INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND
OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

# On the Runge Kutta Fixed Point Iterative Method of Soluition for the Blasius Boundary Value Problem of the Ordinary Differential Equation

Eziokwu, C. Emmanuel [1⋆] and Nkem Ogbonna [2]

1⋆, 2 Department of Mathematics, Michael Okpara University of Agriculture, Umudike, Abia State, Nigeria
⋆ Corresponding Author's email: Okereemm@yahoo.com

**Article Info**
Received: 17 December 2018 Revised:03 April 2019
Accepted: 20 May 2019 Available online: 02 July 2019

**Abstract**

In this paper, we consider the numerical solution of an ordinary differential equation problem with given boundary conditions. In approaching this we used the fixed point iterative method called the Runge Kutta method. This was exactly applied on the Blasius problem which model was formulated and solved iteratively using the FORTRAN programming language software that generated the solution in the last section of this work. The convergence of the solution was seen established.

**Keywords:** Blasius problem, Runge Kutta method, Fixed point iterative method, Convergence

**MSC2010: 34XX, 46B25**

## 1 Introduction

Consider, for example, a second-order equation of the form

$$F\left(x, y, \frac{dy}{dx}, \frac{d^2y}{dx^2}\right) = 0 \tag{1.1}$$

Subject to the boundary conditions $y(a) = \alpha; y(b) = \beta$ : the numerical solution of this problem in most cases is generally much more complicated than the solution for the corresponding initial-value problem [Milne (1953)]. If the original differential equation is linear (that is, if the differential equation is linear in the dependent variable $y$ and all its derivatives), then the set of equations generated by applying a finite difference method to (1.1) is also linear. The system of simultaneous equations can be solved with the iterative methods (for example, the Gauss-Seidel method) if n is large or by the elimination methods if n is small (say $n < 40$). If, as usually happens for equations of order 2, the system of linear equations has a tri-diagonal coefficient matrix, the special form of the Gaussian elimination method [Arden (1963)] may be used, even for $n$ rather large (say n up to 500).


http://ijmao.unilag.edu.ng/article/view/453

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

Unfortunately, when the differential equation is non-linear, the system of equations resulting from a finite-difference method is also non-linear. In addition to the problems of uniqueness associated with the solution of nonlinear equations, the generation of any solution may be very difficult, especially when many base points are used. In some cases, [Colatz (1960)] one can linearize the equations, solve the equations iteratively, then re-linearize about the new solution, find a new solution iteratively, etc. in effect; a complex problem has been replaced by another problem which is somewhat less complex.
Now, having introduced the work as in above background study we then aim to solve the Blasuis boundary value problem using the Runge Kutta numerical method of order four and again establish that this Runge Kutta method is not only a numerical method but also a fixed point iterative method. This will be seen as stated, proved and established in section two below.
The computer result in section three below justifies the objective of this research that the Runge Kutta method is a fixed point iterative method that is strongly pseudo contractive and not just a numerical method only.
However, [McCracken and Dom (1964)] in their work detailed various forms of Boundary value problems together with the associated methods of solution for each of the methods while [Butcher (1964)] detailed the Blasius boundary value problem and the solution using the Runge Kutta method specifically.
However, [Brice, Carrah and Luther (1969)] came out with a clearer write up on the subject yet without answering the question on whether the Blasius boundary value problem is a fixed point iterative method or not and if it is, then what form of fixed point problem is it. These questions, very much important to the mathematical analyst, prompted the authors in addition to solving the boundary value problem to write out this paper addressing the aforementioned questions as vividly presented in section two below.

# 2 Analysis of the Runge-Kutta Iterative Methods for Solving Initial Value Problems

The Runge Kutta iterative methods also called Runge Kutta algorithms are iterative procedures which involve only first order derivative evaluations that produce results equivalent in accuracy to the higher order Taylor formulae. Runge Kutta approximations of the second, third and fourth orders (that is approximations with accuracies equivalent to Taylor's expansions of $y(x)$ retaining terms in $h^2, h^3$, and $h^4$, respectively) require the estimation of $f(x,y)$ at two, three and four values respectively of $x$ on the interval $x_i \leqslant x \leqslant x_{i+1}$.

## 2.1 Main Results

Analytical study of the Runge Kutta methods has revealed the following facts:

**a.** That the domain of existence of solution of the Runge Kutta methods is the complete metric space $(Y,\rho)$ induced by the norm $\rho(x,y) = \|x-y\|,\ x,y \geq 0$

**b.** That the solution of the Runge Kutta methods converge in the metric space $(Y,\rho)$ induced by the norm $\rho(x,y) = \|x-y\|,\ x,y \geq 0$

**c.** The initial value problem $y' = f(x,y); y(x_0) = y_0$ is solvable by the Runge Kutta method in the complete metric space induced by the norm $\rho(x,y) = \|x-y\|,\ x,y \geq 0$ and is also continuous in that norm.

**d.** That the Runge Kutta method satisfies the condition of the strongly pseudo contractive map in the Banach space.

**e.** That the Runge Kutta method is exactly the Mann - like iterative method with a differential operator instead of the usual integral operator.

### Verification

Let $y_{i+1}$ be the iterative fixed point $y^*$ for any given Runge Kutta method defined in $R$, then we want to show that (a) to (e) above are true

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

**a.** The domain of existence of $y^* = y_{i+1}$ is the complete metric space induced by the norm $\|y_{i+1} - y_i\|, y_{i+1}, y_i$ which is Banach such that

- **i.** $\rho(y_{i+1}, y_i) > 0$ is induced by $\|y_{i+1} - y_i\| > 0$
- **ii.** $\rho(y_{i+1}, y_i) = 0$ if and only if $y_{i+1} = y_i$ is induced by $\|y_{i+1} - y_i\| = 0$ if and only if $y_{i+1} = y_i$
- **iii.** $\rho(y_{i+1}, y_i) = \rho(y_i, y_{i+1})$ is induced by $\|y_{i+1} - y_i\| = \|y_i - y_{i+1}\|$
- **iv.** $\rho(y_{i+1}, y_{i-1}) \le \rho(y_{i+1}, y_i) + \rho(y_i, y_{i-1})$ is induced by $\|y_{i+1} - y_{i-1}\| \le \|y_{i+1} - y_i\| + \|y_i - y_{i-1}\|$
- **v.** $\rho(y_{i+1}, y_i)$ is closed and bounded

where $\rho$ is a distance function and $\|\cdot\|$ is a length function for relative points $y_i$.

**i.** If $y_{i+1} \ge 0$ and $y_i \ge 0$such that $y_{i+1} \ge y_i$, then $\rho(y_{i+1}, y_i) \ge 0$ for $\|y_{i+1} - y_i\| \ge 0$ for $\rho$ and $\|\cdot\|$ defined

**ii.** Whenever $y_{i+1} = y_i$, then $\rho(y_{i+1}, y_i) = 0$, then the iteration terminates.

**iii.** If $y_{i+1} \ge 0$ and $y_i \ge 0$ then $\rho(y_{i+1}, y_i) = \rho(y_i, y_{i+1})$ for $\|y_{i+1} - y_i\| = \|y_i - y_{i+1}\|$ and on the other hand if $y_{i+1} < 0$ and $y_i < 0$, also $\rho(y_{i+1}, y_i) = \rho(y_i, y_{i+1}) > 0$

**iv.** Let $y_{i+1}$ and $y_i$ be given then $\rho(y_{i+1}, y_{i-1}) \le \rho(y_{i+1}, y_i) + \rho(y_i, y_{i-1})$

**v.** Let every subsequence in $y^*$ be convergent to a point in $y^*$. Then $y_{i+1}$ is convergent to $y^*$ and hence $\rho(y_{i+1}, y_i)$ defined by $\|y_{i+1} - y_i\|$ is closed and bounded and therefore a complete metric space induced by the norm. Thus we have established that $y^* = y_{i+1}$ exists in the metric space $(Y, \rho)$ induced by the Banach $(X, \|\cdot\|)$.

(b) and (c) obviously follow from (a) proved. We reformulate (d) and (e) as a theorem below and the proof will follow.

**Theorem 2.1.** *Let $Y$ be a Banach space and let $R$ be a region in $(X,Y)$ plane containing $(x_0, y_0)$for $(y_0 \in Y)$. Suppose given*

$$y' = f(x,y); y(x_0) = y_0 \tag{2.1}$$

*a differential equation, if the map $f$ in (2.1) is strongly pseudo-contractive, then the initial value problem (2.1) by the fourth order Runge Kutta method has a unique fixed point*

$$y^* = y_{i+1} = y_i + h\varnothing(x_i, y_i, h) \tag{2.2}$$

*and the fourth order Runge Kutta method is of the form*

$$y^* = y_{i+1} = y_i + h(ak_1 + bk_2 + ck_3 + dk_4) \tag{2.3}$$

*where, $k_1, k_2, k_3, k_4$ are approximate derivative values computed on the interval $x_i \le x \le x_{i+1}$ so that then due to Kutta, the above generalized fourth order reduces to*

$$\left.\begin{array}{c} y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 = f(x_i, y_i) \\ k_2 = f\left(x_i + \frac{1}{2}h, y_i + hk_1\right) \\ k_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}hk_2\right) \\ k_4 = f(x_i + h, y_i + hk_3) \end{array}\right\} \tag{2.4}$$

*Proof.* Given that

$$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Such that

$$k_1 = f(x_i, y_i), k_2 = f\left(x_i + \frac{1}{2}h, y_i + hk_1\right)$$
$$k_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}hk_2\right) \text{ and}$$
$$\mathrm{k}_4 = f(x_i + h, y_i + hk_3)$$

Then

$$\|y_{i+1} - y_{m+1}\| = \left\| \left\{ y_i + \frac{h}{6}\left[ f(x_i, y_i) + 2f\left(x_i + \frac{1}{2}h, y_i + hf(x_i, y_i)\right) + \right.\right.\right.$$
$$2f\left(x_i + \frac{1}{2}h, y_i + hf\left(x_i + \frac{1}{2}h, y_i + hf(x_i, y_i)\right)\right) +$$
$$f\left(x_i + h, y_i + hf\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}h, f\left(x_i + \frac{1}{2}hy_i + hf(x_i, y_i)\right)\right)\right)\Bigg\}$$
$$-\left\{ y_m + \frac{h}{6}\left[ f(x_m, y_m) + 2f\left(x_m + \frac{1}{2}h, y_m + hf(x_m, y_m)\right)\right.\right.$$
$$+2f\left(x_m + \frac{1}{2}h, y_m + hf\left(x_m + \frac{1}{2}h, y_m + hf(x_m, y_m)\right)\right)$$
$$\left.\left.\left. +f\left(x_m + h, y_m + hf\left(x_m + \frac{1}{2}h, y_m + \frac{1}{2}hf\left(x_m + \frac{1}{2}h, y_m + hf(x_m, y_m)\right)\right)\right)\right]\right\}\right\|$$
$$= \left\| \left\{ (y_i - y_m) + \frac{h}{6}[(f(x_i, y_i) - f(x_m - y_m)) \right.\right.$$
$$+2f\left((x_i - x_m) + \frac{1}{2}h, (y_i - y_m) + hf(x_i, y_i) - (x_m, y_m)\right)$$
$$+2f\left((x_i - x_m) + \frac{1}{2}h, (y_i - y_m) + hf((x_i, y_i) - (x_m, y_m))\right)$$
$$+2f(x_i - x_m) + \frac{1}{2}h, (y_i - y_m) + hf(x_i - x_m)$$
$$+hf^2(y_i - y_m) + h^2 f^2((x_i, y_i) - (x_m, y_m))$$
$$+f(x_i - y_i) + hf(y_i, y_m) + hf^2(x_i - x_m)$$
$$+\frac{1}{2}h^2 f^2(y_i - y_m) + \frac{1}{2}h^2 f^3(x_i, x_m)$$
$$\left.\left. +\frac{1}{2}h^3 f^3 \ (y_i - y_m) + \frac{h^3 f^4}{6}(f(x_i, y_i) - f(x_m, y_m))\right]\right\}\right\|$$
$$\le \|y_i - y_m\| + \left\|\frac{h}{6}[f(x_i, y_i) - f(x_m, y_m)]\right\| + \left\|\frac{hf}{3}[x_i - x_m]\right\| + \left\|\frac{h^2 f}{6}[y_i - y_m]\right\|$$
$$+\left\|\frac{h^2 f^2}{3}[f(x_i, y_i) - f(x_m, y_m)]\right\| + \left\|\frac{hf}{3}[x_i - x_m]\right\| + \left\|\frac{h^2 f}{6}[y_i - y_m]\right\| + \left\|\frac{h^2 f^2}{3}[x_i - x_m]\right\|$$
$$+\left\|\frac{h^2 f^3}{3}[y_i - y_m]\right\| + \left\|\frac{h^3 f^2}{3}[f(x_i, y_i) - f(x_m, y_m)]\right\| + \left\|\frac{hf}{6}[x_i - x_m]\right\| + \left\|\frac{h^2 f}{6}[y_i - y_m]\right\|$$
$$+\left\|\frac{h^2 f^2}{6}[x_i - x_m]\right\| + \left\|\frac{h^3 f^2}{3}[y_i - y_m]\right\| + \left\|\frac{h^3 f^3}{3}[x_i - x_m]\right\| + \left\|\frac{h^4 f^3}{3}[y_i - y_m]\right\|$$
$$+\left\|\frac{h^4 f^4}{6}[f(x_i, y_i) - f(x_m, y_m)]\right\|$$

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

$$= \left\{ \|y_i - y_m\| + \left\| \frac{h^2 f}{3} [y_i - y_m] \right\| + \left\| \frac{h^2 f}{6} [y_i - y_m] \right\| + \left\| \frac{h^3 f^2}{3} [y_i - y_m] \right\| \right.$$
$$\left. + \left\| \frac{h^4 f^3}{3} [y_i - y_m] \right\| + \left\| \frac{h^2 f^3}{3} [y_i - y_m] \right\| \right\}$$
$$+ \left\{ \left\| \frac{2hf}{3} [x_i - x_m] \right\| + \left\| \frac{h^2 f^2}{3} [x_i - x_m] \right\| + \left\| \frac{hf}{6} [x_i - x_m] \right\| + \left\| \frac{h^2 f^2}{6} [x_i - x_m] \right\| \right.$$
$$+ \left\| \frac{h^3 f^3}{3} [x_i - x_m] \right\| + \left\| \frac{h}{6} [f(x_i, y_i) - f(x_m, y_m)] \right\|$$
$$\left. + \left\| \frac{h^4 f^4}{6} [f(x_i, y_i) - f(x_m, y_m)] \right\| \right\} + \left\{ \left\| \frac{h^2 f^2}{3} [(x_i, y_i) - (x_m, y_m)] \right\| \right.$$
$$\left. + \left\| \frac{h^3 f^3}{3} [(x_i, y_i) - (x_m, y_m)] \right\| \right\} > \left\| 1 + \frac{h^2 f}{2} + \frac{h^2 f^3}{3} + \frac{2h^3 f^3}{3} + \frac{h^4 f^3}{3} [y_i - y_m] \right\|$$
$$- \left\| \left( \frac{h}{6} + \frac{hf}{2} + \frac{h^2 f^2}{6} + \frac{h^2 f^3}{3} + \frac{h^3 f^3}{3} + \frac{h^4 f^4}{6} \right) [x_i - y_i] [F(x_i, y_i) - F(x_m, y_m)] \right\|$$
$$= \|(1+\alpha)[y_i - y_m] - \alpha KF [x_i - y_i]\| = \|(1+\alpha) I - \alpha KF\| \|x_i - y_i\| > \|x_i - y_i\|$$

or equivalently

$$\|(1+\alpha) I - \alpha KF\| \|y_i - x_i\| > \|y_i - x_i\|$$

where $k =$ arbitrary constant
$F = [F(x_i, y_i) - F(x_m, y_m)]$ a Lipchitzian fixed point map

$$\alpha = \frac{h^2 f}{3} + \frac{h^2 f^3}{3} + \frac{2h^2 f^3}{3} + \frac{h^4 f^3}{3}$$

and/or

$$\frac{h}{6} + \frac{hf}{2} + \frac{h^2 f^2}{6} + \frac{h^2 f^3}{3} + \frac{h^3 f^3}{3} + \frac{h^4 f^4}{6}$$

which indicates that the iterative method is strongly pseudo-contractive because automatically

$$\|(1+\alpha) I - \alpha KF\| > 1$$

Hence the Mann-like iterative method, the Runge Kutta method is a fixed point iterative method which converges to a unique fixed point □

## 2.2 Convergence Analysis

Convergence is the first basic requirement that a reasonable numerical method must satisfy. It requires that we get arbitrary accurate solution if we put sufficient effort into solving the problem by taking sufficiently small. To analyze the convergence of the Runge Kutta methods, we shall first of all define the following term.

### 2.2.1 Local Truncation Error

Let $y_{n+k} = y_n + h\varnothing(x_n, y_n, h)$ be a one-step method and $y(x)$ be a solution of the differential equation $y' = f(x, y)$. The local truncation error for$y(x)$ is defined as [Butcher (2003)]

$$T(x, h) = y_{n+1} - (y_n + h\varnothing(x, y(x), h)) \tag{2.5}$$

i.e. the local truncation error is the amount by which the true solution of the differential equation fails to satisfy the numerical scheme $T(x, h)$ is defined for those $x, h$ for which $0 < h \leq b - a$ and $a \leq x \leq b - h$. Define $r(h, t) = \frac{T(x,h)}{h}$ and set $r_2(h) = r(x_i, h)$ also take

$$r(h) = ^{\max}_{a \leq x \leq b-h} |r(x, h)| \text{ for } 0 < h \leq b - a \tag{2.6}$$

We can now state and prove the following theorem.

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND
OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

**Theorem 2.2.1 [Butcher (2003)]**

Suppose $f(x,y)$ is continuous in $x,y$ and uniformly Lipschitz in $y$ on $[a,b]\times\Re^n$. Let $y(x)$ be the solution of the initial value problem $y' = f(x,y)$, $y(x_0) = y_0$ on $[a,b]$.
Suppose that the function $\varnothing(x,y,h)$ in the one step method satisfies the following two conditions;

**i Stability:** $\varnothing(x,y,h)$ is continuous in $x,y,h$ and uniformly Lipschitz constant $k$ on

$$0 \le h \le h_0, a \le x \le b, y \in \Re^n \tag{2.7}$$

for some $h_0 > 0$ with

$$h_0 \le b-a \tag{2.8}$$

**ii Consistency:** $\lim_{h\to 0}\varnothing(x,y,h) = f(x,y)$ and $e_i(h) = y(x_i(h) - y_i(h))$ where $y_i$ is obtained from the one step method

$$y_{i+1} = y_i + h\varnothing(x_i,y_i,h) \tag{2.9}$$

then,

$$\begin{aligned} |e_i(h)| &\le e^{k(x_i(h))-a}|e_0(h)| + r(h)\left(\tfrac{e^k(x_i(h)-a)-1}{k}\right) \\ |e_i(h)| &\le e^{k(x_i(h))-a}|e_0(h)| + \tfrac{e^k(x_i(h)-a)-1}{k} \end{aligned} \tag{2.10}$$

Moreover, $r(h) \to 0$ as $h \to 0$
Therefore, if $e_0(h) \to 0$ as $h \to 0$, then

$$\max_{0\le x\le \frac{b-a}{a}} |e_i(h)| \to 0 \text{ as } h \to 0 \tag{2.11}$$

i.e. the approximations converge uniformly on the grid to the solution

**Proof**

$$y_{i+1} = y_i + h\varnothing(x_i,y_i,h) \tag{2.12}$$

From

$$\begin{aligned} &y(x_{i+1}) - y(x_i) + h\emptyset(x_i,y(x_i),h) + hr_2 \\ &|e_{i+1}| \le |e_i| + h|\emptyset(x_i,y(x_i),h) - \emptyset(x_i,y_i,h)| + h|r_2| \\ &\le |e_i| + hK|e_i| + hr(h) - (1+hk)|e_i| + hr(h) \end{aligned} \tag{2.13}$$

So,

$$|e_1| \le (1+hk)|e_0| + hr(h) \tag{2.14}$$

and

$$|e_2| \le (1+hk)|e_1| + hr(h) \le (1+hk)^2|e_0| + hr(h)(1+(1+hk)) \tag{2.15}$$

By induction

$$\begin{aligned} |e_1| &\le (1+hk)^2|e_0| + hr(h)\left(1+(1+hk)+(1+hk)^2+\cdots+(1+hk)^{i-1}\right) \\ &= (1+hk)^i|e_0| + hr(h)\tfrac{(1+hk)^i-1}{(1+hk)-1} = (1+hk)^i|e_0| + r(h)\tfrac{(1+hk)^i-1}{K} \end{aligned} \tag{2.16}$$

Since $(1+hk)^{\frac{1}{h}} \uparrow e^k$ as $h \to 0$ (for $k > 0$ and $i = \frac{x_i-a}{h}$ , we have

$$(1+hk)' = (1+hk)^{\frac{x_1-a}{h}} \le e^{k(x_i-a)} \tag{2.17}$$

thus

$$|e_i| \le e^{k(x_i-a)}|e_0| + r(h)\frac{e^{k(x_i-a)}-1}{K} \tag{2.18}$$

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND
OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

If $f$ is sufficiently smooth, then we know that $y(x) \in C^{p+1}$. The theorem thus implies that if a one step method is accurate of order $P$ and stable than for sufficiently smooth $f, T(x,h) = 0\left(h^{p+1}\right)$ and thus $r(h) = 0(h^p)$ if in addition, $e_0(h) = 0(h^p)$

$$\max_i |e_i(h)| = 0(h^p) \tag{2.19}$$

i.e. we have a $pth$ order convergence of the numerical approximation to the solution which completes the proof. Now, for each Runge Kutta method we shall investigate its stability and consistency. We shall assume continuity of $\varnothing(x,y,h)$ since $f(x,y)$ is continuous.

## 2.3 Convergence Corollaries on the various Runge Kutta methods discussed

### 2.3.1 Forward Euler

**Stability:**

$$y_{i+1} = y_n + hf(x_n, y_n) \Rightarrow \varnothing(x_n, y_n, h) = f(x_n, y_n) \tag{2.20}$$

For $\varnothing(x_n, y_n, h)$ to be Lipschitz, we have

$$\begin{aligned} &|\emptyset(x,y_1,h) - \emptyset(x,y_2,h)| = |f(x,y_1) - f(x,y_2)| \le L|y_1 - y_2| \\ &\Rightarrow |\emptyset(x,y_1,h) - \emptyset(x,y_2,h)| \le L|y_1 - y_2| \end{aligned} \tag{2.21}$$

Hence the method is stable.

**Consistency:**

Given that $\varnothing(x_n, y_n, h) = f(x_n, y_n)$ from definition

$$\lim_{h\to 0} \varnothing(x_n, y_n, h) = \lim_{h\to 0} f(x_n, y_n) = f(x_n, y_n) \tag{2.22}$$

and consequently the method is convergent.

# 3 Blasius Problem Statement

In consideration of the Blasius boundary value problem as explained [Young (2009)] to , [Camara (2005)] and [Peter (1994)], then according to [Butcher (1964)] and [Ihiagwam (2001)], the resultant solution is by applying fourth order Runge Kutta method of solution to the resultant single order equation in the form of dimensionless stream functions and its associated boundary conditions. We then use a computer to implement this and finally point tabulated values of $\eta, f'(u), f''$ and $v = (\eta f' - f)$ and from their produce a graph of $u$ and $v$ versus $\eta$ on the online printer. All these are follow up of the method of solution outlined below in the next subsection.

## 3.1 Derivation Method

The governing equations of motion and continuity are

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = v\frac{\partial^2 u}{\partial y^2} \tag{3.1}$$

$$\frac{\partial u}{\partial x} + v\frac{\partial v}{\partial y} = 0 \tag{3.2}$$

Subject to the boundary conditions

$$y = 0 : u = v = 0 \tag{3.3}$$

$$y = \infty; u = u_\infty \tag{3.4}$$

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

by introducing the dimensionless coordinate

$$\eta = y\sqrt{\frac{u_\infty}{vx}} \tag{3.5}$$

and a stream function

$$\psi = \sqrt{vxu_\infty f(\eta)} \tag{3.6}$$

Such that

$$u = \frac{\partial \psi}{dy}, v = \frac{\partial \psi}{dx} \tag{3.7}$$

Equations (3.1.1) and (3.1.2) reduce to a single ordinary differential equation in the dimensionless stream function $f(\eta)$:

$$ff'' + 2f'' = 0 \tag{3.8}$$

the boundary conditions now become

$$\eta = 0 : f = 0, f' = 0 \tag{3.9}$$

$$\eta = \infty : f' = 1 \tag{3.10}$$

From the above equations involving stream function, we can derive the following relations for dimensionless velocities $U$ and $V$

$$U = \frac{u}{u_\infty} = f' \tag{3.11}$$

$$V = \frac{v}{\frac{1}{2}\sqrt{\frac{vu_\infty}{x}}} = \eta f' - f \tag{3.12}$$

Thus, once (3.1.5) has been solved for $f$, we can predict the velocity field from (3.1.8) and (3.1.9)

## 3.2 Method of Solution

By defining $g_1 = f, g_2 = f'$ and $g_3 = f''$, we can replace equation (3.1.5) by an equivalent set of three first order equations

$$\begin{aligned} \frac{dg_1}{d\eta} &= g_2 \\ \frac{dg_2}{d\eta} &= g_3 \\ \frac{dg_3}{d\eta} &= -\frac{1}{2}g_1 g_3 \end{aligned} \tag{3.13}$$

Subject to

$$\eta = 0 : g_1 = g_2 = 0 \tag{3.14}$$

$$n = \infty : g_2 = 1 \tag{3.15}$$

Starting at $\eta = 0$, the integration of (3.2.1) over successive steps $\eta$ is achieved by the fourth order Runge Kutta function RUNGE: this is described fully in and the details will not be prepared here. Since we have initial conditions $(at \eta = 0)$ for $g_1$ and $g_2$ only, we search for a value of $g_3$ at $\eta = 0$ that will generate a solution that yields $g_2 = 1$ at $\eta = \infty$. This is accomplished by the half-interval method. In practice, condition (3.2.3) must be replaced by the appropriate condition

$$\eta = \eta_{\max} : g_2 = 1 \tag{3.16}$$

Where $\eta_{\max}$ is arbitrary, as long as it is chosen large enough so that solution shows little further changing for $\eta$ larger than $\eta_{\max}$. This corresponds to the physical fact that the mainstream velocity is effectively equal to $u_\infty$ far we start with two limits $g_{3L}$ and $g_{3R}$, between which is the missing initial condition $g_{3O}$, is thought to lie. The improving at each iteration the value chosen for $g_{3O}$ and at each iteration, we set $g_{3O}$ of the current interval to have been adjusted according to the half-interval method. Then the Fortran programming package was used to implement the above process and the result generated was presented below.

INTERNATIONAL JOURNAL OF MATHEMATICAL ANALYSIS AND
OPTIMIZATION: THEORY AND APPLICATIONS
VOL. 2019, NO. 1, PP. 483 - 495

## Computer Output

ITER=1
G3LEFT=0.100000
G3ZERO=0.300000
G3RITE=0.500000

Table 3.1: Results for the 1st Half Interval Iteration

| ETA | G(1) | G(2) | G(3) | ORD |
|---|---|---|---|---|
| 0.0 | 0.0 | 0.0 | 0.30000000 | 0.0 |
| 0.2000 | 0.0059999 | 0.0599970 | 0.2994001 | 0.00299976 |
| 0.4000 | 0.0239962 | 0.1199520 | 0.29952042 | 0.01199232 |
| 0.6000 | 0.0539709 | 0.1797574 | 0.29838480 | 0.02694178 |
| 0.8000 | 0.0958775 | 0.2392351 | 0.29618689 | 0.04775530 |
| 1.0000 | 0.1496269 | 0.2981396 | 0.29260209 | 0.07425638 |
| 1.2000 | 0.2150748 | 0.3561641 | 0.28734261 | 0.10616110 |
| 1.4000 | 0.2920100 | 0.4129493 | 0.28734261 | 0.14305952 |
| 1.6000 | 0.3801453 | 0.4680962 | 0.27094067 | 0.18440433 |
| 1.8000 | 0.4791112 | 0.5211829 | 0.25956977 | 0.22950905 |
| 2.0000 | 0.5884528 | 0.5717839 | 0.24609846 | 0.27755748 |
| 2.2000 | 0.7076321 | 0.6194919 | 0.23067436 | 0.32762517 |
| 2.6000 | 0.9729702 | 0.7048249 | 0.19510072 | 0.42978727 |
| 2.8000 | 1.1177093 | 0.7419206 | 0.17574671 | 0.47983427 |
| 3.0000 | 1.2694769 | 0.7750961 | 0.15598188 | 0.52790577 |

| | | | | |
|---|---|---|---|---|
| 3.2000 | 1.4274840 | 0.8043197 | 0.13631121 | 0.57316950 |
| 3.4000 | 1.5909456 | 0.8296594 | 0.11722104 | 0.61494820 |
| 3.6000 | 1.7590994 | 0.8512760 | 0.09914617 | 0.65274702 |
| 3.8000 | 1.9312237 | 0.8694095 | 0.08244318 | 0.68626629 |
| 4.0000 | 2.1066511 | 0.8843623 | 0.06737276 | 0.71539902 |
| 4.2000 | 2.2847795 | 0.8964782 | 0.05409218 | 0.74021444 |
| 4.4000 | 2.4650776 | 0.9061224 | 0.04265794 | 0.76093052 |
| 4.6000 | 2.6470882 | 0.9136624 | 0.03303667 | 0.77787931 |
| 4.8000 | 2.8304259 | 0.9194510 | 0.02512223 | 0.79146938 |
| 5.0000 | 3.0147737 | 0.9238145 | 0.01875577 | 0.80214937 |
| 5.2000 | 3.1998762 | 0.9270438 | 0.01374639 | 0.81037566 |
| 5.4000 | 3.3855324 | 0.9293899 | 0.00988988 | 0.81658642 |
| 5.6000 | 3.5715874 | 0.9310630 | 0.00698428 | 0.82118282 |
| 5.8000 | 3.7579243 | 0.9322343 | 0.00484134 | 0.82451747 |
| 6.0000 | 3.9444568 | 0.9330392 | 0.00329382 | 0.82854309 |
| 6.2000 | 4.1311225 | 0.9335821 | 0.00219967 | 0.82854309 |
| 6.4000 | 4.3178774 | 0.9339414 | 0.00144176 | 0.82854309 |
| 6.6000 | 4.5046908 | 0.9341749 | 0.00092751 | 0.82967386 |
| 6.8000 | 4.6915418 | 0.9343239 | 0.00058564 | 0.83043193 |
| 7.0000 | 4.8784168 | 0.9344171 | 0.00036294 | 0.83093028 |
| 7.2000 | 5.0653063 | 0.9344744 | 0.00022076 | 0.83125155 |
| 7.4000 | 5.2522049 | 0.9345089 | 0.00013179 | 0.83145467 |
| 7.6000 | 5.4391089 | 0.9345294 | 0.00007722 | 0.83158061 |
| 7.8000 | 5.6260161 | 0.9345413 | 0.00004441 | 0.83165720 |
| 8.0000 | 5.8129251 | 0.9345480 | 0.00002507 | 0.83170288 |
| 8.2000 | 5.9998351 | 0.9345518 | 0.00001389 | 0.83172961 |
| 8.4000 | 6.1867457 | 0.9345539 | 0.00000755 | 0.83174494 |
| 8.6000 | 6.3736566 | 0.9345550 | 0.00000403 | 0.83175358 |
| 8.8000 | 6.5605677 | 0.9345556 | 0.00000211 | 0.83175834 |
| 9.0000 | 6.7474788 | 0.9345559 | 0.00000109 | 0.83176093 |
| 9.2000 | 6.9343900 | 0.9345561 | 0.00000055 | 0.83176230 |
| 9.4000 | 7.1213012 | 0.9345562 | 0.00000027 | 0.83176301 |
| 9.6000 | 7.3082125 | 0.9345562 | 0.00000013 | 0.83176338 |
| 9.8000 | 7.4951237 | 0.9345562 | 0.00000003 | 0.83176370 |

# COMPUTER OUTPUT

ITER = 20
G3LEFT = 0.332057
G3ZERO = 0.332058
G3RITE = 0.332058

Table 3.2: Results for the $20^{th}$ (Final) Half-Interval Iteration

| ETA | G(1) | G(2) | G(3) | ORD |
|---|---|---|---|---|
| 0.0 | 0.0 | 0.0 | 0.33205757 | 0.0 |
| 0.2000 | 0.0066410 | 0.0664078 | 0.33198407 | 0.00332028 |
| 0.4000 | 0.0265599 | 0.1327643 | 0.33198407 | 0.01327289 |
| 0.6000 | 0.0597347 | 0.1989374 | 0.33007936 | 0.02981386 |
| 0.8000 | 0.1061083 | 0.2647093 | 0.32738950 | 0.05282956 |
| 1.0000 | 0.1655719 | 0.3297803 | 0.32300734 | 0.08210419 |
| 1.2000 | 0.2379489 | 0.3937764 | 0.31658941 | 0.11729136 |
| 1.4000 | 0.3229819 | 0.4562621 | 0.30786560 | 0.15789252 |
| 1.6000 | 0.4203211 | 0.5167571 | 0.29666365 | 0.20342514 |
| 1.8000 | 0.5295185 | 0.5747585 | 0.28293119 | 0.25252343 |
| 2.0000 | 0.6500249 | 0.6297661 | 0.26675170 | 0.30475369 |
| 2.2000 | 0.7811939 | 0.6813108 | 0.24835105 | 0.35884491 |
| 2.6000 | 0.9222908 | 0.7289824 | 0.22809187 | 0.41363344 |
| 2.8000 | 0.0725068 | 0.7724555 | 0.20645472 | 0.46793874 |
| 3.2000 | 0.2309782 | 0.8115101 | 0.18400666 | 0.52062503 |
| 3.4000 | 0.7469513 | 0.8460449 | 0.16136037 | 0.57066276 |
| 3.6000 | 0.9295265 | 0.8760819 | 0.13912809 | 0.61718306 |
| 3.8000 | 2.1160312 | 0.9017617 | 0.11787627 | 0.65951923 |
| 4.0000 | 2.3057479 | 0.9233301 | 0.09808630 | 0.69723100 |
| 4.2000 | 2.4980413 | 0.9411185 | 0.08012594 | 0.73010945 |
| 4.4000 | 2.6923627 | 0.9555187 | 0.06423415 | 0.75816339 |
| 4.6000 | 2.8882498 | 0.9669575 | 0.0.0551979 | 0.78159014 |
| 4.8000 | 2.0853226 | 0.9758713 | 0.03897266 | 0.80073545 |
| 5.0000 | 3.2832757 | 0.9915423 | 0.01590668 | 0.83721795 |
| 5.2000 | 3.4818697 | 0.9942459 | 0.01134187 | 0.84410461 |

| | | | | |
|---|---|---|---|---|
| 5.4000 | 3.6809213 | 0.9961557 | 0.00792775 | 0.84915981 |
| 5.6000 | 3.8802930 | 0.9974782 | 0.00543204 | 0.85279243 |
| 5.8000 | 4.0798843 | 0.9983759 | 0.00364849 | 0.85534799 |
| 6.0000 | 4.2796234 | 0.9989733 | 0.00240211 | 0.85710820 |
| 6.2000 | 4.4794599 | 0.9993630 | 0.00155023 | 0.85829528 |
| 6.4000 | 4.6793593 | 0.9996121 | 0.00088067 | 0.85907921 |
| 6.6000 | 4.8792986 | 0.9997683 | 0.00060808 | 0.85958615 |
| 6.8000 | 5.0792626 | 0.9998643 | 0.00036959 | 0.85990719 |
| 7.0000 | 5.2792417 | 0.9999221 | 0.00022019 | 0.86010631 |
| 7.2000 | 5.4792299 | 0.9999562 | 0.00012859 | 0.86022727 |
| 7.4000 | 5.6792232 | 0.9999759 | 0.00007361 | 0.86029924 |
| 7.6000 | 5.8792197 | 0.9999871 | 0.00004130 | 0.86034118 |
| 7.8000 | 5.0792178 | 0.9999933 | 0.00002271 | 0.86036512 |
| 8.0000 | 6.2792168 | 0.9999967 | 0.00001224 | 0.86037851 |
| 8.2000 | 6.4792164 | 0.9999985 | 0.00000647 | 0.86038585 |
| 8.4000 | 6.6792162 | 0.9999995 | 0.00000335 | 0.86038978 |
| 8.6000 | 6.8792161 | 1.0000000 | 0.00000170 | 0.86039185 |
| 8.8000 | 7.0792161 | 1.0000002 | 0.00000085 | 0.86039292 |
| 9.0000 | 7.2792162 | 1.0000003 | 0.00000041 | 0.86039346 |
| 9.2000 | 7.4792163 | 1.0000004 | 0.00000020 | 0.86039372 |
| 9.4000 | 7.6792164 | 1.0000004 | 0.00000009 | 0.86039385 |
| 9.6000 | 7.8792164 | 1.0000004 | 0.00000004 | 0.86039391 |
| 9.8000 | 8.0792165 | 1.0000005 | 0.00000002 | 0.86039394 |
| 10.000 | 8.2792166 | 1.0000005 | 0.00000001 | 0.86039395 |

## Discussion of Results

The results of the computations, performed in double precision arithmetic, are displayed above for the first and 20th (final) half-interval iteration. The success of the method can be seen by examining $g_2(\eta max)$, which differs only marginally from the required value of 1. The corresponding values for $g_2(\eta max)$ at the end of the 1st and 10th iterations were 0.9345562 and 1.0007317, respectively. The results for the final iteration agree to at least six significant digits with those reported. Observe that $U$ essentially attains its mainstream value for $h = 6$ or greater. Since $\partial u / \partial x$ is everywhere negative (at a given distance from the plate, the $u$ velocity is retarded in the direction of flow), a positive $V$ velocity occurs in order to satisfy the continuity equation (3.1.2).

## References

[1] Arden, B.W. (1963) "An introduction to digital computing", Addison-Wesley, Reading, Massachusetts

[2] Brice, Carnaha, Luther, H.A., Wikes, James O. (1969) "Applied Numerical methods." John Wesley and sons Inc. USA

[3] Butcher, J.C. (1964) "Implicit Runge Kutta Processes," Math. Comp. 18.50-64

[4] Butcher (2003) "The Numerical Analysis of Ordinary Differential Equations" John Willey and Sons New York, U.S.A.

[5] Camara, S.K. (2005) "Analysis of different Runge Kutta Methods as applied to Boundary value problems of Ordinary Differential Equations" *Journal of computational methods* Nol. 8(54-71)

[6] Collatz, L. (1960) "The Numerical Treatment of Differential Equations", Third Ed. Springer-Verlag, Berlin

[7] Eziokwu, E.C. (2019) "Application of Runge Kutta iterative methods in the solution of ordinary differential equations" *Osaka Journal of Mathematics* Vol. 1(21-42)

[8] Gear, C.W. (1965) "Hybrid Methods for initial Value Problems in Ordinary Differential Equations," *Journal of the S.I.A.M* 2 69-86

[9] Hildebrand, F.B. (1956) "Introduction to Numerical Analysis", McGraw-Hill, New York

[10] Ihiagwam P.C. (2001) "Abstract of the Nigerian Mathematical Society" Vol. 1(5-8)

[11] Lapidus, L. (1962) "Digital Computation for Chemical Engineers," McGraw-Hill, New York

[12] McCracken, D.D. and Dorn, W.S.(1964)"Numerical Methods and FORTRAN Programming",Wiley, NewYork

[13] Milne,W.E and Reynolds, R.R. (1959) "Stability of a Numerical Solution of Differential Equations," *Journal of the A.C.M.* 6 196-203

[14] Milne, W.E. (1953) "Numerical Solution of Differential Equations," Wiley, New York

[15] Perry, J.H. (1950) "Chemical Engineers' Handbook," 3rd ed. McGraw-Hill, New York

[16] Peter, R.T. (1994) "Numerical Analysis" PSTF

[17] Young, M.T. (2009) "Runge Kutta Methods in the solution of Ordinary Differential Equations. Bulletin of the Brazilian Mathematical Society ol. 1(18-30)