$				
$b_3a_{32}c_2c_3 + b_4a_{42}c_2c_4 + b_4a_{43}c_3c_4 = 1/8$	$f f_x f_{yy}$	4		{t[t]}	$\sum_{i=3,j=2}^{4,3} b_i\, c_i a_{ij}c_j = 1/8$
$b_3a_{32}c_2c_3 + b_4a_{42}c_2c_4 + b_4a_{43}c_3c_4 = 1/8$	$f_x f_{xy}$				
$b_3a_{32}c_2c_3 + b_4a_{42}c_2c_4 + b_4a_{43}c_3c_4 = 1/8$	$f f_y f_{xy}$				

Hence, set $c_1 = 0$, $c_4 = 1$, $c_2 = 1/4$, $c_3 = 3/4$
(8) becomes $b_2 + 3b_3 + 4b_4 = 2$ (20)
(9) becomes $3b_2 + 27b_3 + 48b_4 = 16$ (21)
(12) becomes $b_2 + 27b_3 + 48b_4 = 16$ (22)
Solving (7), (20), (21) and (22), we have:
$b_1 = 1/18$, $b_2 = 4/9$, $b_3 = 4/9$, $b_4 = 1/18$
From (10), we have: $2a_{21} + 6(a_{31} + a_{32}) + (a_{41} + a_{42} + a_{43}) = 6$ (23)
From (11), we have: $8a_{32} + a_{42} + 3a_{43} = 12$ (24)
From (13), we have: $a_{21} + 9(a_{31} + a_{32}) + 2(a_{41} + a_{42} + a_{43}) = 9$ (25)
From (14), we have: $4a_{21}^2 + 12(a_{31} + a_{32})^2 + 2(a_{41} + a_{42} + a_{43})^2 = 9$ (26)
From (15), we have: $8a_{21}(a_{31} + a_{32}) + a_{42}(a_{41} + a_{42} + a_{43}) + 3a_{42}(a_{41} + a_{42} + a_{43}) = 9$
(27)
From (16), we have: $32a_{31}a_{32} + 5a_{21}a_{42} + 7a_{43}(a_{31} + a_{32}) = 15$ (28)
From (17), we have: $8a_{32} + a_{42} + 9a_{43} = 24$ (29)
From (18), we have: $6a_{32} + a_{42} + 3a_{43} = 9$ (30)
From (19), we have: $a_{32}a_{43} = 3$ (31)
Solving (24), (29) and (30), we have:
$a_{32} = \frac{3}{2}$, $a_{42} = -6$, $a_{43} = 2$,
Let $A = a_{21}$, $B = a_{31} + a_{32}$, $D = a_{41} + a_{42} + a_{43}$
Hence, (23) becomes: $2A + 6B + D = 6$ (32)
(25) becomes: $A + 9B + 2D = 9$ (33)
(26) becomes: $4A^2 + 12B^2 + 2D^2 = 9$ (34)
(27) becomes: $12B = 9$ (35)
(28) becomes: $18A + 14B = 15$ (36)
From (35) $B = 3/4$ (37)
Hence, (32) becomes: $2A + D = 3/2$ (38)
(33) becomes: $A + 2D = 9/4$ (39)
(34) becomes: $4A^2 + 2D^2 = 9/4$ (40)
From (36), $A = 1/4$ (41)
Putting A into (38), (39) and (40), we get:
$D = 1$, $A = a_{21} = \frac{1}{4}$, $B = a_{31} + a_{32} = \frac{3}{4}, a_{31} = -\frac{3}{4}$, $a_{32} = \frac{3}{2}$,
Also, $D = a_{41} + a_{42} + a_{43} = 1$, but $a_{42} = -6$, $a_{43} = 2, a_{41} = 5$, (42)
The parameters put together are:
$C_1 = 0$, $c_2 = \frac{1}{4}$ $c_3 = \frac{3}{4}$, $c_4 = 1$, $b_1 = \frac{1}{18}$, $b_2 = \frac{4}{9}$, $b_3 = \frac{4}{9}$, $b_4 = \frac{4}{9}$
$a_{21} = \frac{1}{4}$, $a_{31} = -\frac{3}{4}$, $a_{32} = \frac{3}{2}$, $a_{41} = 5$, $a_{42} = -6$, $a_{43} = 2$, (43)
Putting (43) into (3), the fourth order method becomes:

$$y_{n+1} = y_n + \frac{h}{18}(k_1 + 8k_2 + 8k_3 + k_4)$$
$$k_1 = f(x_n, y_n)$$
$$k_2 = f(x_n + \frac{h}{4}, y_n + \frac{h}{4}k_1)$$
$$k_3 = f(x_n + \frac{3h}{4}, y_n + \frac{h}{4}(-3k_1 + 6k_2))$$
$$k_4 = f(x_n + h, y_n + h(5k_1 - 6k_2 + 2k_3)) \quad (44)$$

The Butcher's tableau for the parameters in (44) is:

0				
1/4	1/4			
3/4	- 3/4	3/2		
1	5	- 6	2	
	1/18	4/9	4/9	1/18

4.0 Implementation of the Formulas and Results

The formula is implemented on the initial – value problems below with the aid of FORTRAN programming language:

(i) $y^1 = -y, \ y(0) = 1, 0 \le x \le 1, \quad y(x_n) = \frac{1}{e^{xn}}$

(ii) $y^1 = y, \ y(0) = 1, 0 \le x \le 1, \quad y(x_n) = e^{xn}$

(iii) $y^1 = 1 + y^2, \ y(0) = 1, 0 \le x \le 1, \quad y(x_n) = \tan\left(x_n + {}^{\pi}/_{4}\right), h = 0.1$

TABLE 2 **TABLE OF RESULTS**

PROBLEM 1

XN	YN	TSOL	ERROR
.1D+00	0.9048375000000D+00	0.9048374180360D+00	-.8196404044369D-07
.2D+00	0.8187309014063D+00	0.8187307530780D+00	-.1483282683346D-06
.3D+00	0.7408184220012D+00	0.7408182206817D+00	-.2013194597694D-06
.4D+00	0.6703202889175D+00	0.6703200460356D+00	-.2428818514089D-06
.5D+00	0.6065309344234D+00	0.6065306597126D+00	-.2747107467060D-06
.6D+00	0.5488119343763D+00	0.5488116360940D+00	-.2982822888686D-06
.7D+00	0.4965856186712D+00	0.4965853037914D+00	-.3148798197183D-06
.8D+00	0.4493292897344D+00	0.4493289641172D+00	-.3256172068089D-06
.9D+00	0.4065699912001D+00	0.4065696597406D+00	-.3314594766990D-06
.1D+01	0.3678797744125D+00	0.3678794411714D+00	-.3332410563051D-06

PROBLEM 2

XN	YN	TSOL	ERROR
.1D+00	0.1105170833333D+01	0.1105170918076D+01	0.8474231405486D-07
.2D+00	0.1221402570851D+01	0.1221402758160D+01	0.1873094752636D-06
.3D+00	0.1349858497063D+01	0.1349858807576D+01	0.3105134649406D-06
.4D+00	0.1491824240081D+01	0.1491824697641D+01	0.4575605843105D-06
.5D+00	0.1648720638597D+01	0.1648721270700D+01	0.6321032899326D-06
.6D+00	0.1822117962092D+01	0.1822118800391D+01	0.8382985758892D-06
.7D+00	0.2013751626597D+01	0.2013752707470D+01	0.1080873699877D-05
.8D+00	0.2225539563292D+01	0.2225540928492D+01	0.1365200152481D-05
.9D+00	0.2459601413780D+01	0.2459603111157D+01	0.1697376878607D-05
.1D+01	0.2718279744135D+01	0.2718281828459D+01	0.2084323879270D-05

PROBLEM 3

XN	YN	TSOL	ERROR
.1D+00	0.1223051005569D+01	0.1223048880450D+01	-.2125119075158D-05
.2D+00	0.1508502732390D+01	0.1508497647121D+01	-.5085268468541D-05
.3D+00	0.1895771003842D+01	0.1895765122854D+01	-.5880987590245D-05
.4D+00	0.2464942965339D+01	0.2464962756723D+01	0.1979138375674D-04
.5D+00	0.3407951033347D+01	0.3408223442336D+01	0.2724089890727D-03
.6D+00	0.5328707710968D+01	0.5331855223459D+01	0.3147512490389D-02
.7D+00	0.1159500710295D+02	0.1168137380031D+02	0.8636669735614D-01
.8D+00	0.2841447010395D+03	-.6847966834558D+02	-.3526243693850D+03
.9D+00	0.8635045424394D+20	-.8687629546482D+01	-.8635045424394D+20
.1D+01	0.1640237043432+300	-.4588037824984D+01	-.1640237043432+300

5.0 Conclusion

After our implementation, it shows from the tables of numerical results that the method compared favourably well. It also revealed the fact that when the f(x,y) functional derivatives are considered alone it can generate a formula that can improve performance on results. As a result, this will make it less stressful in deriving a formula because the f(Y) functional derivatives were discarded for simplicity sake. Table 1 also revealed how the equations and their individual f(x,y) functional derivatives were represented on Butcher's rooted trees. Since $\sum_{i=1}^{4} b_i = 1$, this shows that the formula is consistent.

6.0 References

[1] Byrne, G. and Hindmarsh, A., (1990);" RK *methods prove popular at* IMA *conference on numerical* ODE_S",SIAM News, 23/2, pp. 14 – 15.

[2] Heun, K., (1900);"*Neue method zur approximativen integration der Differentailgleichungen einer un-abhangigen Veranderlichen, Zeitsch"*, Fur Math. U. Phys., 45, pp. 23 – 38.
[3] Kutta, W., (1901);" Beitrag zur naherungsweisen integration totaler Differentialgleichungen", Zeitschr. Fur Math. U. Phys., 46, pp. 435 – 453.
[4] Runge, C.,(1895);" Ueber die numerische Auflosung von Differentialgleichungen", Math. Ann., 46, pp. 167 – 178.
[5] Lambert, J., (1991);" Numerical Methods for Ordinary Differential Systems", Wiley publication.
[6] Fehlberg, E., (1970);" Klassiche Runge – Kutta – Formeln vierter und niedriger Ordnung mit Schrittweiten – Kontrolle und ihre Andwendung auf Warmeleitungsprobleme", Computing, 6, pp. 61 – 71.
[7] Kahaner D., Moler, C. And Nash,S., (1989);" Numerical Methods and Software", Prentice Hall, New Jersey.
[8] Shampine, L. and Gladwell,I., (1992);" The next generation of Runge – Kutta codes in computational Ordinary Differential Equations", Cash and Gladwell, eds, the Institute of Mathematics & its Applications, pp. 145 – 164.
[9] Agbeboh, G.U., (2006); " Comparison of some one – step integrators for solving singular initial value problems", Ph. D thesis, A.A.U., Ekpoma.
[10] Agbeboh, G.U. and Omonkaro B. (2010); "On the solution of initial value problems in Ordinary Differential Equations Using a new third order inverse Runge-Kutta Method" (International Journal Of Physical Sciences 5(4) PP299-307). Available online at http:/www.academicjournal.org/IJPS. ISSN 1992-19950 Academic Journals).
[11] Agbeboh, G.U., Ukpebor, L.A. and Esekhaigbe, A.C., (2009); "A modified sixth- order Runge- kutta method for solving initial – value problems in ordinary differential equations", Journal Of Mathematical sciences,20(2)PP 97-110. International Centre For Advance Studies, Nebadhoi School-Para, Dattapukur, North 24 Prgs West Bengal 743248, India. Reef. No. ICFAS/M.16167 (241)/2008).
[12] Esekhaigbe, A.C., (2007); "on the coefficients analysis of a sixth – order Runge – kutta methods for solving initial value problems in ordinary differential equations", M.sc thesis, A.A.U., Ekpoma.
[13] Butcher J.C., (2006); "Numerical methods for ordinary differential equations", Wiley Publication, chichester.
[14] Van der Houwen, P. J., Sommeijer, B. P., (2013); "Numerical solution of second-order fuzzy differential equation using improved Runge-Kutta Nystrom method", Journal of mathematics problems in Engineering vol 3, pp 1-10.
[15] Van der Houwen, P. J., Sommeijer, B. P., (2014); " Runge-Kutta type methods with special properties for the numerical integration or ordinary differential equations", physics reports, 536: 75-146.
[16] Van der Houwen, P. J., Sommeijer, B. P., (2015);" Runge-Kutta projection methods with low dispersion and dissipation errors". Advances in computational methods, 41: 231-251.
[17] Butcher J.C., (1963);" coefficient for the study of Runge-kutta integration processes", J. Austral maths soc., 3:185-201.
[18] Butcher J.C., (1966);" On the convergence of numerical solutions of ordinary differential equations", math. Comp. 20 : 1-10.
[19] Butcher, J. C., (1987);" The Numerical Analysis of Ordinary Differential Equations", J. Wiley & Sons.
[20] Butcher J.C., (2003); "Numerical methods for ordinary differential equations", wiley, chichester.
[21] Butcher J.C., (2010);" Trees and numerical methods for ordinary differential equations", IMA J. Numer. Algorithms 53: 153 – 170.
[22] Connes, A. and kreimer, D., (1999);" lessons from quantum field theory : Hopf algebras and space time geometries" letters in mathematical physics, 48 : 85 – 96.
[23] Kreimer, D., (2007);" Factorization in quantum field theory: an exercise in Hopf algebras and local singularites, frontiers in number theory, physics and geometry II", springer, pp. 715 – 736.
[24] Butcher J.C., (2008);"Numerical methods for ordinary differential equations (2nd ed.) ", John wiley and sons ltd.
[25] Butcher J.C., (2009);" Trees and Numerical methods for ordinary differential equations", Numerical Algorithms (Springer online).
[26] Butcher J.C (2010) ;" Trees, B- series and exponential integrators" , IMA J. numer. Anal., 30: 131 – 140.
[27] Brouder, C., (2000);"Runge – kutta methods and renormalization", Eur. Phys. J .C 12: 521 -534.
[28] Brouder, C (2004);" Trees, renormalization and differential equations", BIT numerical mathematics, 44(3): 425 – 438.

APPENDIX
FORTRAN PROGRAM THAT GENERATED THE RESULTS
PROBLEM 1

```
C PRO
C FOURTH ORDER FOURTH STAGE RUNGE KUTTA METHOD
C OUR PROBLEM IS :Y'=-Y, Y(0)=1
C THEORETICAL SOLUTION:Y(XN)= 1/EXP(XN)
1       DOUBLE PRECISION XN,YN,H,ONE,TWO,EIGHT,EIT,FIV,THREE
```

```
      DOUBLE PRECISION TSOL,ERROR,K1,K2,K3,K4,SIX,FOUR
      OPEN(6,FILE='RUNG2.OUT')
      H=0.1D0
      YN=1.0D0
      XN=0.1D0
      TWO=2.0D0
      EIGHT=18.0D0
      EIT=8.0D0
      THREE=3.0D0
      FIV=5.0D0
      FOUR=4.0D0
      SIX=6.0D0
      ONE=1.0D0
      WRITE(6,101)
 3    K1=-YN
      K2=-(YN+ONE/FOUR*(H*K1))
      K3=-(YN+H/FOUR*(-THREE*K1+SIX*K2))
      K4=-(YN+H*(FIV*K1-SIX*K2+TWO*K3))
      YN=YN+H/EIGHT*(K1+EIT*K2+EIT*K3+K4)
      TSOL=ONE/EXP(XN)
      ERROR=TSOL-YN
      WRITE(6,100)XN,YN,TSOL,ERROR
      XN=XN+H
      IF(XN.LE.ONE) GOTO 3
100   FORMAT(D6.1,1X,3(3X,D19.13))
101   FORMAT(2X,'XN',13X,'YN',16X,'TSOL',16X,'ERROR')
      END
```

PROBLEM 2

```
C PRO
C FOURTH ORDER FOURTH STAGE RUNGE KUTTA METHOD
C OUR PROBLEM IS :Y'=Y, Y(0)=1
C THEORETICAL SOLUTION:Y(XN)=EXP(XN)
      DOUBLE PRECISION XN,YN,H,ONE,THREE,TWO,EIGHT,EIT,FOUR
      DOUBLE PRECISION TSOL,ERROR,K1,K2,K3,K4,FIVE,SIX
      OPEN(6,FILE='RUNG2.OUT')
      H=0.1D0
      YN=1.0D0
      XN=0.1D0
      TWO=2.0D0
      THREE=3.0D0
      FOUR=4.0D0
      SIX=6.0D0
      ONE=1.0D0
      EIT=8.0D0
      EIGHT=18.0D0
      FIVE=5.0D0
      WRITE(6,101)
 3    K1=YN
      K2=YN+ONE/FOUR*(H*K1)
      K3=YN+H/FOUR*(-THREE*K1+SIX*K2)
      K4=YN+H*(FIVE*K1-SIX*K2+TWO*K3)
      YN=YN+H/EIGHT*(K1+EIT*K2+EIT*K3+K4)
      TSOL=EXP(XN)
      ERROR=TSOL-YN
      WRITE(6,100)XN,YN,TSOL,ERROR
      XN=XN+H
      IF(XN.LE.ONE) GOTO 3
```

```
100   FORMAT(D6.1,1X,3(3X,D19.13))
101   FORMAT(2X,'XN',13X,'YN',16X,'TSOL',16X,'ERROR')
      END
```

PROBLEM 3

```
C PRO
C FOURTH ORDER FOURTH STAGE RUNGE KUTTA METHOD
C OUR PROBLEM IS :Y'=1+Y**2, Y(0)=1
C THEORETICAL SOLUTION:Y(XN)=TAN(XN+PI/4)
      DOUBLE PRECISION XN,YN,H,ONE,TWO,EIGHT,PI,THREE,EIT
      DOUBLE PRECISION TSOL,ERROR,K1,K2,K3,K4,SIX,FOUR,FIV
      OPEN(6,FILE='RUNG2.OUT')
      H=0.1D0
      YN=1.0D0
      ONE=1.0D0
      FOUR=4.0D0
      XN=0.1D0
      TWO=2.0D0
      EIGHT=18.0D0
      EIT=8.0D0
      THREE=3.0D0
      FIV=5.0D0
      SIX=6.0D0
      PI=FOUR*DATAN(ONE)
      WRITE(6,101)
3     K1=ONE+YN*YN
      K2=ONE+(YN+(H/FOUR)*K1)**TWO
      K3=ONE+(YN+(H/FOUR)*(-THREE*K1+SIX*K2))**TWO
      K4=ONE+(YN+H*(FIV*K1-SIX*K2+TWO*K3))**TWO
      YN=YN+H/EIGHT*(K1+EIT*K2+EIT*K3+K4)
      TSOL=TAN(XN+(PI/FOUR))
      ERROR=TSOL-YN
      WRITE(6,100)XN,YN,TSOL,ERROR
      XN=XN+H
      IF(XN.LE.ONE) GOTO 3
100   FORMAT(D6.1,1X,3(3X,D19.13))
101   FORMAT(2X,'XN',13X,'YN',16X,'TSOL',16X,'ERROR')
      END
```