

Research Article

A New Spectral Local Linearization Method for Nonlinear Boundary Layer Flow Problems

S. S. Motsa

School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Private Bag X01, Scottsville, Pietermaritzburg 3209, South Africa

Correspondence should be addressed to S. S. Motsa; sandilemotsa@gmail.com

Received 7 June 2013; Revised 9 August 2013; Accepted 9 August 2013

Academic Editor: Chein-Shan Liu

Copyright © 2013 S. S. Motsa. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a simple and efficient method for solving highly nonlinear systems of boundary layer flow problems with exponentially decaying profiles. The algorithm of the proposed method is based on an innovative idea of linearizing and decoupling the governing systems of equations and reducing them into a sequence of subsystems of differential equations which are solved using spectral collocation methods. The applicability of the proposed method, hereinafter referred to as the spectral local linearization method (SLLM), is tested on some well-known boundary layer flow equations. The numerical results presented in this investigation indicate that the proposed method, despite being easy to develop and numerically implement, is very robust in that it converges rapidly to yield accurate results and is more efficient in solving very large systems of nonlinear boundary value problems of the similarity variable boundary layer type. The accuracy and numerical stability of the SLLM can further be improved by using successive overrelaxation techniques.

1. Introduction

The important properties and applications of the boundary layer flows in many engineering areas are well documented and have been widely researched in the last few decades. The mathematical model of the boundary layer flow system is described, in general, by a set of nonlinear partial differential equations which cannot be solved exactly. A common approach that is employed in the solution of boundary layer flow problems is the reduction of the governing partial differential equations into a set of ordinary differential equations (ODEs) after using appropriate assumptions and a set of transformations based on the so-called similarity variables and using the so-called stream function formulations. The transformed ODEs with the corresponding boundary conditions becomes a system of highly nonlinear boundary value problems (BVP) which, in most cases, does not have closed form solutions. The solution of these nonlinear BVPs continues to fascinate and inspire researchers to develop methods of obtaining solutions to these equations which elucidate the intricate properties of the underlying boundary layer flow problem under different conditions.

The standard way of solving the transformed similarity variable boundary layer equations is the numerical approach based on the shooting algorithm with the Runge-Kutta scheme [1, 2]. Other numerical approaches that have been preferred by some researchers include the finite difference method [3], Keller-Box method [4, 5], spectral homotopy analysis method [6, 7], element free Galerkin method [8], and Newton-Raphson based methods such as the quasilinearization method of [9, 10] and the successive linearization method [11–13]. Some analytical approaches have also been found to be very useful in solving boundary layer equations. Examples include the homotopy analysis method [14–16], the homotopy perturbation method [17–19], the differential transformation method [20–22], the Lie-group shooting method [23], the parameter iteration method [24], and the variational iteration method [25], among many other approaches. However, analytical methods are limited in their applications and can only be used in simple systems with few equations. Most real life applications in fluid mechanics involve multiple interacting physical processes and are modelled using complex multiple equation systems.

For such problems, it is impractical to use analytical methods because their solution process becomes too cumbersome, and convergence to the true solution can be very slow or not possible at all. For, this reason numerical methods are by far the most practical way of seeking solutions to the boundary layer flow type of highly nonlinear systems. Numerical solution methods also have their own disadvantages such as the stability and convergence issues, difficulty in dealing with singularities, and limiting cases. In addition, some numerical solutions are difficult to interpret such as the occurrence of multiple solutions. The quest for the most optimal method of solving nonlinear problems in fluid mechanics is what drives, ever growing interest in the development of new methods and the modification and improvement of existing analytical and numerical methods.

The prime objective of this paper is to present a new numerical method of solving boundary layer equations that seeks to address some of the aforementioned numerical difficulties. We propose a very simple, yet very accurate and convergent iterative algorithm for solving nonlinear systems of equations that model boundary layer flow problems. The proposed method, hereinafter referred to as spectral local linearization method (SLLM), is based on, decoupling and linearizing systems of equations using a combination of a univariate linearization technique and a spectral collocation discretization. The key feature of the SLLM algorithm is that it breaks down a large coupled system of equations into a sequence of smaller subsystems which can be solved sequentially in a very computationally efficient manner. The applicability of the proposed method is tested on the well-known Blasius boundary layer problem and a three-equation coupled system that models the problem of unsteady free convective heat and mass transfer which are used as test cases to compare the performance of the proposed SLLM when compared to other existing methods. The computed SLLM results demonstrate that the method is easy to develop, accurate, convergent, stable, and very efficient when compared with other existing methods of solving some large systems of boundary value problems.

2. Description of the Methods of Solution

This section presents a brief description of how the proposed iterative methods of solution are developed for a general system of m nonlinear ordinary differential equations in m unknown functions.

2.1. Spectral Local Linearization Method (SLLM). Here, we describe the development of the spectral local linearization method (SLLM). Consider a system of m nonlinear ordinary differential equations in m unknown functions $Z_i(\eta)$ $i = 1, 2, \dots, m$, where $\eta \in [a, b]$ is the dependent variable. The system can be written in terms of Z_i as a sum of its linear (L_i) and nonlinear components (N_i) as

$$L_i [Z_1, Z_2, \dots, Z_m] + N_i [Z_1, Z_2, \dots, Z_m] = 0, \quad (1)$$

$$i = 1, \dots, m.$$

To develop the iteration scheme, we apply local linearization of N_i about $Z_{i,r}$ (the previous iteration) to the i th nonlinear equation assuming that all other $Z_{k,r}$ ($k \neq i$) are known. Thus, at the i th equation, N_i is linearized as follows:

$$N_i [Z_1, Z_2, \dots, Z_m] = N_i [Z_{1,r}, Z_{2,r}, \dots, Z_{m,r}] + \frac{\partial N_i}{\partial Z_i} [Z_{1,r}, Z_{2,r}, \dots, Z_{m,r}] (Z_i - Z_{i,r}). \quad (2)$$

Thus, at the current iteration with $Z_i = Z_{i,r+1}$, (1) becomes

$$L_i [Z_{1,r+1}, \dots, Z_{m,r+1}] + \frac{\partial N_i}{\partial Z_i} [\dots] Z_{i,r+1} = \frac{\partial N_i}{\partial Z_i} [\dots] Z_{i,r} - N_i [Z_{1,r}, \dots, Z_{m,r}], \quad (3)$$

where $[\dots]$ denotes $[Z_{1,r}, Z_{2,r}, \dots, Z_{m,r}]$ and $Z_{i,r+1}$ and $Z_{i,r}$ are the approximations of Z_i at the current and the previous iteration, respectively. To obtain a decoupled iteration scheme, we appeal to the Gauss-Seidel approach of decoupling linear algebraic systems in linear algebra applications. We therefore arrange the equations in a particular order and solve them in a chronological order. In seeking the solution of Z_i in the current iteration level, $Z_{i,r+1}$, we use updated solutions of Z_s ($s < i$) obtained as solutions of the previous $i = 1, 2, \dots, s$ equations. Thus, for a system of m equations, the local linearization iteration scheme becomes

$$L_1 [Z_{1,r+1}, Z_{2,r}, \dots, Z_{m,r}] + \frac{\partial N_1}{\partial Z_1} [\dots] Z_{1,r+1} = \frac{\partial N_1}{\partial Z_1} [\dots] Z_{1,r} - N_1 [Z_{1,r}, \dots, Z_{m,r}],$$

$$L_2 [Z_{1,r+1}, Z_{2,r+1}, Z_{3,r}, \dots, Z_{m,r}] + \frac{\partial N_2}{\partial Z_2} [\dots] Z_{2,r+1} = \frac{\partial N_2}{\partial Z_2} [\dots] Z_{2,r} - N_2 [Z_{1,r+1}, Z_{2,r}, \dots, Z_{m,r}],$$

$$\vdots$$

$$L_m [Z_{1,r+1}, Z_{2,r+1}, \dots, Z_{m,r+1}] + \frac{\partial N_m}{\partial Z_m} [\dots] Z_{m,r+1} = \frac{\partial N_m}{\partial Z_m} [\dots] Z_{m,r} - N_m [Z_{1,r+1}, \dots, Z_{m-1,r+1}, Z_{m,r}], \quad (4)$$

where, at the i th equation, $[\dots] \equiv [Z_{1,r+1}, Z_{2,r+1}, \dots, Z_{i-1,r+1}, Z_{i,r}, \dots, Z_{m,r}]$.

Thus, starting from an initial approximation $Z_{1,0}, Z_{2,0}, \dots, Z_{m,0}$, the proposed iterative scheme (4) is then solved as a loop until the system converges at a consistent solution for all the variables. To solve the iteration scheme (4), it is convenient to use the Chebyshev pseudospectral method. For this reason the proposed method is referred to as the spectral

local linearization iteration method (SLLM) in this work. Spectral methods are now becoming the preferred tools for solving ordinary and partial differential equations because of their elegance and high accuracy in resolving problems with smooth functions.

For brevity, we omit the details of the spectral methods and refer interested readers to [26, 27]. Before applying the spectral method, it is convenient to transform the domain on which the governing equation is defined to the interval $[-1, 1]$ on which the spectral method can be implemented. We use the transformation $\eta = (b - a)(\tau + 1)/2$ to map the interval $[a, b]$ on $[-1, 1]$. The basic idea behind the spectral collocation method is the introduction of a differentiation matrix D which is used to approximate the derivatives of the unknown variables $Z_i(\eta)$ at the collocation points as the matrix vector product

$$\frac{dZ_i}{d\eta} = \sum_{k=0}^{\bar{N}} \mathbf{D}_{lk} Z_i(\tau_k) = \mathbf{D}Z_i, \quad l = 0, 1, \dots, \bar{N}, \quad (5)$$

where $\bar{N} + 1$ is the number of the collocation points (grid points), $\mathbf{D} = 2D/(b - a)$, and $\mathbf{Z} = [z(\tau_0), z(\tau_1), \dots, z(\tau_N)]^T$ is the vector function at the collocation points. Higher order derivatives are obtained as powers of \mathbf{D} , that is

$$Z_j^{(p)} = \mathbf{D}^p Z_j, \quad (6)$$

where p is the order of the derivative.

2.2. Spectral Quasilinearization Method (SQLM). In this section, we describe a quasilinearization method that does not use the decoupling approach described in the previous section. This quasilinearization method (QLM) is a generalisation of the Newton-Raphson method and was first proposed by Bellman and Kalaba [28] for solving nonlinear boundary value problems. To obtain the QLM iteration scheme the nonlinear component of a differential equation is linearized using the multivariable Taylor series expansion as opposed to the local linearization approach used in the previous section. The QLM scheme is solved using the Chebyshev pseudospectral method as described in the previous section. Consequently, the method is referred to as the spectral quasilinearization method (SQLM) in this work.

To develop the SQLM scheme we consider a system of m nonlinear ordinary differential equations in m unknowns functions $z_i(\eta)$ $i = 1, 2, \dots, m$, where η is the dependent variable. The system can be written as a sum of its linear \mathcal{L} and nonlinear components \mathcal{N} as

$$\mathcal{L}[z_1(\eta), z_2(\eta), \dots, z_m(\eta)] + \mathcal{N}[z_1(\eta), z_2(\eta), \dots, z_m(\eta)] = 0, \quad (\eta) \in (a, b), \quad (7)$$

subject to the separated boundary conditions

$$\begin{aligned} A_i [z_1(a), z_2(a), \dots, z_m(a)] &= K_{a,i}, \\ B_b [z_1(b), z_2(b), \dots, z_m(b)] &= K_{b,i}, \end{aligned} \quad (8)$$

where A_i and B_i are linear operators and $K_{a,i}$ and $K_{b,i}$ are constants for $i = 1, 2, \dots, m$. Define the vector Z_i to be the vector of the derivatives of the variable z_i with respect to the dependent variable η , that is,

$$Z_i = [z_i^{(0)}, z_i^{(1)}, \dots, z_i^{(n_i)}], \quad (9)$$

where $z_i^{(0)} = z_i$, $z_i^{(p)}$ is the p th derivative of z_i with respect to η and n_i ($i = 1, 2, \dots, m$) is the highest derivative order of the variable z_i appearing in the system of equations. In addition, we define L_i and N_i to be the linear and nonlinear operators, respectively, that operate on the Z_i for $i = 1, 2, \dots, m$. With these definitions, (7) and (8) can be written as

$$\begin{aligned} L_i [Z_1, Z_2, \dots, Z_m] + N_i [Z_1, Z_2, \dots, Z_m] \\ = \sum_{j=1}^m \sum_{p=0}^{n_j} \alpha_{i,j}^{[p]} z_j^{(p)} + N_i [Z_1, Z_2, \dots, Z_m] = 0, \end{aligned} \quad (10)$$

where $\alpha_{i,j}^{[p]}$ are the constant coefficients of $z_j^{(p)}$, and the derivative of z_j ($j = 1, 2, \dots, m$) that appears in the i th equation for $i = 1, 2, \dots, m$.

Noting that, for each variable z_i , the derivatives in the boundary conditions can at most be one less than the highest derivative of z_i in the governing system (7), we define the vector \tilde{Z}_i to be the vector of the derivatives of the variable z_i with respect to the dependent variable η from 0 up to $n_i - 1$, that is,

$$\tilde{Z}_i = [z_i^{(0)}, z_i^{(1)}, \dots, z_i^{(n_i-1)}]. \quad (11)$$

The boundary conditions (8) can be written as

$$\begin{aligned} A_\nu [\tilde{Z}_1(a), \tilde{Z}_2(a), \dots, \tilde{Z}_m(a)] \\ = \sum_{j=1}^m \sum_{p=0}^{n_j-1} \beta_{\nu,j}^{[p]} z_j^{(p)}(a) = K_{a,\nu}, \quad \nu = 1, 2, \dots, m_a, \\ B_\sigma [\tilde{Z}_1(b), \tilde{Z}_2(b), \dots, \tilde{Z}_m(b)] \\ = \sum_{j=1}^m \sum_{p=0}^{n_j-1} \gamma_{\sigma,j}^{[p]} z_j^{(p)}(b) = K_{b,\sigma}, \quad \sigma = 1, 2, \dots, m_b, \end{aligned} \quad (12)$$

where $\beta_{\nu,j}^{[p]}$ ($\gamma_{\sigma,j}^{[p]}$) are the constant coefficients of $z_j^{(p)}$ in the boundary conditions and m_a, m_b are the total number of prescribed boundary conditions at $x = a$ and $x = b$, respectively. We remark that the sum $m_a + m_b$ is equal to the sum of the highest orders of the derivatives corresponding to the dependent variables z_i , that is

$$m_a + m_b = \sum_{i=1}^m n_i. \quad (13)$$

Assume that the solution $z_i(\eta)$ of (10) at the $(r + 1)$ th iteration is $z_{i,r+1}$. If the solution at the previous iteration $z_{i,r}(\eta)$ is sufficiently close to $z_{i,r+1}$, the nonlinear component N_i of

(10) can be linearized using the one-term Taylor series for multiple variables so that (10) can be approximated as

$$L_i [Z_{1,r+1}, \dots, Z_{m,r+1}] + N_i [\dots] + \sum_{j=1}^m \sum_{p=0}^{n_j} (z_{j,r+1}^{(p)} - z_{j,r}^{(p)}) \frac{\partial N_i}{\partial z_j^{(p)}} [\dots] = 0, \tag{14}$$

subject to

$$\begin{aligned} \sum_{j=1}^m \sum_{p=0}^{n_j-1} \beta_{\nu,j}^{[p]} z_{j,r+1}^{(p)} (a) &= 0, \\ \nu &= 1, 2, \dots, m_a, \\ \sum_{j=1}^m \sum_{p=0}^{n_j-1} \gamma_{\sigma,j}^{[p]} z_{j,r+1}^{(p)} (b) &= 0, \\ \sigma &= 1, 2, \dots, m_b, \end{aligned} \tag{15}$$

where

$$[\dots] = [Z_{1,r}, Z_{2,r}, \dots, Z_{m,r}]. \tag{16}$$

Equation (14) can be rewritten as

$$\begin{aligned} L_i [Z_{1,r+1}, \dots, Z_{m,r+1}] + \sum_{j=1}^m \sum_{p=0}^{n_j} z_{j,r+1}^{(p)} \frac{\partial N_i}{\partial z_j^{(p)}} [\dots] \\ = \sum_{j=1}^m \sum_{p=0}^{n_j} z_{j,r}^{(p)} \frac{\partial N_i}{\partial z_j^{(p)}} [\dots] - N_i [\dots]. \end{aligned} \tag{17}$$

The initial approximation, $z_{j,0}(\eta)$, required to start the iteration scheme (17) is chosen to be a function that satisfies the boundary conditions (8). Applying the Chebyshev spectral collocation on the recursive iteration scheme (17) gives

$$\sum_{j=1}^m [\Lambda_{i,j} + \Pi_{i,j}] \mathbf{Z}_{j,r+1} = \Phi_{i,r}, \quad i, j = 1, 2, \dots, m, \tag{18}$$

where $\mathbf{Z}_{i,r+1} = [z_{i,r+1}(\tau_0), z_{i,r+1}(\tau_1), \dots, z_{i,r+1}(\tau_N)]^T$, $\Lambda_{i,j}$, $\Pi_{i,j}$ and Φ_i are given by

$$\begin{aligned} \Lambda_{i,j} &= \sum_{p=0}^{n_j} \alpha_{i,j}^p \mathbf{D}^p, & \Pi_{i,j} &= \sum_{p=0}^{n_j} \left[\frac{\partial N_i}{\partial z_j^{(p)}} \right]_d \mathbf{D}^p, \\ & & i, j &= 1, 2, \dots, m, \end{aligned} \tag{19}$$

$$\begin{aligned} \Phi_{i,r} &= \sum_{j=1}^m \sum_{p=0}^{n_j} z_{j,r}^{(p)} \frac{\partial N_i}{\partial z_j^{(p)}} [\dots] \\ &\quad - N_i [Z_{1,r}, Z_{2,r}, \dots, Z_{m,r}], \end{aligned}$$

respectively, where $[\dots]_d$ means denote diagonal matrix. Defining $\Delta = \Lambda + \Pi$, we can write (18) in matrix form as

$$\begin{aligned} \begin{bmatrix} \Delta_{1,1} & \Delta_{1,2} & \dots & \Delta_{1,m} \\ \Delta_{2,1} & \Delta_{2,2} & \dots & \Delta_{2,m} \\ \vdots & \vdots & & \vdots \\ \Delta_{m,1} & \Delta_{m,2} & \dots & \Delta_{m,m} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_{1,r+1} \\ \mathbf{Z}_{2,r+1} \\ \vdots \\ \mathbf{Z}_{m,r+1} \end{bmatrix} \\ = \begin{bmatrix} \Phi_{1,r} \\ \Phi_{2,r} \\ \vdots \\ \Phi_{m,r} \end{bmatrix}, \end{aligned} \tag{20}$$

where $\mathbf{Z}_{i,r}$, $\Phi_{i,r}$ are vectors of size $(\bar{N} + 1) \times 1$ and $\Delta_{i,j}$ are $(\bar{N} + 1) \times (\bar{N} + 1)$ matrices. Starting from $Z_{i,0}$, the recursive sequence (20) is solved iteratively for $r = 0, 1, 2, 3, \dots$

Thus, the size of the coefficient matrix in (20) is $m(\bar{N} + 1) \times m(\bar{N} + 1)$ and the column vector on the right-hand side has the dimension $m(\bar{N} + 1) \times 1$.

3. Numerical Experiments

In this section, we discuss the implementation of the SLLM and SQLM approaches on two examples of boundary layer flow systems. We consider the Blasius boundary layer problem and a three-equation system that models the problem of unsteady free convective heat and mass transfer on a stretching surface in a porous medium in the presence of a chemical reaction.

3.1. Blasius Boundary Layer Equation. The Blasius boundary layer flow equation, in a dimensionless form, is expressed in terms of the similarity variable $\eta \in [0, \infty)$ and function $f(\eta)$ as

$$\begin{aligned} f'''(\eta) + \frac{1}{2} f(\eta) f''(\eta) &= 0, & f(0) &= 0, \\ f'(0) &= 0, & f'(\infty) &= 1, \end{aligned} \tag{21}$$

where the primes denote differentiation with respect to η . To apply the proposed SLLM iterative method on (21), it is convenient to reduce the order of the governing equations by one through setting $f'(\eta) = g(\eta)$. This results in the following system:

$$f' = g, \quad f(0) = 0, \tag{22}$$

$$g'' + \frac{1}{2} f(\eta) g'(\eta) = 0, \quad g(0) = 0, \quad g(\infty) = 1. \tag{23}$$

We observe that (23) is essentially a differential equation whose unknown is $g(\eta)$, but the equation requires $f(\eta)$ which can be considered to be a known input function. Thus, assuming that $f(\eta)$ is known at a particular iteration level,

the local linearization iteration scheme can be expressed by the following iteration formula:

$$\begin{aligned} g''_{r+1} + \frac{1}{2}f_r g'_{r+1} &= 0, & g_{r+1}(0) &= 0, \\ g_{r+1}(\infty) &= 1, \\ f'_{r+1} &= g_{r+1}, & f_{r+1}(0) &= 0. \end{aligned} \tag{24}$$

Applying the Chebyshev pseudospectral method on (24), we obtain the following decoupled system of equations:

$$\begin{aligned} A_1 \mathbf{g}_{r+1} &= B_1, & g_{r+1}(\tau_{\bar{N}}) &= 0, \\ g_{r+1}(\tau_0) &= 1, \\ \mathbf{D} \mathbf{f}_{r+1} &= \mathbf{g}_{r+1}, & f_{r+1}(\tau_{\bar{N}}) &= 0, \end{aligned} \tag{25}$$

where

$$A_1 = \mathbf{D}^2 + \frac{1}{2} \text{diag}[\mathbf{f}_r] \mathbf{D}, \quad B_1 = \mathbf{O}_1, \tag{26}$$

where \mathbf{O}_1 is an $(\bar{N} + 1) \times 1$ vector on which the boundary conditions are imposed and \mathbf{f} and \mathbf{g} are the values of functions f and g , respectively, when evaluated at the collocation points. Equation (25) constitutes the SLLM iteration scheme.

The initial approximation required to start the iteration process can be chosen as a function that satisfies the governing boundary conditions and from known physical considerations of the flow properties. For most boundary layer flow problems it is well known that the flow velocity tends to the mainstream flow exponentially. Thus, a suitable initial approximation for the boundary layer problem (21) is

$$f_0(\eta) = \eta - 1 + e^{-\eta}, \quad g_0(\eta) = 1 - e^{-\eta}. \tag{27}$$

To obtain the SQLM iterative scheme for the Blasius problem (21), we set

$$\begin{aligned} L &= f''', & N &= \frac{1}{2}ff'', \\ [f, f', f'', f'''] &= [z, z^{(1)}, z^{(2)}, z^{(3)}]. \end{aligned} \tag{28}$$

Thus, using (17), we obtain

$$f''_{r+1} + \frac{1}{2}f_r f''_{r+1} + \frac{1}{2}f''_r f_{r+1} = \frac{1}{2}f''_r f_r, \tag{29}$$

subject to

$$f_{r+1}(0) = 0, \quad f'_{r+1}(0) = 0, \quad f'_{r+1}(\infty) = 1. \tag{30}$$

Applying the spectral method, with derivative matrices on the iteration scheme (29) and the corresponding boundary conditions, gives the following matrix system:

$$\mathbf{A}_r \mathbf{F}_{r+1} = \Phi_r, \tag{31}$$

with the boundary conditions

$$\begin{aligned} f_{r+1}(\tau_N) &= 0, & \sum_{k=0}^{\bar{N}} \mathbf{D}_{\bar{N}k} f_{r+1}(\tau_k) &= 0, \\ \sum_{k=0}^{\bar{N}} \mathbf{D}_{0k} f_{r+1}(\tau_k) &= 1, \end{aligned} \tag{32}$$

where

$$\begin{aligned} \mathbf{A}_r &= \mathbf{D}^3 + \frac{1}{2} \text{diag}(\mathbf{f}_r) \mathbf{D}^2 + \frac{1}{2} \text{diag}(\mathbf{f}'_r), \\ \Phi_r &= \frac{1}{2} \mathbf{f}_r \mathbf{f}''_r. \end{aligned} \tag{33}$$

3.2. Unsteady Free Convective Heat and Mass Transfer on a Stretching Surface in a Porous Medium with Suction/Injection.

In this section we consider a three-equation system that models the problem of unsteady free convective heat and mass transfer on a stretching surface in a porous medium in the presence of a chemical reaction. The governing equations [29, 30] for this problem are given as the following dimensionless system of equations and boundary conditions:

$$\begin{aligned} f'''' + ff'' - (f')^2 - Kf' - A \left(f' + \frac{\eta}{2} f'' \right) \\ + Gr\theta + Gc\phi &= 0, \\ \frac{1}{Pr} \theta'' - f'\theta + f\theta' - A \left(\theta + \frac{1}{2} \eta \theta' \right) &= 0, \end{aligned} \tag{34}$$

$$\begin{aligned} \frac{1}{Sc} \phi'' - f'\phi + f\phi' - A \left(\phi + \frac{1}{2} \eta \phi' \right) - \gamma\phi &= 0, \\ f(0) = f_w, & f'(0) = 1, \\ \theta(0) = 1, & \phi(0) = 1, \end{aligned} \tag{35}$$

$$f'(\infty) = 0, \quad \theta(\infty) = 0, \quad \phi(\infty) = 0,$$

where $f(\eta)$, $\theta(\eta)$, and $\phi(\eta)$ are, respectively, the dimensionless velocity, temperature, and concentration, f_w is the suction/injection parameter, γ is the chemical reaction constant, Pr is the Prandtl number, Sc is the Schmidt number, K is the permeability parameter, and Gr and Gc are the temperature and concentration dependent Grashof numbers, respectively.

To apply the proposed SLLM iterative method on (34), we set $f'(\eta) = g(\eta)$ to obtain

$$f' = g, \tag{36}$$

$$\begin{aligned} g'' + fg'' - g^2 - Kg - A \left(g + \frac{\eta}{2} g' \right) \\ + Gr\theta + Gc\phi &= 0, \end{aligned} \tag{37}$$

$$\frac{1}{Pr} \theta'' - g\theta + f\theta' - A \left(\theta + \frac{1}{2} \eta \theta' \right) = 0, \tag{38}$$

$$\frac{1}{Sc} \phi'' - g\phi + f\phi' - A \left(\phi + \frac{1}{2} \eta \phi' \right) - \gamma\phi = 0. \tag{39}$$

We observe that the energy equation (38) requires only f and g to resolve the solution of θ . The same applies to the mass transfer equation (39) for ϕ . Equation (37), on the other hand, requires the solutions of f , θ and ϕ . In developing the SLLM iteration scheme we solve (38) and (39) for θ , and ϕ first, then use their updated solutions in the (37) for g . Thus, assuming that $f(\eta)$ and $g(\eta)$ are known at a particular iteration level, the local linearization iteration algorithm can be expressed as

$$\begin{aligned} \frac{1}{\text{Pr}} \theta''_{r+1} - g_r \theta_{r+1} + f_r \theta'_{r+1} - A \left(\theta_{r+1} + \frac{\eta}{2} \theta'_{r+1} \right) &= 0, \\ \frac{1}{\text{Sc}} \phi''_{r+1} - g_r \phi_{r+1} + f_r \phi'_{r+1} \\ - (A + \gamma) \phi_{r+1} - A \frac{\eta}{2} \phi'_{r+1} &= 0, \\ g''_{r+1} + f_r g'_{r+1} - (A + K) g_{r+1} + \frac{\eta A}{2} g'_{r+1} \\ - 2g_r g_{r+1} + \text{Gr} \theta_{r+1} + \text{Gc} \phi_{r+1} &= -g_r^2, \\ f'_{r+1} = g_{r+1}, \quad f_{r+1}(0) = f_w, \end{aligned} \quad (40)$$

subject to

$$\begin{aligned} g_{r+1}(0) = 1, \quad g_{r+1}(\infty) = 0, \\ \theta_{r+1}(0) = 1, \quad \theta_{r+1}(\infty) = 0, \\ \phi_{r+1}(0) = 1, \quad \phi_{r+1}(\infty) = 0. \end{aligned} \quad (41)$$

Applying the Chebyshev pseudospectral method on (40)-(41), we obtain the following decoupled matrix system of equations:

$$\begin{aligned} A_1 \Theta_{r+1} = B_1, \quad \theta_{r+1}(\tau_{\bar{N}}) = 1, \quad \theta_{r+1}(\tau_0) = 0, \\ A_2 \Phi_{r+1} = B_2, \quad \phi_{r+1}(\tau_{\bar{N}}) = 1, \quad \phi_{r+1}(\tau_0) = 0, \\ A_3 \mathbf{g}_{r+1} = B_3 - \text{Gr} \Theta_{r+1} - \text{Gc} \Phi_{r+1}, \quad g_{r+1}(\tau_{\bar{N}}) = 1, \\ g_{r+1}(\tau_0) = 0, \\ \mathbf{D} \mathbf{f}_{r+1} = \mathbf{g}_{r+1}, \quad f_{r+1}(\tau_{\bar{N}}) = f_w, \end{aligned} \quad (42)$$

where

$$\begin{aligned} A_1 &= \frac{1}{\text{Pr}} \mathbf{D}^2 + \text{diag} \left[\mathbf{f}_r - A \frac{\eta}{2} \right] \mathbf{D} - \text{diag} [\mathbf{g}_r] - A \mathbf{I}, \\ B_1 &= \mathbf{O}_1, \\ A_2 &= \frac{1}{\text{Sc}} \mathbf{D}^2 + \text{diag} \left[\mathbf{f}_r - A \frac{\eta}{2} \right] \mathbf{D} - \text{diag} [\mathbf{g}_r] - (A + \gamma) \mathbf{I}, \\ B_2 &= \mathbf{O}_1, \\ A_3 &= \mathbf{D}^2 + \text{diag} \left[\mathbf{f}_r - A \frac{\eta}{2} \right] \mathbf{D} - \text{diag} [2\mathbf{g}_r] - (K + A) \mathbf{I}, \\ B_3 &= -\mathbf{g}_r^2, \end{aligned} \quad (43)$$

where \mathbf{I} is an $(\bar{N} + 1) \times (\bar{N} + 1)$ identity matrix, \mathbf{O}_1 is an $(\bar{N} + 1) \times 1$ vector, and Θ , Φ , \mathbf{g} , and \mathbf{f} are the approximate values of θ , ϕ , g , and f evaluated at the collocation points. After incorporating the boundary conditions in the matrices in (42), the approximate solutions of θ , ϕ , g , and f at each iteration level can be obtained by solving (42) in the order in which they are listed. A suitable initial approximation for the SLLM scheme is

$$\begin{aligned} f_0(\eta) = f_w + 1 - e^{-\eta}, \quad \theta_0(\eta) = e^{-\eta}, \\ \phi_0(\eta) = e^{-\eta}. \end{aligned} \quad (44)$$

To obtain the SQLM iterative scheme for the system (34), we set

$$\begin{aligned} L_1 &= f''' - A \left(f' + \frac{\eta}{2} f'' \right) + \text{Gr} \theta + \text{Gc} \phi, \\ N_1 &= f f'' - (f')^2, \\ L_2 &= \frac{1}{\text{Pr}} \theta'' - A \left(\theta + \frac{1}{2} \eta \theta' \right), \quad N_2 = -f' \theta + f \theta', \\ L_3 &= \frac{1}{\text{Sc}} \phi'' - A \left(\phi + \frac{1}{2} \eta \phi' \right) - \gamma \phi, \quad N_3 = -f' \phi + f \phi'. \end{aligned} \quad (45)$$

To use formula (17) we define the following variables:

$$\begin{aligned} [f, f', f'', f'''] &= [z_1, z_1^{(1)}, z_1^{(2)}, z_1^{(3)}], \\ [\theta, \theta', \theta''] &= [z_2, z_2^{(1)}, z_2^{(2)}], \\ [\phi, \phi', \phi''] &= [z_3, z_3^{(1)}, z_3^{(2)}]. \end{aligned} \quad (46)$$

Thus, using (45)-(46) in (17), we obtain

$$\begin{aligned} f'''_{r+1} + \left(f_r - \frac{A\eta}{2} \right) f''_{r+1} - (2f'_r + A) f'_{r+1} \\ + f''_r f_{r+1} + \text{Gr} \theta_{r+1} + \text{Gc} \phi_{r+1} \\ = f_r f''_r - f_r'^2, \\ \frac{1}{\text{Pr}} \theta''_{r+1} + \left(f_r - \frac{A\eta}{2} \right) \theta'_{r+1} - (f'_r + A) \theta_{r+1} \\ - \theta_r f'_{r+1} + \theta'_r f_{r+1} = f_r \theta'_r - \theta_r f'_r, \\ \frac{1}{\text{Sc}} \phi''_{r+1} + \left(f_r - \frac{A\eta}{2} \right) \phi'_{r+1} - (f'_r + A + \gamma) \phi_{r+1} \\ - \phi_r f'_{r+1} + \phi'_r f_{r+1} = f_r \phi'_r - \phi_r f'_r, \\ f_{r+1}(0) = f_w, \quad f'_{r+1}(0) = 1, \\ \theta_{r+1}(0) = 1, \quad \phi_{r+1}(0) = 1, \\ f'_{r+1}(\infty) = 0, \quad \theta_{r+1}(\infty) = 0, \\ \phi_{r+1}(\infty) = 0. \end{aligned} \quad (48)$$

Applying the Chebyshev pseudospectral method on (47) we obtain the following SQLM scheme in a matrix form:

$$\begin{bmatrix} \Delta_{1,1} & \Delta_{1,2} & \Delta_{1,3} \\ \Delta_{2,1} & \Delta_{2,2} & \Delta_{2,3} \\ \Delta_{3,1} & \Delta_{3,2} & \Delta_{3,3} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_{1,r+1} \\ \mathbf{Z}_{2,r+1} \\ \mathbf{Z}_{3,r+1} \end{bmatrix} = \begin{bmatrix} \Phi_{1,r} \\ \Phi_{2,r} \\ \Phi_{3,r} \end{bmatrix} \quad (49)$$

subject to the boundary conditions

$$f_{r+1}(\tau_N) = 0, \quad \sum_{k=0}^{\bar{N}} \mathbf{D}_{\bar{N}k} f_{r+1}(\tau_k) = 0, \quad \sum_{k=0}^{\bar{N}} \mathbf{D}_{0k} f_{r+1}(\tau_k) = 1, \quad (50)$$

$$\theta_{r+1}(\tau_{\bar{N}}) = 1, \quad \theta_{r+1}(\tau_0) = 0,$$

$$\phi_{r+1}(\tau_{\bar{N}}) = 1, \quad \phi_{r+1}(\tau_0) = 0,$$

where

$$\begin{aligned} \Delta_{1,1} &= \Lambda_{1,1} + \Pi_{1,1} = \mathbf{D}^3 - \frac{A\eta}{2} \mathbf{D}^2 \\ &\quad - (K + A) \mathbf{D} + [\mathbf{f}_r'']_d - [2\mathbf{f}_r']_d \mathbf{D} + [\mathbf{f}_r]_d \mathbf{D}^2, \\ \Delta_{1,2} &= \Lambda_{1,2} + \Pi_{1,2} = \mathbf{G} \mathbf{r} \mathbf{I}, \\ \Delta_{1,3} &= \Lambda_{1,3} + \Pi_{1,3} = \mathbf{G} \mathbf{c} \mathbf{I}, \\ \Delta_{2,1} &= \Lambda_{2,1} + \Pi_{2,1} = [\Theta_r']_d - [\Theta_r]_d \mathbf{D}, \\ \Delta_{2,2} &= \Lambda_{2,2} + \Pi_{2,2} = \frac{1}{\text{Pr}} \mathbf{D}^2 \\ &\quad - \frac{A\eta}{2} \mathbf{D} - \mathbf{A} \mathbf{I} + [\mathbf{f}_r]_d \mathbf{D} - [\mathbf{f}_r']_d, \\ \Delta_{2,3} &= \Lambda_{2,3} + \Pi_{2,3} = \mathbf{O}_1, \\ \Delta_{3,1} &= \Lambda_{3,1} + \Pi_{3,1} = [\Phi_r']_d - [\Phi_r]_d \mathbf{D}, \\ \Delta_{3,2} &= \Lambda_{3,2} + \Pi_{3,2} = \mathbf{O}_1, \\ \Delta_{3,3} &= \Lambda_{3,3} + \Pi_{3,3} = \frac{1}{\text{Sc}} \mathbf{D}^2 - \frac{A\eta}{2} \mathbf{D} \\ &\quad - (A + \gamma) \mathbf{I} + [\mathbf{f}_r]_d \mathbf{D} - [\mathbf{f}_r']_d, \\ \Phi_{1,r} &= \mathbf{f}_r \mathbf{f}_r'' - \mathbf{f}_r'^2, \quad \Phi_{2,r} = \mathbf{f}_r \Theta_r' - \mathbf{f}_r' \Theta_r, \\ \Phi_{3,r} &= \mathbf{f}_r \Phi_r' - \mathbf{f}_r' \Phi_r, \\ \mathbf{Z}_{1,r} &= \mathbf{f}_r, \quad \mathbf{Z}_{2,r} = \Theta_r, \quad \mathbf{Z}_{3,r} = \Phi_r, \end{aligned} \quad (51)$$

and $[\]_d$ denotes $\text{diag}(\)$; that is, the vector elements are placed on the main diagonal of a matrix whose entries everywhere else are zero. Starting from the initial approximations (44), the approximate SQLM solutions for $f(\eta)$, $\theta(\eta)$, and $\phi(\eta)$ are obtained by solving (49).

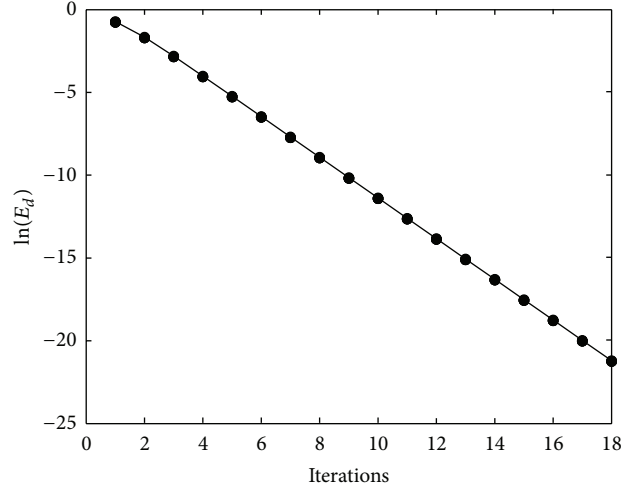


FIGURE 1: Comparison of Logarithm of error E_d for the Blasius boundary layer computed using the SLLM without SOR, SLLM with SOR ($\omega = 1.2$), and SQLM results.

4. Numerical Convergence, Error, and Stability Analysis of the Iteration Schemes

The convergence and stability of the iteration schemes can be assessed by considering the norm of the difference in the values of functions between two successive iterations. Thus, for each iteration scheme, we define the following maximum error (E_d) at the $(r + 1)$ th iteration:

$$E_d = \text{Max} \left(\|z_{1,r+1} - z_{1,r}\|_{\infty}, \|z_{2,r+1} - z_{2,r}\|_{\infty}, \dots, \|z_{m,r+1} - z_{m,r}\|_{\infty} \right), \quad (52)$$

where z_i ; $i = 1, \dots, m$ are the governing unknown functions in the nonlinear system. If the iteration scheme converges, the error E_d is expected to decrease with an increase in the number of iterations. In this paper, the unknowns were calculated, for a given number of collocation points \bar{N} , until the following criteria for convergence was fulfilled at iteration r :

$$E_d \leq \varepsilon, \quad (53)$$

where ε is the convergence tolerance level. In this study, the convergence tolerance is set to be $\varepsilon = 10^{-9}$. The effect of the number of collocation points \bar{N} was examined in order to select the smallest value of \bar{N} which gives a consistent solution to the ε error level. This is achieved by repeatedly solving the governing equations using the proposed iteration schemes with different values of \bar{N} until the consistent solution is reached.

5. Improving the Convergence of the Spectral Local Linearization Method (SLLM)

In numerical linear algebra, successive overrelaxation (SOR) methods are used to accelerate the convergence rates of the

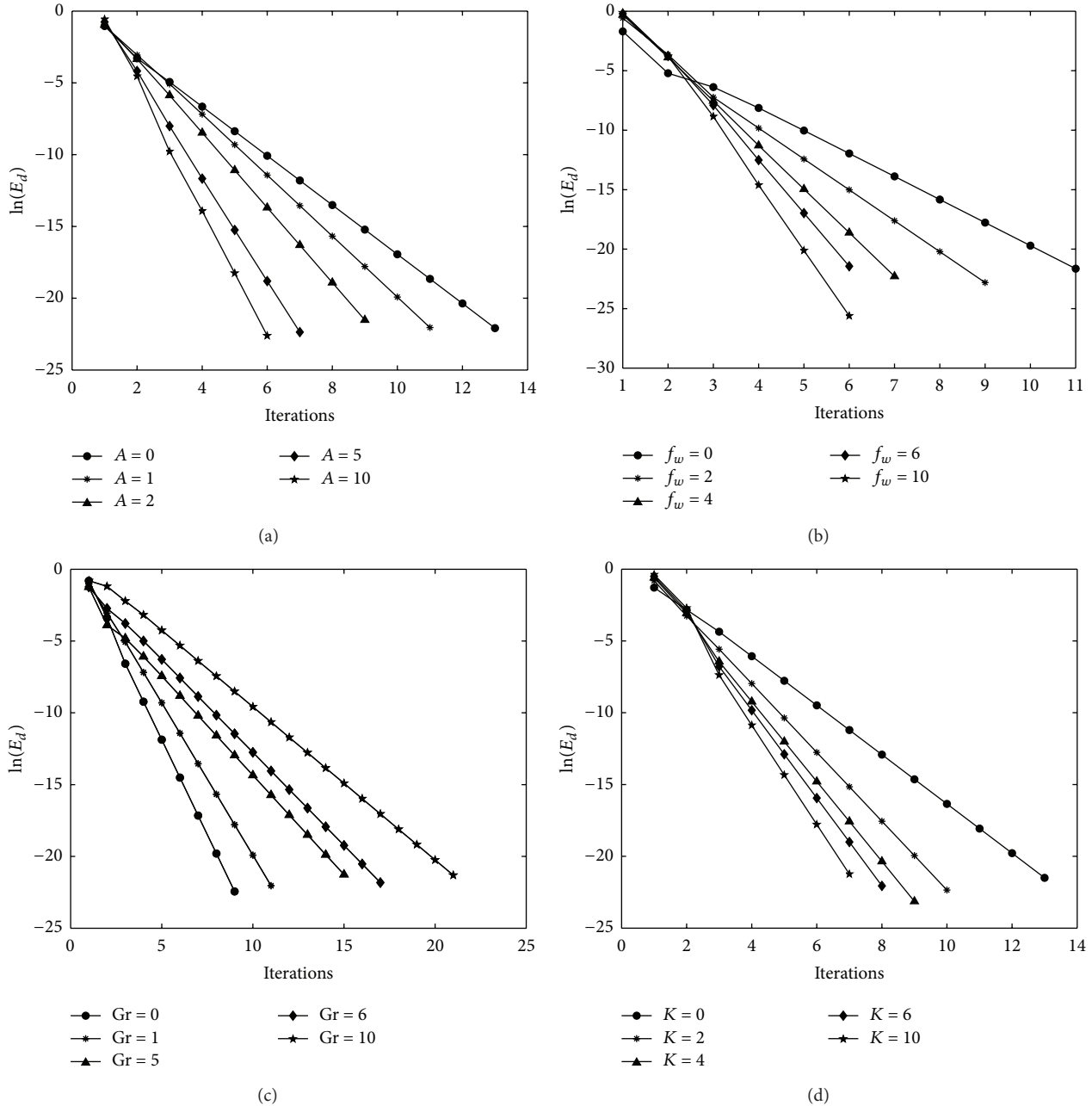


FIGURE 2: Effect of varying parameters on the Logarithm of the SLLM error.

Gauss-Seidel method in the solution of the linear systems of equation. In this section we propose a similar method to improve the convergence of the spectral relaxation method (SLLM) described in the previous section. If the SLLM scheme for solving the function X at the $(r + 1)$ th iteration is

$$AX_{r+1} = B, \tag{54}$$

then the modified version of the SLLM is defined as

$$AX_{r+1} = (1 - \omega) AX_r + \omega B, \tag{55}$$

where A, B are matrices and ω is the convergence controlling relaxation parameter. We observe that when $\omega = 1$ (55) reduces to the original SLLM method. The results in the next section will show that, for certain values of ω , solving the modified SLLM results in improved efficiency and accuracy.

6. Results and Discussion

We solved the governing systems of (21) and (34) we solved using the spectral local linearization method (SLLM) and spectral quasilinearization method (SQLM) as described in the previous section. In this section, we present the results

TABLE 1: Comparison between the basic SLLM computed values of $f''(0)$ for the Blasius flow against the results of [31, 32].

Iter.	
1	0.403512726
2	0.350208935
3	0.337194863
4	0.333547347
5	0.332492283
6	0.332184532
7	0.332094553
8	0.332068227
9	0.332060524
10	0.332058269
11	0.332057609
12	0.332057416
13	0.332057360
14	0.332057343
15	0.332057338
16	0.332057337
17	0.332057336
18	0.332057336
20	0.33205733621949529
25	0.33205733621520553
26	0.33205733621519900
28	0.33205733621519653
30	0.33205733621519632
31	0.33205733621519630
32	0.33205733621519630
⋮	
(52)	0.33205733621519630 (see [31]) 0.3320573362151965 (see [32])

of the numerical computations for the velocity, temperature, and concentration profiles for various input parameters. Values of other flow properties such as skin friction $f''(0)$, surface heat transfer rate at the $\theta'(0)$, and mass transfer rate at the wall $\phi'(0)$ are also discussed in order to illustrate some special features of the solution and to compare the accuracy, convergence, and efficiency of the SQLM and SLLM algorithms proposed in this study. The accuracy of the present results was verified by comparing them with other results from literature which have been reported to be accurate to within a certain number of decimal digits. Further validation of the solution was established by comparing the present results with the numerical solutions obtained using the MATLAB inbuilt routine `bvp4c` which is a finite difference code that implements the three-stage Lobatto III formula [33]. In the calculations presented here, the values of the governing physical parameters were chosen deliberately to match some of the results from published literature in order to enable effective comparison.

In Figure 1 we give a comparison of the SLLM and SQLM for the plot of the Logarithm of the error E_d against the number of iterations in the Blasius boundary layer problem. It can be seen from the figure that the spectral

TABLE 2: Comparison between the SOR improved SLLM ($\omega = 1.2$) values of $f''(0)$ for the Blasius flow against the results of [31, 32].

1	0.284215262
2	0.350088883
3	0.329960091
4	0.332654418
5	0.331973067
6	0.332078099
7	0.332054014
8	0.332058085
9	0.332057206
10	0.332057364
11	0.332057331
12	0.332057337
13	0.332057336
14	0.332057336
15	0.332057336
16	0.332057336
17	0.332057336
18	0.332057336
19	0.33205733621518464
20	0.33205733621519862
21	0.33205733621519584
22	0.33205733621519639
23	0.33205733621519628
24	0.33205733621519630
25	0.33205733621519630
⋮	
(52)	0.33205733621519630 (see [31]) 0.3320573362151965 (see [32])

TABLE 3: SQLM computed results for the skin friction $f''(0)$ of the Blasius flow.

Iter.	$f''(0)$
1	0.36124527510805664
2	0.33293906079206191
3	0.33205878995514977
4	0.33205733621994973
5	0.33205733621519630
6	0.33205733621519630
⋮	
(52)	0.33205733621519630 (see [31]) 0.3320573362151965 (see [32])

quasi-linearization method converges much faster than the spectral local linearization method. The fast convergence rate of the quasilinearization method was established by many researchers including [9, 10, 34–36] who extended the application of the quasilinearization method to a wide variety of nonlinear boundary value problems and established that the method converges quadratically. The convergence rate of the proposed SLLM is seen to be linear in this case. The effect of introducing successive overrelaxation with $\omega = 1.2$ has the beneficial effect of increasing the convergence speed which, in

TABLE 4: Effect of varying \bar{N} on $f''(0)$ and $\theta'(0)$ using the SLLM, when $K = 1$, $Gr = Gc = A = 1$, and $f_w = 2, \gamma = 0$.

Iter.	\bar{N}	$f''(0)$	$\theta'(0)$
(SLLM)			
8	30	-2.31812851	-2.62227419
8	40	-2.31812936	-2.62228212
8	50	-2.31812942	-2.62228134
8	100	-2.31812942	-2.62228134
8	150	-2.31812942	-2.62228134
8	200	-2.31812942	-2.62228134
(SQLM)			
5	30	-2.31691806	-2.62229791
5	40	-2.31816177	-2.62228177
5	50	-2.31813338	-2.62228131
5	100	-2.31812953	-2.62228134
100	150	-2.31811874	-2.62228134
100	200	-2.31821552	-2.62228134
bvp4c		-2.31812942	-2.62228134
[30]		-2.3181294	-2.6222813

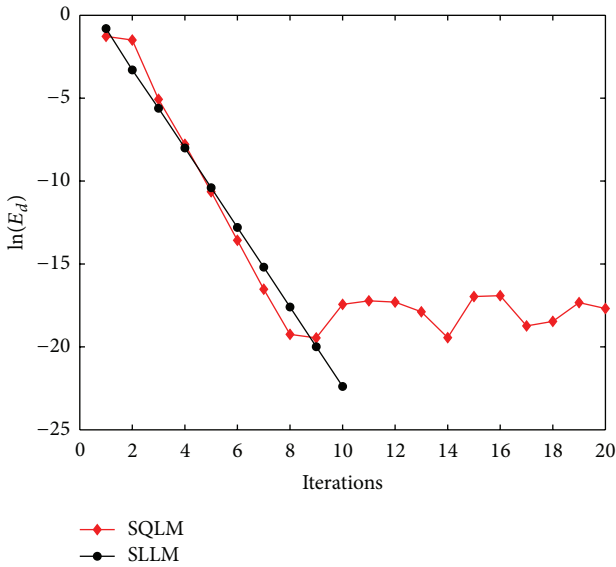


FIGURE 3: Comparison of the Logarithm of the error E_d for the unsteady boundary layer problem between the SLLM and SQLM results when $\bar{N} = 150$.

this case, can be characterized by the steeper line. To obtain the optimal value of ω that yields the best accuracy, the results obtained using several values of $\omega > 1$ and $\omega < 1$ near $\omega = 1$ were compared. It was found that underrelaxation ($\omega < 1$) slows down the convergence, whereas overrelaxation with $1 < \omega \leq 1.2$ improves the convergence of the SLLM iteration scheme. The best results for the Blasius boundary layer problem were obtained with $\omega \approx 1.2$. We remark that the optimal ω was obtained through numerical experimentation.

To further illustrate how the solution converges with an increase in iterations, we give the SLLM computed results for the skin friction $f''(0)$ in Tables 1 and 2 (with SOR).

The results are compared with the recently reported sixteen decimal digit accurate results of [31, 32]. Ganapol [32] used an algorithm based on a Maclaurin series with a Wynn-epsilon convergence acceleration and analytical continuation to obtain highly accurate skin friction coefficients for the Blasius layer flows. It can be seen from Table 1 that full convergence to 9-digit accurate results is achieved after 17 iterations. The 17-digit accurate results of [31] are matched exactly after 31 iterations. We remark that this value was reached after 52 iterations as reported in the work of [32]. This shows that the basic SLLM is more efficient than the method used in [32].

Table 2 gives the SOR-improved SLLM results (with $\omega = 1.2$) for the skin friction at different iterations. The table indicates that 9-digit accurate results are obtained after only 13 iterations and 17-digit accurate results are reached after 24 iterations. This clearly shows that using the SOR improves the convergence and efficiency of the basic SLLM.

Table 3 gives the SQLM results for the skin friction of the Blasius flow. It can be seen that the SQLM converges very rapidly compared to the SLLM in this example. Full convergence to the 17-digit accurate results reported in [31] is achieved after only 5 iterations. We remark that the results presented in Tables 1–3 were generated using $\bar{N} = 120$ and $\eta_\infty = 16$. Thus, under the same conditions, the SQLM is more efficient than the SLLM in this case.

The SLLM Log-error plots for the problem of unsteady free convective heat and mass transfer on a stretching surface are given in Figure 2 for varying input parameters. It can be seen that the Logarithm of the error strictly decreases with an increase in the number of iterations when all the parameters are varied. This demonstrates the convergence and stability of the SLLM in solving the three-equation coupled system (34) for varying physical constants. It can also be seen that the convergence speed increases with an increase in A, f_w , and K but decreases with an increase in Gr .

In Table 4 we give a comparison of the values of $f''(0)$ and $\theta'(0)$ computed using the SLLM and SQLM approaches at fixed values of the governing input parameters. The table shows the effects of the number of the collocation points \bar{N} on the convergence with respect to the benchmark bvp4c numerical results. We remark that the stopping criteria used in both methods are given by (52) where the running of the computations is terminated when the maximum norm of the difference between the values of the governing functions at two successive iterations is less than the prescribed tolerance level (set to be 10^{-9} in this work). For the selected values of \bar{N} , the SLLM converged after 8 iterations. Increasing \bar{N} improved the accuracy, and the bvp4c results were matched exactly when $\bar{N} \geq 50$ for both $f''(0)$ and $\theta'(0)$. The SQLM algorithm was found converge after only 5 iterations when $N \leq 100$, but the results did not convergent to the bvp4c results. Increasing the number of collocation points slightly improved the accuracy but full convergence to eight decimal digits was not achieved. Further increase in \bar{N} led to the poor accuracy and convergence. This can be seen in the results corresponding to $\bar{N} \geq 150$ where convergence was not reached even after 100 iterations. These

TABLE 5: Effect of varying A , f_w , Gr , and K on $f''(0)$ and $\theta'(0)$, when $Pr = Sc = \gamma = Gc = 1$.

A	f_w	Gr	K	Basic SLLM			SLLM with SOR (ω)			
				Iter.	$f''(0)$	$\theta'(0)$	ω	Iter.	$f''(0)$	$\theta'(0)$
0	1	1	1	11	-1.31052731	-1.65084042	0.95	7	-1.31052731	-1.65084042
1	1	1	1	10	-1.59363827	-1.89854697	0.95	7	-1.59363827	-1.89854697
2	1	1	1	8	-1.85550919	-2.12936888	0.95	7	-1.85550919	-2.12936888
5	1	1	1	6	-2.49344366	-2.70569628	1.00	6	-2.49344366	-2.70569628
10	1	1	1	6	-3.27938346	-3.44160771	1.00	6	-3.27938346	-3.44160771
1	0	1	1	10	-1.03558899	-1.34595467	0.95	7	-1.03558899	-1.34595467
1	2	1	1	8	-2.34393108	-2.61971104	0.95	6	-2.34393108	-2.61971104
1	4	1	1	6	-4.15359239	-4.35571036	1.00	6	-4.15359239	-4.35571036
1	6	1	1	6	-6.09385439	-6.24419885	1.00	6	-6.09385439	-6.24419885
1	10	1	1	5	-10.05256264	-10.14874633	1.00	5	-10.05256264	-10.14874633
1	1	0	1	8	-1.91528421	-1.87004601	1.00	8	-1.91528421	-1.87004601
1	1	1	1	10	-1.59363827	-1.89854697	0.95	7	-1.59363827	-1.89854697
1	1	5	1	13	-0.41099838	-1.98803325	0.90	9	-0.41099838	-1.98803325
1	1	6	1	15	-0.13374591	-2.00648856	0.90	9	-0.13374591	-2.00648856
1	1	10	1	19	0.92495170	-2.07084699	0.90	10	0.92495170	-2.07084699
1	1	1	0	12	-1.21175687	-1.93465038	0.95	8	-1.21175687	-1.93465038
1	1	1	2	9	-1.90764681	-1.87149682	0.95	7	-1.90764681	-1.87149682
1	1	1	4	8	-2.42217698	-1.83197529	0.95	7	-2.42217698	-1.83197529
1	1	1	6	7	-2.84650750	-1.80348579	0.95	7	-2.84650750	-1.80348579
1	1	1	10	7	-3.54376838	-1.76361252	0.95	7	-3.54376838	-1.76361252

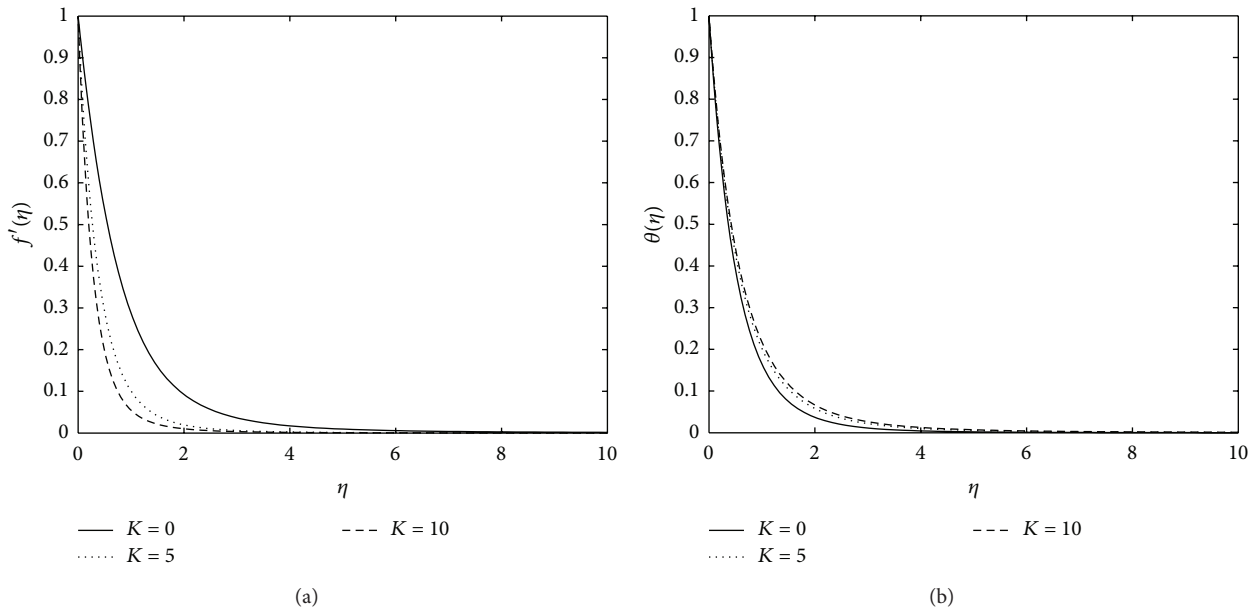


FIGURE 4: Effect of K on the velocity and temperature profiles.

results indicate that the SQLM is not ideal for solving very large systems of equations. The poor accuracy of the SQLM is caused by the numerical difficulties associated with solving large matrix equations. The advantage of using the SLLM in solving large coupled systems of differential equations is that the SLLM algorithm leads to decoupled systems of sub-matrix equations which can easily be solved using standard

direct methods in a sequential manner. With the SQLM the coupled system leads to a coupled iteration scheme which after discretization, using the spectral method, leads to large matrices. The dimension of the coefficient matrices and vectors to be solved in the SQLM algorithm is $m(\bar{N} \times 1) \times m(\bar{N} \times 1)$, where m is the number of unknowns in the governing systems. For example, in the case when $\bar{N} = 200$,

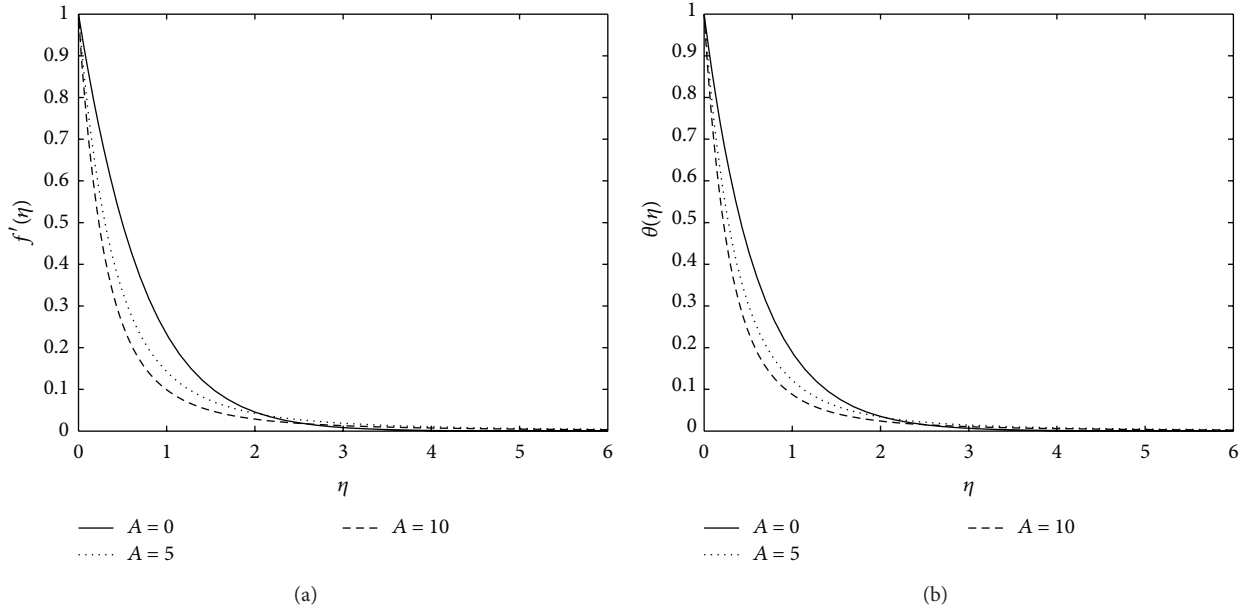


FIGURE 5: Effect of A on the velocity and temperature profiles.

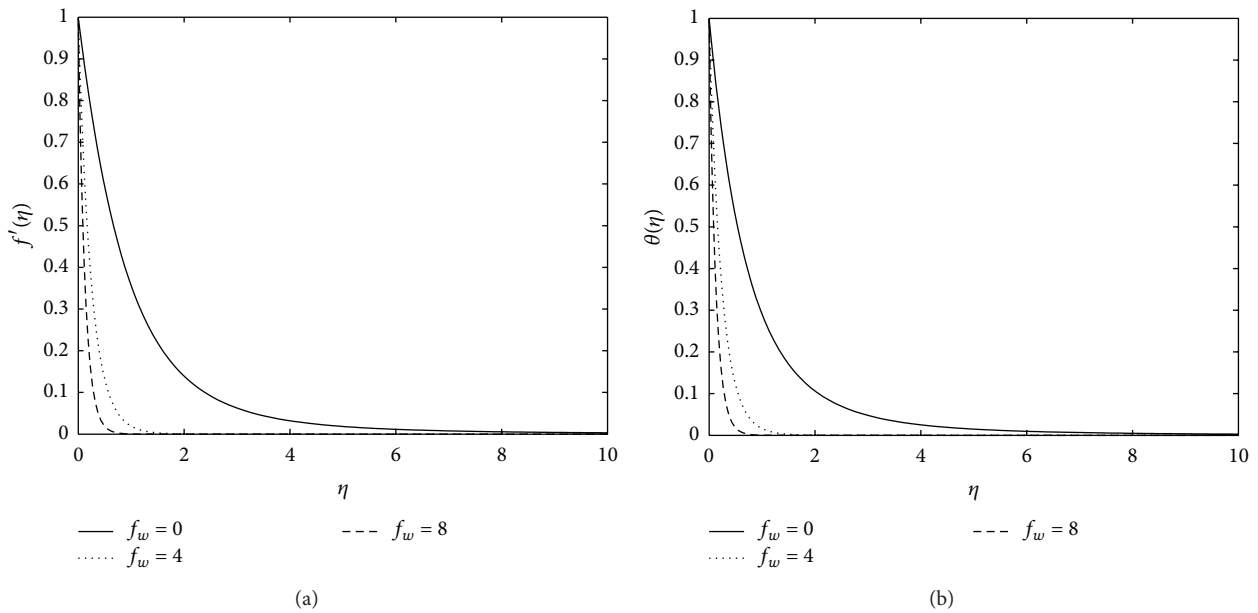


FIGURE 6: Effect of f_w on the velocity and temperature profiles.

the dimension of the coefficient matrix used in the solution for f , θ , and ϕ is 603×603 in the SQLM, compared to 201×201 in the case of the SLLM. Furthermore, the displayed accuracy and stability of the SLLM results for both small and large values of \bar{N} suggest that the efficiency of SLLM is due more to the algorithm behind the iteration scheme than to the specific numerical method used to solve the underlying differential equations of the scheme. The SQLM accuracy and stability are strongly linked to the numerical scheme used to solve the differential equations of the scheme. The Chebyshev differentiation matrix is known to be very

susceptible to very large round off errors and ill-conditioning for large numbers of collocation points \bar{N} (see e.g., [27, 37]), that is why the SQLM results in Table 4 seem to be less accurate when \bar{N} becomes very large. Table 4 clearly indicates that the SLLM is more practical to use for large systems of equations than SQLM as it yields more accurate results using only few collocation points, and there is no loss of accuracy when the side of the mesh is refined (when \bar{N} becomes large).

In Figure 3 we give a comparison of the Log-error against iterations plot (E_d) between the SLLM and SQLM results for a fixed number of collocation points. From Figure 3 it can

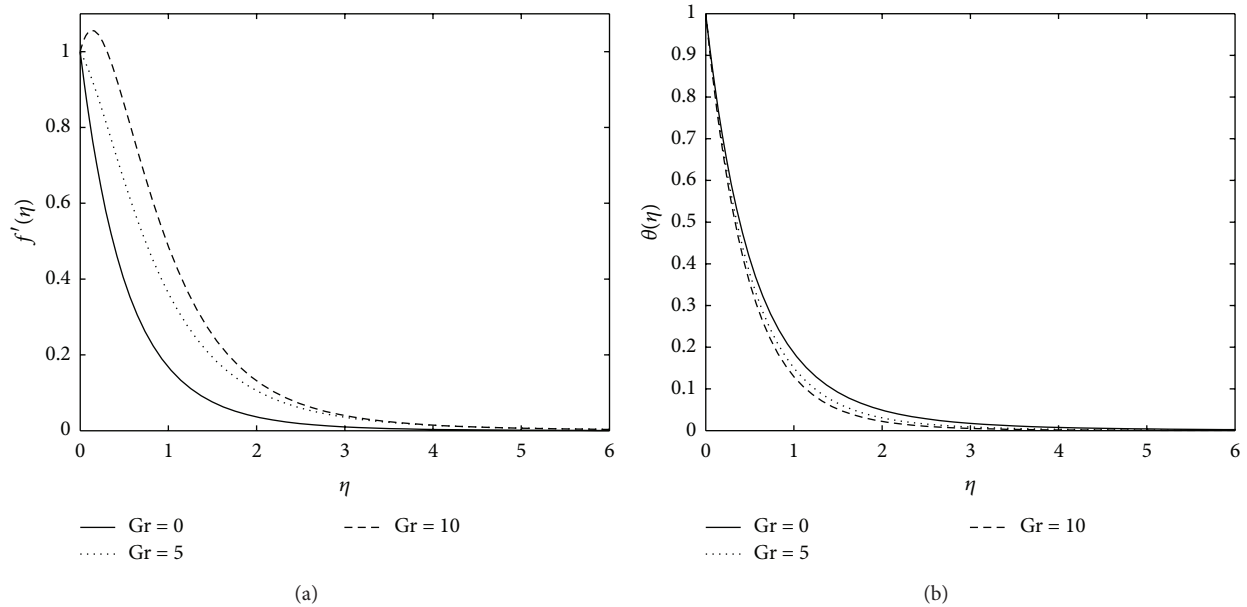


FIGURE 7: Effect of Gr on the velocity and temperature profiles.

be seen that the SQLM rapidly converges but the accuracy does not improve after 5 iterations. After 5 iterations the error plot seems to plateau at a certain level and remains at that level even when considering more iterations. The SLLM results indicate strict convergence with the error progressively reduced with an increase in the number of iterations. This explains why the SLLM gives more accurate results than the SQLM for larger systems as was observed in Table 4.

In Table 5 we give a comparison between the basic SLLM results and the results obtained using the SOR accelerated version of the SLLM for the skin friction $f''(0)$ and wall heat transfer rate $\theta'(0)$. It can be noted from Table 3 that introducing the convergence accelerating relaxation parameter ω leads to the improvement of the basic SLLM for some cases. It can be seen that convergence improves with an increase in f_w , A , and K and slows down with an increase in Gr . This result is consistent with the observation made in Figure 2. In cases where convergence is slow, underrelaxation with ($0.9 \leq \omega < 1$) accelerates the convergence. We also observe that the effect of increasing the values of Gr results in an increase in the skin friction and in the absolute values of the wall heat transfer rate. The wall heat transfer rate decreases in absolute value as K increases. An increase in the unsteadiness parameter A and the suction parameter f_w results in an increase in the skin friction and the wall heat transfer rate. We remark that the trends in the effect of the mass transfer rate at the wall were found to be similar to the effects observed in the surface heat transfer rate for all varied input parameters. The trends observed in Table 5 are consistent with the observations made in related studies [29, 30].

Figures 4, 5, 6, and 7 illustrate the effects of the input parameters K , A , f_w and Gr , respectively, on the temperature and concentration profiles. It is noted that the velocity

decreases with K , A , and f_w but increases with the buoyancy parameter Gr . The temperature decreases with A , f_w , and Gr but increases with an increase in K . Again, the observed trends are consistent with the observations reported in [29, 30]. The effect of the varied parameters on the concentration was found to be similar to the trends observed in the temperature profiles.

7. Conclusion

In this work we have introduced a new method for solving systems of nonlinear boundary value equations. The proposed method, called spectral local linearization method (SLLM), is based on a simple idea of the decoupling systems of equations using the linearization of the unknown functions in a sequential manner according to the order of the listing of the governing equations of the system. The linearized equations, which now form a sequence of linear differential equations with variable coefficients, were solved using the Chebyshev spectral collocation method. The applicability of the method was examined on similarity boundary layer problems with exponentially decaying profiles. The Blasius boundary layer problem and a three-equation coupled system that models the problem of unsteady, free convective heat and mass transfer were considered as test cases to examine the applicability of the proposed SLLM. The accuracy of the SLLM was confirmed against known accurate results reported in the published literature and against numerical results generated using Matlab's `bvp4c` routine for solving boundary value problems. The performance of the SLLM was measured against the spectral quasi-linearization method (SQLM) which uses the Chebyshev collocation method to discretize the quasilinearization method (a Newton-Raphson based iteration method for solving boundary value problem).

From this preliminary study, the following observations were made about the proposed SLLM.

- (1) The algorithm of the SLLM is very easy to develop and implement, as it is based on a simple univariate linearization of nonlinear functions.
- (2) The method is numerically efficient, since it results in a series of equations which are solved in a sequential manner by reusing the information from the solution of one equation in the next equation.
- (3) The method was found to converge rapidly to the expected solutions for all the input parameters considered in this study.
- (4) The convergence speed of the method can readily be improved by using successive overrelaxation (SRM) techniques. In the case of the Blasius boundary layer, overrelaxation improved convergence, and in the unsteady heat and mass transfer problem, the convergence was improved by underrelaxation
- (5) The convergence speed of the SLLM improves with an increase in the values of the parameters A , f_w , and K and slows down with an increase in Gr .
- (6) The method is more computationally efficient than the SQLM in the sense that it gives accurate results even with a few collocation points (small \bar{N}). The accuracy of the SQLM was seen to deteriorate when the number of collocation points became very large. This is due to the numerical difficulties associated with solving large matrix systems. Unlike the SQLM, the SLLM does not suffer from the loss of accuracy when the number of collocation points is large.

Because of its demonstrated accuracy, efficiency, and simplicity, it is envisaged that the proposed SLLM method could be used as a viable method for solving some classes of similarity variable boundary layer problems.

Acknowledgment

This work is based on the research supported in part by the National Research Foundation of South Africa (Grant no. 85596).

References

- [1] A. A. Ahmed, "Similarity solution in MHD: effects of thermal diffusion and diffusion thermo on free convective heat and mass transfer over a stretching surface considering suction or injection," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, pp. 2202–2214, 2009.
- [2] R. Cortell, "Numerical solutions of the classical Blasius flat-plate problem," *Applied Mathematics and Computation*, vol. 170, pp. 706–710, 2005.
- [3] A. Asaithambi, "A second-order finite-difference method for the Falkner-Skan equation," *Applied Mathematics and Computation*, vol. 156, no. 3, pp. 779–786, 2004.
- [4] H. B. Keller and T. Cebeci, "Accurate numerical methods for boundary layer flow-I. Two dimensional laminar flows," in *Proceedings of the 2nd International Conference on Numerical Methods in Fluid Dynamics*, pp. 92–100, Berkeley, Calif, USA, 1971.
- [5] T. Cebeci and P. Bradshaw, *Physical and Computational Aspects of Convective Heat Transfer*, Springer, New York, NY, USA, 1984.
- [6] S. S. Motsa and P. Sibanda, "On the solution of MHD flow over a nonlinear stretching sheet by an efficient semi-analytical technique," *International Journal for Numerical Methods in Fluids*, vol. 68, pp. 1524–1537, 2012.
- [7] S. S. Motsa, S. Shateyi, and P. Sibanda, "A model of steady viscous flow of a micropolar fluid driven by injection or suction between a porous disk and a non-porous disk using a novel numerical technique," *The Canadian Journal of Chemical Engineering*, vol. 88, no. 6, pp. 991–1002, 2010.
- [8] R. Sharma, R. Bhargava, and P. Bhargava, "A numerical solution of unsteady MHD convection heat and mass transfer past a semi-infinite vertical porous moving plate using element free galerkin method," *Computational Material Science*, vol. 48, pp. 537–543, 2010.
- [9] V. B. Mandelzweig and F. Tabakin, "Quasilinearization approach to nonlinear problems in physics with application to nonlinear ODEs," *Computer Physics Communications*, vol. 141, pp. 268–281, 2001.
- [10] S. Zhu, Q. Wu, and X. Cheng, "Numerical solution of the Falkner-Skan equation based on quasilinearization," *Applied Mathematics and Computation*, vol. 215, no. 7, pp. 2472–2485, 2009.
- [11] S. S. Motsa, T. Hayat, and O. M. Aldossary, "MHD flow of upper-convected Maxwell fluid over porous stretching sheet using successive Taylor series linearization method," *Applied Mathematics and Mechanics*, vol. 33, no. 8, pp. 975–990, 2012.
- [12] S. S. Motsa and S. Shateyi, "Successive linearization analysis of the effects of partial slip, thermal diffusion, and diffusion-thermo on steady MHD convective flow due to a rotating disk," *Mathematical Problems in Engineering*, vol. 2012, Article ID 397637, 15 pages, 2012.
- [13] F. G. Awad, P. Sibanda, S. S. Motsa, and O. D. Makinde, "Convection from an inverted cone in a porous medium with cross-diffusion effects," *Computers & Mathematics with Applications*, vol. 61, no. 5, pp. 1431–1441, 2011.
- [14] S.-J. Liao, "A uniformly valid analytic solution of two-dimensional viscous flow over a semi-infinite flat plate," *Journal of Fluid Mechanics*, vol. 385, pp. 101–128, 1999.
- [15] S. Abbasbandy and T. Hayat, "Solution of the MHD Falkner-Skan flow by homotopy analysis method," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 9–10, pp. 3591–3598, 2009.
- [16] S.-J. Liao, "An explicit, totally analytic approximate solution for Blasius' viscous flow problems," *International Journal of Non-Linear Mechanics*, vol. 34, no. 4, pp. 759–778, 1999.
- [17] A. Alizadeh-Pahlavan and S. Borjian-Boroujeni, "On the analytical solution of viscous fluid flow past a flat plate," *Physics Letters*, vol. 372, pp. 3678–3682, 2008.
- [18] J. H. He, "Approximate analytical solution of Blasius' equation," *Communications in Nonlinear Science and Numerical Simulation*, vol. 4, pp. 75–78, 1999.
- [19] J.-H. He, "A simple perturbation approach to Blasius equation," *Applied Mathematics and Computation*, vol. 140, no. 2–3, pp. 217–222, 2003.
- [20] B. Kuo, "Thermal boundary-layer problems in a semi-infinite flat plate by the differential transformation method," *Applied*

- Mathematics and Computation*, vol. 150, no. 2, pp. 303–320, 2004.
- [21] M. M. Rashidi, “The modified differential transform method for solving MHD boundary-layer equations,” *Computer Physics Communications*, vol. 180, no. 11, pp. 2210–2217, 2009.
- [22] L.-T. Yu and C.-K. Chen, “The solution of the Blasius equation by the differential transformation method,” *Mathematical and Computer Modelling*, vol. 28, no. 1, pp. 101–111, 1998.
- [23] C.-W. Chang, J.-R. Chang, and C.-S. Liu, “The lie-group shooting method for solving classical Blasius flat-plate problem,” *Computers, Materials, & Continua*, vol. 7, no. 3, pp. 139–153, 2008.
- [24] J. Lin, “A new approximate iteration solution of Blasius equation,” *Communications in Nonlinear Science & Numerical Simulation*, vol. 4, no. 2, pp. 91–99, 1999.
- [25] A. Wazwaz, “The variational iteration method for solving two forms of Blasius equation on a half-infinite domain,” *Applied Mathematics and Computation*, vol. 188, no. 1, pp. 485–491, 2007.
- [26] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer, Berlin, Germany, 1988.
- [27] L. N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, Pa, USA, 2000.
- [28] R. E. Bellman and R. E. Kalaba, *Quasilinearization and Nonlinear Boundary-Value Problems*, Elsevier, New York, NY, USA, 1965.
- [29] A. J. Chamkha, A. M. Aly, and M. A. Mansour, “Similarity solution for unsteady heat and mass transfer from a stretching surface embedded in a porous medium with suction/injection and chemical reaction effects,” *Chemical Engineering Communications*, vol. 197, pp. 846–858, 2010.
- [30] S. S. Motsa and S. Shateyi, “Successive linearisation analysis of unsteady heat and mass transfer from a stretching surface embedded in a porous medium with suction/injection and thermal radiation effect,” *The Canadian Journal of Chemical Engineering*, vol. 90, no. 5, pp. 1323–1335, 2011.
- [31] J. P. Boyd, “The Blasius function: computations before computers, the value of tricks, undergraduate projects, and open research problems,” *SIAM Review*, vol. 50, pp. 791–804, 2008.
- [32] B. D. Ganapol, “Highly accurate solutions of the Blasius and Falkner-Skan boundary layer equations via convergence acceleration,” <http://arxiv.org/abs/1006.3888>.
- [33] L. F. Shampine, M. W. Reichelt, and J. Kierzenka, “Solving boundary value problems for ordinary differential equations in MATLAB with `bvp4c`,” http://www.mathworks.com/bvp_tutorial.
- [34] R. Krivec and V. B. Mandelzweig, “Numerical investigation of quasilinearization method in quantum mechanics,” *Computer Physics Communications*, vol. 138, pp. 69–79, 2001.
- [35] V. B. Mandelzweig, “Quasilinearization method and its verification on exactly solvable models in quantum mechanics,” *Journal of Mathematical Physics*, vol. 40, no. 12, pp. 6266–6291, 1999.
- [36] V. B. Mandelzweig, “Quasilinearization method: nonperturbative approach to physical problems,” *Physics of Atomic Nuclei*, vol. 68, no. 7, pp. 1227–1258, 2005.
- [37] W. S. Don and A. Solomonoff, “Accuracy and speed in computing the chebyshev collocation derivative,” *SIAM Journal on Scientific Computing*, vol. 16, no. 6, pp. 1253–1268, 1995.