Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 497863, 13 pages
http://dx.doi.org/10.1155/2013/497863

Research Article

A Third-Order p-Laplacian Boundary Value Problem Solved by an SL$(3, \mathbb{R})$ Lie-Group Shooting Method

Chein-Shan Liu

Department of Civil Engineering, National Taiwan University, Taipei, Taiwan

Correspondence should be addressed to Chein-Shan Liu; liucs@ntu.edu.tw

Received 3 November 2012; Accepted 18 February 2013

Academic Editor: Ch Tsitouras

The boundary layer problem for power-law fluid can be recast to a third-order p-Laplacian boundary value problem (BVP). In this paper, we transform the third-order p-Laplacian into a new system which exhibits a Lie-symmetry SL$(3, \mathbb{R})$. Then, the closure property of the Lie-group is used to derive a linear transformation between the boundary values at two ends of a spatial interval. Hence, we can iteratively solve the missing left boundary conditions, which are determined by matching the right boundary conditions through a finer tuning of $r \in [0, 1]$. The present SL$(3, \mathbb{R})$ Lie-group shooting method is easily implemented and is efficient to tackle the multiple solutions of the third-order p-Laplacian. When the missing left boundary values can be determined accurately, we can apply the fourth-order Runge-Kutta (RK4) method to obtain a quite accurate numerical solution of the p-Laplacian.

1. Introduction

The power-law fluids have been called the Ostwald-de Waele fluids which have been well examined in the past several decades, because the constitutive equation of such a fluid not only gives a good expression for a large portion of the non-Newtonian fluids but also encompasses a Newtonian fluid as well. The theoretical boundary layer theory for power-law fluids was first investigated by Schowalter [1], and then Acrivos et al. [2] obtained a similarity solution. The experimental results that a significant drag reduction can be achieved by injecting fluid into the boundary layer motivated the investigations of the non-Newtonian boundary layer flows with injection or suction at the surface. Flows with suction or injection through a porous wall are of practical interest for cooling, delaying transition to turbulence, and the prevention of separation in an adverse pressure gradient.

The drag force inside the shear layer is a consequence of pressure distribution on the surface. Realizing the nature of this force by a mathematical modelling to predict the drag force and the associated behavior of fluid flow has been the focus of considerable research. The reason for the interest in the analysis of the boundary layer flows along solid surfaces is the possibility of applying the theory to the efficient design of supersonic and hypersonic flights. Besides, the mathematical model considered in the present research has importance in studying many problems of engineering, meteorology, and oceanography, for example, Howell et al. [3], Nachman and Callegari [4], Ozisik [5], Schlichting [6], Shu and Wilks [7], and Zheng and Zhang [8].

We assume that the moving flat plate is semi-infinite with a porous surface and that the plate is moving at a constant speed U_w in the direction parallel to an oncoming flow with a constant speed U_∞. By the assumption of incompressibility and the conservation of momentum, the laminar flow satisfies

$$\begin{gathered} \frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0, \\ U \frac{\partial U}{\partial X} + V \frac{\partial U}{\partial Y} = \frac{1}{\rho} \frac{\partial \tau_{XY}}{\partial Y}. \end{gathered} \tag{1}$$

In the above, X and Y are the coordinates attached to the plate in the horizontal and perpendicular directions, and U and V are, respectively, the velocity components of the flow in the X and Y directions. The fluid density ρ is assumed to be a constant.

The shear stress is governed by a power law

$$\tau_{XY} = K \left| \frac{\partial U}{\partial Y} \right|^{N-1} \frac{\partial U}{\partial Y}, \tag{2}$$

where $K > 0$ is a constant and the power $N > 0$ reflects the discrepancy to the Newtonian fluids (with $N = 1$). The case with $N < 1$ is the power law of pseudoplastic fluids, and $N > 1$ is the dilatant fluids. The corresponding boundary conditions are given by

$$U(X,0) = U_w, \qquad U(X,+\infty) = U_\infty, \qquad V(X,0) = V_w(X) = V_0 X^{-N/(N+1)}. \tag{3}$$

After introducing a similarity variable and a stream function

$$\eta = BX^\beta Y, \qquad \phi(X,Y) = AX^\sigma f(\eta) \tag{4}$$

with

$$\sigma = \frac{1}{N+1}, \qquad \beta = -\sigma, \qquad B = \left(\frac{\rho U_\infty^{2-N}}{(N+1)K}\right)^{1/(N+1)}, \qquad A = \frac{U_\infty}{B}, \tag{5}$$

we can obtain

$$\left(\left|f''(\eta)\right|^{N-1} f''(\eta)\right)' + f(\eta) f''(\eta) = 0, \tag{6}$$

which is subjected to the following boundary conditions:

$$f(0) = -C_0, \qquad f'(0) = \xi, \qquad f'(+\infty) = 1. \tag{7}$$

In the above, $\xi = U_w/U_\infty$ is the velocity ratio. When $\xi < 0$, we have a reverse flow attached near the boundary. When $0 < \xi < 1$, the speed of the oncoming fluid is larger than that of the plate. When $\xi > 1$, the speed of the moving plate is faster than the speed of the oncoming fluid. The term $C_0 = (N+1)BV_0/U_\infty$ is a constant related to the situation of suction if it is negative or injection if it is positive.

Previously, the author and his coworkers have developed the Lie-group shooting method based on the Lorentz-group [9, 10] to solve the boundary layer equations [11–14]. Liu et al. [15] have found the multiple solutions of boundary layer problem by an enhanced fictitious time integration method for the nonlinear algebraic equations discretized from the governing equation. In this paper, we propose a more simple and powerful Lie-group shooting method directly based on the three-dimensional special linear group $SL(3,\mathbb{R})$ to solve the $p(= N+1)$-Laplacian boundary layer equations with multiple solutions. Liu [16] used the $SL(2,\mathbb{R})$ Lie-group shooting method to solve the eigenvalue problem of the second-order Sturm-Liouville equation. Recently, Liu [17] has successfully applied the $SL(3,\mathbb{R})$ Lie-group shooting method to effectively solve the Falkner-Skan boundary layer problem. The present study is an extension of these researches.

The operator $(|f''(\eta)|^{N-1} f''(\eta))'$ in (6) is one sort of the third-order $p(= N+1)$-Laplacian [18]. Besides the non-Newtonian fluid [19–21], the p-Laplacian operator arises in some different physical and mathematical modelling of the combustion theory [22], population dynamics [23, 24], and also the Monge-Kantorovich partial differential equation [25]. There had been many papers for studying the second-order p-Laplacian about its existence of positive solutions. Only a few papers are devoted to the numerical solution of the third-order p-Laplacian.

This paper is organized as follows. In Section 2, we consider (6) as a special case of the third-order p-Laplacian, and then by a translation of the variable to a new variable with a positive value, we can transform the p-Laplacian into a new system, which exhibits a Lie-symmetry of $SL(3,\mathbb{R})$. We introduce some mathematical requirements of the Lie-group formulation of the resulting ODEs and construct a Lie-group shooting method based on the Lie-group $SL(3,\mathbb{R})$. The boundary conditions that the present $SL(3,\mathbb{R})$ Lie-group shooting method can be applied to are discussed. In Section 3, we test the performance of the newly developed $SL(3,\mathbb{R})$ Lie-group shooting method for several examples, and in Section 4, the multiple solutions of the boundary layer problem of power-law fluids are investigated. Finally, we draw conclusions in Section 5.

2. An $SL(3,\mathbb{R})$ Lie-Group Shooting Method

Equation (6) is a special case of the following p-Laplacian:

$$\left(\phi_p\left(y''(x)\right)\right)' = F\left(x, y(x), y'(x), y''(x)\right), \quad x \in \left(x_0, x_f\right), \tag{8}$$

of which the boundary conditions have many types to be discussed below. In the above, $\phi_p(s) = |s|^{p-2}s$, $p > 1$, $\phi_p^{-1} = \phi_q$, where $1/p + 1/q = 1$.

In order to develop an $SL(3,\mathbb{R})$ Lie-group shooting method, we suppose that $y > -\infty$, such that there exists a constant k_0 rendering

$$u(x) = y(x) + k_0 > 0, \quad \forall x \in \left[x_0, x_f\right]. \tag{9}$$

Then, (8) becomes

$$\left(\phi_p\left(u''(x)\right)\right)' = F\left(x, u(x) - k_0, u'(x), u''(x)\right) =: H\left(x, u(x), u'(x), u''(x)\right). \tag{10}$$

2.1. A Group-Preserving Scheme. Let

$$v(x) = \phi_p\left(u''(x)\right), \tag{11}$$

and thus by $\phi_p^{-1} = \phi_q$, we have

$$u''(x) = \phi_q(v(x)). \tag{12}$$

At the same time, from (10) and (11), it follows that

$$v'(x) = H\left(x, u(x), u'(x), \phi_q(v(x))\right) =: f\left(x, u(x), u'(x), v(x)\right). \tag{13}$$

Let $w(x) = u'(x)$, and we have $w'(x) = \phi_q(v(x))$ by (12). Now, let

$$u_1 = u, \qquad u_2 = w = u', \qquad u_3 = v. \tag{14}$$

The above equations $u'(x) = w(x)$, $w'(x) = \phi_q(v(x))$, and (13) can be written as a system of three first-order ordinary differential equations (ODEs) by

$$\frac{d}{dx}\begin{bmatrix} u_1(x) \\ u_2(x) \\ u_3(x) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{\phi_q(u_3(x))}{u_1} & 0 & 0 \\ \dfrac{f(x, u_1(x), u_2(x), u_3(x))}{u_1} & 0 & 0 \end{bmatrix} \times \begin{bmatrix} u_1(x) \\ u_2(x) \\ u_3(x) \end{bmatrix}, \tag{15}$$

which is well defined because of $u_1 = u > 0$ by (9).

The differential equations system (15) is highly nonlinear due to the appearance of $f(x, u_1, u_2, u_3)/u_1$ and $\phi_q(u_3(x))/u_1$ in the coefficient matrix; however, it allows a Lie-symmetry $SL(3, \mathbb{R})$:

$$\frac{d}{dx}\mathbf{G}(x) = \mathbf{A}(x)\mathbf{G}(x), \quad \mathbf{G}(x_0) = \mathbf{I}_3, \tag{16}$$

$$\mathbf{A}(x) = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{\phi_q}{u_1} & 0 & 0 \\ \dfrac{f}{u_1} & 0 & 0 \end{bmatrix}, \tag{17}$$

$$\det \mathbf{G}(x) = 1, \tag{18}$$

because of $\operatorname{tr}\mathbf{A} = 0$. Here, for saving notations, we use $f = f(x, u_1, u_2, u_3)$ and $\phi_q = \phi_q(u_3)$. The above $\mathbf{G}$ is a fundamental matrix of (15).

Accordingly, we can develop a group-preserving scheme (GPS) to solve (15)

$$\mathbf{U}_{k+1} = \mathbf{G}(k)\mathbf{U}_k, \tag{19}$$

where $\mathbf{U}_k$ denotes the numerical value of $\mathbf{U} = (u_1, u_2, u_3)^T$ at the discrete space x_k and $\mathbf{G}(k) \in SL(3, \mathbb{R})$. This Lie-symmetry is known as the *three-dimensional real-valued special linear group*, denoted by $SL(3, \mathbb{R})$.

2.2. A Generalized Midpoint Rule. Applying the GPS in (19) to (15) with an initial condition $\mathbf{U}(x_0) = \mathbf{U}_0$, we can find $\mathbf{U}(x)$. Assuming that the stepsize used in the GPS is $\Delta x = \ell/K$, where $\ell = x_f - x_0$, we can calculate the value of $\mathbf{U}$ at $x = x_f$ by

$$\mathbf{U}_f = \mathbf{G}_K(\Delta x)\cdots\mathbf{G}_1(\Delta x)\mathbf{U}_0. \tag{20}$$

Now, we prove the following closure property of the Lie-group $SL(3, \mathbb{R})$:

$$\mathbf{G}_1(x), \mathbf{G}_2(x) \in SL(3, \mathbb{R}) \Longrightarrow \mathbf{G}_2(x)\mathbf{G}_1(x) \in SL(3, \mathbb{R}). \tag{21}$$

By the assumptions of $\mathbf{G}_1(x) \in SL(3, \mathbb{R})$ and $\mathbf{G}_2(x) \in SL(3, \mathbb{R})$, we have $\det \mathbf{G}_1(x) = 1$ and $\det \mathbf{G}_2(x) = 1$. Then, by using the following result:

$$\det[\mathbf{G}_2(x)\mathbf{G}_1(x)] = \det \mathbf{G}_2(x) \det \mathbf{G}_1(x), \tag{22}$$

it is straightforward to verify that $\det[\mathbf{G}_2(x)\mathbf{G}_1(x)] = 1$, which means that $\mathbf{G}_2(x)\mathbf{G}_1(x) \in SL(3, \mathbb{R})$. Thus, we have proven (21).

Because each $\mathbf{G}_i$, $i = 1, \ldots, K$ in (20) is an element of the Lie-group $SL(3, \mathbb{R})$, and by the above closure property of the Lie-group $SL(3, \mathbb{R})$, $\mathbf{G}_K(\Delta x)\cdots\mathbf{G}_1(\Delta x)$ is also a Lie-group element of $SL(3, \mathbb{R})$, denoted by $\mathbf{G}$. Hence, we have

$$\mathbf{U}_f = \mathbf{G}\mathbf{U}_0. \tag{23}$$

This is a *one-step Lie-group transformation* from $\mathbf{U}_0$ to $\mathbf{U}_f$, acting by $\mathbf{G} \in SL(3, \mathbb{R})$.

However, it is very hard to obtain an exact solution of $\mathbf{G}$ because the differential equations system is highly nonlinear. Before the derivation of a suitable form for $\mathbf{G}$, let us recall the mean value theorem for a continuous function $f(x)$, which is defined in an interval of $x \in [a, b]$. The mean value theorem asserts that there exists at least one $c \in [a, b]$, such that the following equality holds

$$\int_a^b f(x)\,dx = f(c)[b - a], \tag{24}$$

where the value of c depends on the function $f(x)$. In terms of the weighting factor $r \in [0, 1]$, we can write $c = ra + (1 - r)b$. Therefore, it means that there exists at least one $r \in [0, 1]$, such that (24) is satisfied. The above theorem enables us to evaluate the value of the integral in (24) by an area of a rectangle with a width $b - a$ times a height $f(c)$, where $f(c)$ is calculated by a mid-point rule with a suitable $c \in [a, b]$.

Because $\mathbf{G}$ is a solution of (16), we can formally write it by an exponential mapping

$$\mathbf{G}(x) = \exp\left[\int_{x_0}^{x} \mathbf{A}(\xi)\,d\xi\right]. \tag{25}$$

When $\mathbf{A}$ is not a constant matrix, in general we do not have a closed-form solution of $\mathbf{G}(x)$. However, motivated by the above mean value theorem and to be a reasonable approximation, we can calculate $\mathbf{G}$ in (23) by a generalized mid-point rule, which is obtained from an exponential mapping of $\mathbf{A}$ by taking the values of the variables in $\mathbf{A}$ at a suitable mid-point: $\hat{x} = rx_0 + (1 - r)x_f$, where $r \in [0, 1]$ is an unknown constant to be determined by the shooting method. So we can compute this $\mathbf{G}$ by

$$\mathbf{G}(r) = \exp\left[\left(x_f - x_0\right)\hat{\mathbf{A}}\right], \tag{26}$$

which is corresponding to a constant matrix

$$\hat{\mathbf{A}} =: \begin{bmatrix} 0 & 1 & 0 \\ b & 0 & 0 \\ a & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{\hat{\phi}_q}{\hat{u}_1} & 0 & 0 \\ \dfrac{\hat{f}}{\hat{u}_1} & 0 & 0 \end{bmatrix}, \tag{27}$$

where $a = \hat{f}/\hat{u}_1$ and $b = \hat{\phi}_q/\hat{u}_1$ are supposed to be constant.

This $SL(3,\mathbb{R})$ Lie-group element generated from such a constant matrix $\widehat{\mathbf{A}} \in sl(3,\mathbb{R})$ has a closed-form solution. If $b > 0$ and $c = a/\sqrt{b}$, then we have

$$\mathbf{G}(r) = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} = \begin{bmatrix} \cosh\left(\ell\sqrt{b}\right) & \frac{1}{\sqrt{b}}\sinh\left(\ell\sqrt{b}\right) & 0 \\ \sqrt{b}\sinh\left(\ell\sqrt{b}\right) & \cosh\left(\ell\sqrt{b}\right) & 0 \\ c\sinh\left(\ell\sqrt{b}\right) & \frac{c}{\sqrt{b}}\left[\cosh\left(\ell\sqrt{b}\right)-1\right] & 1 \end{bmatrix}, \tag{28}$$

where $\ell = x_f - x_0$ denotes the length of the interval $[x_0, x_f]$.

If $b < 0$ and $c = a/\sqrt{-b}$, then we have

$$\mathbf{G}(r) = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} = \begin{bmatrix} \cos\left(\ell\sqrt{-b}\right) & \frac{1}{\sqrt{-b}}\sin\left(\ell\sqrt{-b}\right) & 0 \\ -\sqrt{-b}\sin\left(\ell\sqrt{-b}\right) & \cos\left(\ell\sqrt{-b}\right) & 0 \\ c\sin\left(\ell\sqrt{-b}\right) & \frac{c}{\sqrt{-b}}\left[1-\cos\left(\ell\sqrt{-b}\right)\right] & 1 \end{bmatrix}. \tag{29}$$

In the above, we have taken

$$\begin{gathered} \widehat{x} = r x_0 + (1-r)x_f, \\ \widehat{u}_1 = r u_1^0 + (1-r)u_1^f, \\ \widehat{u}_2 = r u_2^0 + (1-r)u_2^f, \\ \widehat{u}_3 = r u_3^0 + (1-r)u_3^f, \\ \widehat{f} := f(\widehat{x}, \widehat{u}_1, \widehat{u}_2, \widehat{u}_3), \\ \widehat{\phi}_q := \phi_q(\widehat{u}_3), \\ a = \frac{\widehat{f}}{\widehat{u}_1}, \qquad b = \frac{\widehat{\phi}_q}{\widehat{u}_1}, \qquad c = \frac{a}{\sqrt{|b|}}. \end{gathered} \tag{30}$$

For the special case of $b = 0$, we can derive

$$\mathbf{G}(r) = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} = \begin{bmatrix} 1 & \ell & 0 \\ 0 & 1 & 0 \\ a\ell & \frac{a\ell^2}{2} & 1 \end{bmatrix}. \tag{31}$$

2.3. Specification of Boundary Conditions. For the third-order ODEs, there are several different type boundary conditions. In this section, we study this problem that under what type boundary conditions the present $SL(3,\mathbb{R})$ Lie-group shooting method is applicable.

Let

$$\mathbf{U}_0 = \begin{bmatrix} u_1^0 \\ u_2^0 \\ u_3^0 \end{bmatrix}, \qquad \mathbf{U}_f = \begin{bmatrix} u_1^f \\ u_2^f \\ u_3^f \end{bmatrix} \tag{32}$$

denote, respectively, the left-end and right-end boundary values of $\mathbf{U} = (u_1, u_2, u_3)^T$. For linear type boundary conditions (separable or nonseparable), we can describe the boundary conditions by the following equation:

$$\mathbf{B}_0\mathbf{U}_0 + \mathbf{B}_f\mathbf{U}_f = \mathbf{b}, \tag{33}$$

where both $\mathbf{B}_0$ and $\mathbf{B}_f$ are 3×3 matrices and $\mathbf{b} \in \mathbb{R}^3$ is a constant vector, which might be zero.

Inserting (23) into (33), we have

$$\left[\mathbf{B}_0 + \mathbf{B}_f\mathbf{G}\right]\mathbf{U}_0 = \mathbf{b}, \tag{34}$$

such that for a nonempty solution of $\mathbf{U}_0$, we require

$$\det\left[\mathbf{B}_0 + \mathbf{B}_f\mathbf{G}\right] \neq 0. \tag{35}$$

It means that the matrix $\mathbf{B}_0 + \mathbf{B}_f\mathbf{G}$ must be invertible.

In order to demonstrate the above idea about the specification of the boundary conditions, of which the present $SL(3,\mathbb{R})$ Lie-group shooting method is applicable, let us take $p = N + 1$ and $F = -uu''$ and (6) under the following boundary conditions:

$$\begin{gathered} u(0) = -C_0, \qquad u'(0) = \xi, \qquad u'(\eta_\infty) = 1, \\ \text{that is, } u_1(0) = -C_0, \qquad u_2(0) = \xi, \qquad u_2(\eta_\infty) = 1, \end{gathered} \tag{36}$$

where we have replaced ∞ by a finite number η_∞. In terms of (33), we have

$$\mathbf{B}_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{B}_f = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} -C_0 \\ \xi \\ 1 \end{bmatrix}. \tag{37}$$

Then, by (28) or (29), we can obtain

$$\mathbf{B}_0 + \mathbf{B}_f\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} G_{11} & G_{12} & 0 \\ G_{21} & G_{22} & 0 \\ G_{31} & G_{32} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ G_{21} & G_{22} & 0 \end{bmatrix}. \tag{38}$$

Thus, the above matrix $\mathbf{B}_0 + \mathbf{B}_f\mathbf{G}$ is not invertible, because of $\det[\mathbf{B}_0 + \mathbf{B}_f\mathbf{G}] = 0$. So we can conclude that for the power-law fluid under the boundary conditions (36), the present method is not applicable. In Section 4, we will give another type approach.

Physically, we can specify that at a large distance from the boundary layer the shear stress is quite small. Then, instead of (36), we can specify

$$u(0) = -C_0, \quad u'(0) = \xi, \quad u''(\eta_\infty) = s_0,$$
$$\text{that is, } u_1(0) = -C_0, \quad u_2(0) = \xi, \quad u_3(\eta_\infty) = \phi_{N+1}(s_0), \tag{39}$$

where s_0 is a small number. In terms of (33), we have

$$\mathbf{B}_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{B}_f = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$
$$\mathbf{b} = \begin{bmatrix} -C_0 \\ \xi \\ \phi_{N+1}(s_0) \end{bmatrix}. \tag{40}$$

Then, by (28) or (29), we can obtain

$$\mathbf{B}_0 + \mathbf{B}_f\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} G_{11} & G_{12} & 0 \\ G_{21} & G_{22} & 0 \\ G_{31} & G_{32} & 1 \end{bmatrix}$$
$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ G_{31} & G_{32} & 1 \end{bmatrix}, \tag{41}$$

which is invertible, due to $\det[\mathbf{B}_0 + \mathbf{B}_f\mathbf{G}] = 1$. Then, the present method is applicable. Below, we discuss the Lie-group shooting solution for the boundary layer problem of the power-law fluid under the boundary conditions (39).

2.4. An SL(3,$\mathbb{R}$) *Lie-Group Shooting Method.* In order to demonstrate the application of the SL(3, $\mathbb{R}$) Lie-group shooting method to find the missing left boundary conditions, as a representative case, let us take $p = N + 1$ and $F = -yy''$, and then we can recover to (6)

$$\left(\left|y''(x)\right|^{N-1} y''(x)\right)' = -y(x)\,y''(x), \quad x \in (0, \eta_\infty), \tag{42}$$

which is subjected to the following boundary conditions:

$$y(0) = -C_0, \qquad y'(0) = \xi, \qquad y''(\eta_\infty) = s_0 = 0.001, \tag{43}$$

where C_0 and ξ are given constants, and we use a large value η_∞, say $\eta_\infty = 6$, to replace the last boundary condition in (6), and also $y'(\eta_\infty) = 1$ is changed to $y''(\eta_\infty) = s_0$ as just mentioned above.

The stepping technique developed for solving the initial value problem (IVP) requires both the initial conditions of $u_1 = u$, $u_2 = u'$, and $u_3 = v$ for the third-order ODEs. Starting from the initial values of u_1, u_2, and u_3, we can numerically integrate the following IVP step by step from $x = x_0$ to $x = x_f$:

$$\frac{d}{dx}\begin{bmatrix} u_1(x) \\ u_2(x) \\ u_3(x) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{\phi_q(u_3(x))}{u_1} & 0 & 0 \\ \dfrac{f(x, u_1(x), u_2(x), u_3(x))}{u_1} & 0 & 0 \end{bmatrix} \times \begin{bmatrix} u_1(x) \\ u_2(x) \\ u_3(x) \end{bmatrix}, \tag{44}$$

$$u_1(0) = u_1^0, \tag{45}$$

$$u_2(0) = u_2^0, \tag{46}$$

$$u_3(0) = u_3^0, \tag{47}$$

where some unknown initial values are to be found by the SL(3, $\mathbb{R}$) Lie-group shooting method.

In (43), $u_1^0 = -C_0$ and $u_2^0 = \xi$ are given, but u_3^0 is an unknown constant to be determined such that we can satisfy the target equation of $u_3^f = \phi_{N+1}(s_0)$. Starting from an initial guess of u_1^f, u_3^f, and u_3^0, we can solve the unknown initial value u_3^0 by the following iterative processes:

$$u_3^0 = \frac{u_3^f - G_{31}u_1^0 - G_{32}u_2^0}{G_{33}},$$
$$u_1^f = G_{11}u_1^0 + G_{12}u_2^0 + G_{13}u_3^0,$$
$$u_2^f = G_{21}u_1^0 + G_{22}u_2^0 + G_{23}u_3^0, \tag{48}$$

which are obtained from (23). Inserting the initially guessed values of u_1^f, u_3^f, and u_3^0 and the given values of $u_1^0 = -C_0$, $u_2^0 = \xi$, and $u_3^f = \phi_{N+1}(s_0)$ into (28) or (29) with a specified $r \in [0, 1]$, we can evaluate G_{ij}, $i, j = 1, 2, 3$, and then by (48), we can generate the new values of u_3^0, u_1^f, and u_2^f, until they are convergent. If the new values of u_3^0, u_1^f, and u_2^f converge to satisfy the following convergence criterion:

$$\left(\left[u_3^0(k+1) - u_3^0(k)\right]^2 + \left[u_1^f(k+1) - u_1^f(k)\right]^2 + \left[u_2^f(k+1) - u_2^f(k)\right]^2\right)^{1/2} \le \varepsilon, \tag{49}$$

then the iterations stop. Here, $u_3^0(k+1)$ and $u_3^0(k)$ denote, respectively, the $(k+1)$th and the kth iteration values of u_3^0. They are defined similarly for u_1^f and u_2^f.

For a trial r, we can calculate u_3^0 from the above equations by a few iterations and then numerically integrate (44) by the fourth-order Runge-Kutta method (RK4) from 0 to η_∞ and compare the end value of u_3^f with the exact one $u_2(\eta_\infty) = \phi_{N+1}(s_0)$, which is a target equation to be matched. Indeed, we need to find the root of the equation $u_3^f - \phi_{N+1}(s_0) = 0$, where u_3^f is a numerically integrated result, depending on r. It can be done in practice by adjusting the value of r to a point such that the curve of mismatching error is intersected with the zero line at that point.

3. Numerical Examples

Example 1. First, we consider the following p-Laplacian:

$$\begin{gathered}\left(\phi_p\left(y''(x)\right)\right)' = h(x) + y(x), \quad x \in (0,1), \\ y(0) = 0, \qquad y'(1) = \pi, \quad y''(0) = 0.\end{gathered} \tag{50}$$

We assume that the closed-form solution is $y(x) = -\sin(\pi x)$. Hence,

$$h(x) = \pi^3 (p-1)\left[\pi^2 \sin(\pi x)\right]^{p-2} \cos(\pi x) + \sin(\pi x). \tag{51}$$

We can use the following equations to iteratively solve the unknown initial value of u_2^0:

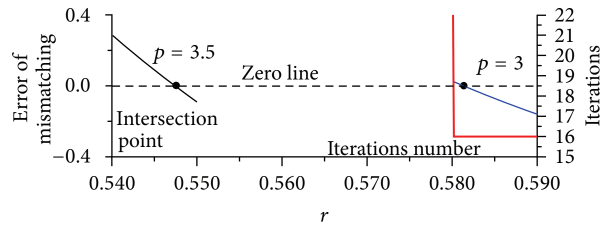
$$\begin{aligned} u_2^0 &= \frac{u_2^f - G_{21}u_1^0 - G_{23}u_3^0}{G_{22}}, \\ u_1^f &= G_{11}u_1^0 + G_{12}u_2^0 + G_{13}u_3^0, \\ u_3^f &= G_{31}u_1^0 + G_{32}u_2^0 + G_{33}u_3^0. \end{aligned} \tag{52}$$

We take $k_0 = 3.5$ for $p = 3$ and $k_0 = 4$ for $p = 3.5$ and use the $SL(3,\mathbb{R})$ Lie-group shooting method developed in Section 2.4 by adjusting the value of u_2^0. If the target equation $u'(1) = \pi$ is satisfied, then we obtain the numerical solution.

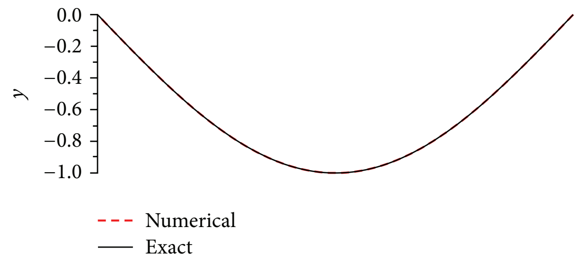
The convergence criterion is $\varepsilon = 10^{-8}$. Although under this stringent convergence criterion the iteration process to find u_2^0 is convergent very fast as shown in Figure 1(a), where for $r \in [0.58, 0.59]$ the iteration numbers are between 16 and 23 for the case $p = 3$. In Figure 1(a), we also plot the mismatching errors with respect to r in a range $[0.58, 0.59]$ for $p = 3$, while in a range $[0.54, 0.55]$ for $p = 3.5$. Both have an intersection point with the zero line. Then, through a finer tuning of the value to $r = 0.58135439224$ for the case $p = 3$ and to $r = 0.547436304748$ for the case $p = 3.5$, we can match the right-end boundary condition very precisely with an error in the order 10^{-11}. The numerical solutions of y and y' are, respectively, plotted in Figures 1(b) and 1(c), which are almost coincident with the closed-form solutions. Therefore, we plot the numerical errors, which are the absolute differences between exact solutions and numerical solutions, in Figure 2 for $p = 3$ and $p = 3.5$. It can be seen that the numerical results are quite accurate.

Example 2. Then, we consider a different case of (50) by

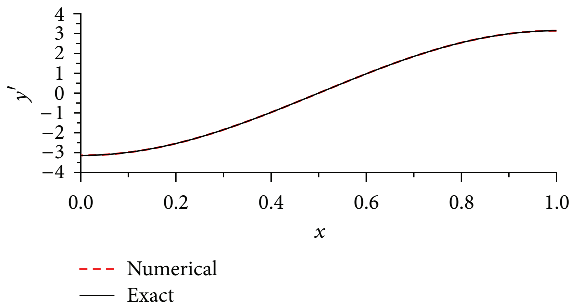
$$\begin{gathered}\left(\phi_p\left(y''(x)\right)\right)' = h(x) - y^2(x), \quad x \in (0,1), \\ y(0) = 0, \qquad y(1) = 0, \qquad y''(0) = 0.\end{gathered} \tag{53}$$



(a)



(b)



(c)

Figure 1: Example 1 solved by the $SL(3,\mathbb{R})$ method, (a) showing the error of mismatching and iterations number and comparing numerical solutions and exact solutions (b) of y and (c) y'.

Similarly, the closed-form solution is $y(x) = -\sin(\pi x)$. Hence,

$$h(x) = \pi^3 (p-1)\left[\pi^2 \sin(\pi x)\right]^{p-2} \cos(\pi x) + \sin^2(\pi x). \tag{54}$$

We can use the following equations to iteratively solve the unknown initial value of u_2^0:

$$\begin{aligned} u_2^0 &= \frac{u_1^f - G_{11}u_1^0 - G_{13}u_3^0}{G_{12}}, \\ u_2^f &= G_{21}u_1^0 + G_{22}u_2^0 + G_{23}u_3^0, \\ u_3^f &= G_{31}u_1^0 + G_{32}u_2^0 + G_{33}u_3^0. \end{aligned} \tag{55}$$

We take $k_0 = 3.5$ and $p = 3$. If the target equation $u(1) = k_0$ is satisfied, then we obtain the numerical solution. In Figure 3(a), we plot the mismatching error and the number of iterations with respect to r in a range $[0.575, 0.585]$. Then through a finer tuning of the value to $r = 0.5813782916$, we can match the right-end boundary condition very precisely

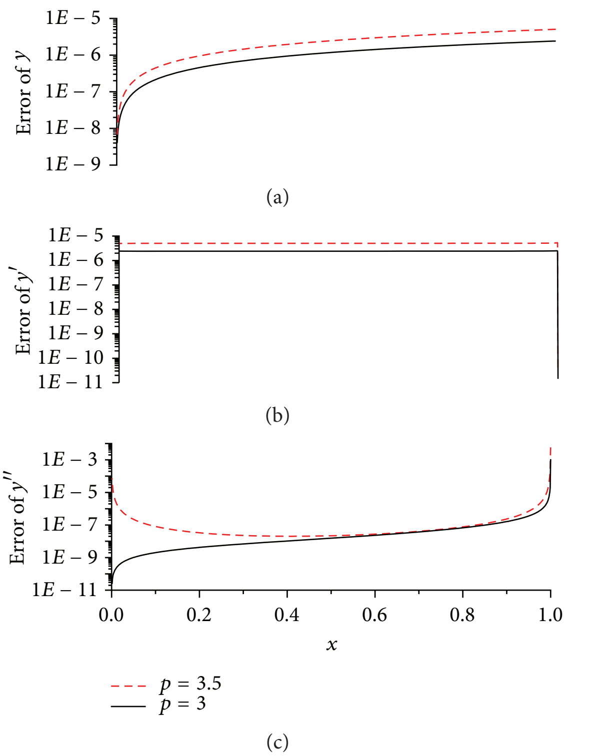


FIGURE 2: Example 1 with $p = 3$ and $p = 3.5$, showing the numerical errors of (a) y, (b) y', and (c) y''.

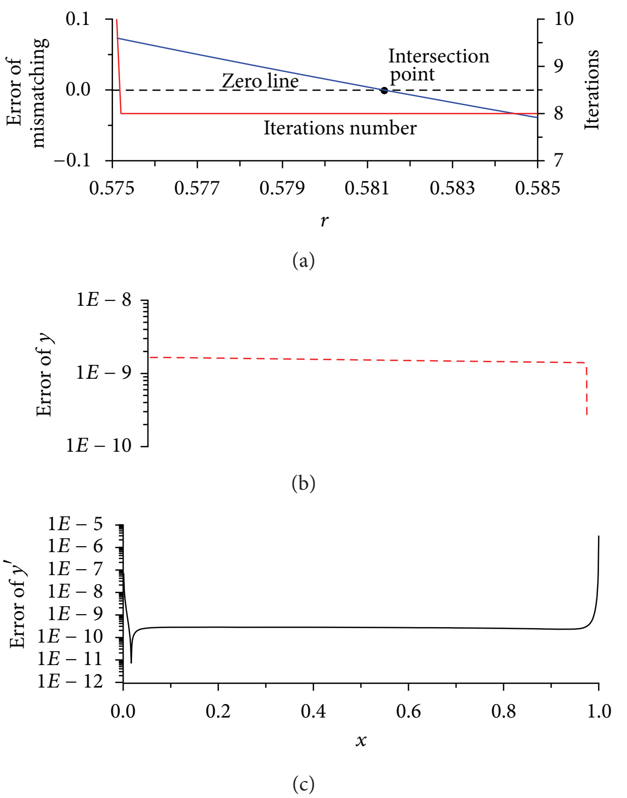


FIGURE 3: Example 2 solved by the SL(3, ℝ) method, (a) showing the error of mismatching and iterations number and showing the numerical errors (b) of y and (c) y'.

with an error in the order 10^{-10}. Upon comparing the numerical solutions with the closed-form solutions, the numerical errors of y and y' are, respectively, plotted in Figures 3(b) and 3(c). It can be seen that for y the accuracy is in the order of 10^{-9}, while that for y' the accuracy is in the order of 10^{-6}.

Example 3. We consider the same equation (53) but under the following boundary conditions:

$$y(0) = 0, \qquad y''(0) = 1, \qquad y''(1) = 1. \tag{56}$$

When the closed-form solution is given by

$$y(x) = \frac{x^2}{2} - \sin(\pi x), \tag{57}$$

the term $h(x)$ is given by

$$\begin{aligned} h(x) = {}& \pi^3 (p-1) \left[1 + \pi^2 \sin(\pi x)\right]^{p-2} \cos(\pi x) \\ & + \left[\frac{x^2}{2} - \sin(\pi x)\right]^2. \end{aligned} \tag{58}$$

For the above boundary conditions, we can use the following equations to iteratively solve the unknown initial value of u_2^0:

$$\begin{aligned} u_2^0 &= \frac{u_3^f - G_{31}u_1^0 - G_{33}u_3^0}{G_{32}}, \\ u_1^f &= G_{11}u_1^0 + G_{12}u_2^0 + G_{13}u_3^0, \\ u_2^f &= G_{21}u_1^0 + G_{22}u_2^0 + G_{23}u_3^0. \end{aligned} \tag{59}$$

We take $k_0 = 3$ and $p = 1.5$. If the target equation $u''(1) = k_0 + 1$ is satisfied, then we obtain the numerical solution. When we plot the mismatching error with respect to r in a range $[0.4675, 0.4755]$ in Figure 4, we find that there exist two intersection points at $r = 0.467958888$ and $r = 0.47500045$, which means that there exist two solutions. In (57), we only give one exact solution, but we do not have another solution as given in a closed-form.

Then, through a finer tuning of the value of r, we can match the right-end boundary condition very precisely with the error in the order 10^{-8}, and the first numerical solution is obtained with $r = 0.467958888$, while the second numerical solution is obtained with $r = 0.47500045$. Upon comparing the first numerical solution with the closed-form solution in (57), the numerical errors of y, y', and y'' are, respectively, plotted in Figures 5(a), 5(b), and 5(c). It can be seen that all the accuracies are in the order of 10^{-8}. In Figure 6, we compare the first numerical solution and the second numerical solution with the exact one. It can be seen that when the first numerical solution is almost coincident with the exact solution, the second numerical solution is obviously different from the first numerical solution. We can also observe that the second numerical solution satisfies the boundary conditions in (56) very precisely.

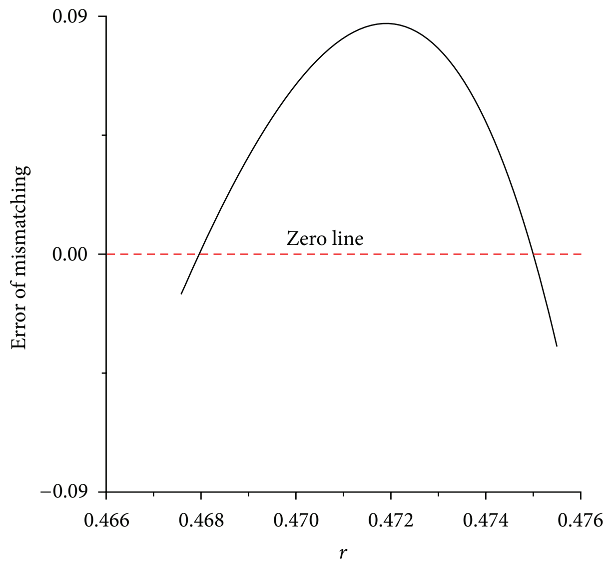


Figure 4: Example 3 solved by the $SL(3,\mathbb{R})$ method, showing the error of mismatching. There are two roots.

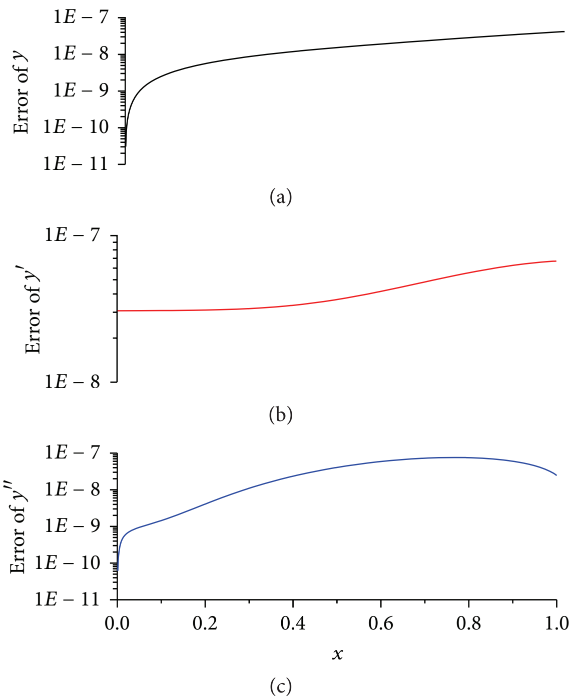


Figure 5: Example 3 showing the errors of the first numerical solution by comparing with the exact solution.

Example 4. Then, we consider a more general boundary conditions for the following p-Laplacian:

$$\begin{gathered}(\phi_p(y''(x)))' = F(x, y(x)), \quad x \in (0,\pi), \\ a_{11}y(0) + a_{12}y'(0) = B_1, \qquad a_{21}y(\pi) + a_{22}y'(\pi) = B_2, \\ y''(0) = 0.\end{gathered} \tag{60}$$

Similarly, we consider a translation with $u(x) = y(x)+k_0 > 0$, such that we have

$$\begin{gathered}(\phi_p(u''(x)))' = f(x, u(x)) \\ := F(x, u(x) - k_0), \quad x \in (0,\pi), \\ a_{11}u(0) + a_{12}u'(0) = b_1 := a_{11}k_0 + B_1, \\ a_{21}u(\pi) + a_{22}u'(\pi) = a_{21}k_0 + B_2, \qquad u''(0) = 0.\end{gathered} \tag{61}$$

In terms of (33), we can write

$$\begin{aligned}&(\mathbf{B}_0 + \mathbf{B}_f\mathbf{G})\mathbf{U}_0 \\ &= \left(\begin{bmatrix} a_{11} & a_{12} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1\end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix} G_{11} & G_{12} & 0 \\ G_{21} & G_{22} & 0 \\ G_{31} & G_{32} & 1\end{bmatrix}\right) \\ &\times \begin{bmatrix} u_1^0 \\ u_2^0 \\ u_3^0\end{bmatrix} \\ &= \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21}G_{11} + a_{22}G_{21} & a_{21}G_{12} + a_{22}G_{22} & 0 \\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix} u_1^0 \\ u_2^0 \\ u_3^0\end{bmatrix} \\ &= \begin{bmatrix} b_1 \\ b_2 \\ 0\end{bmatrix}.\end{aligned} \tag{62}$$

Thus, from (62) and (23), we can solve

$$\begin{gathered}u_1^0 = \frac{b_1(a_{21}G_{12} + a_{22}G_{22}) - b_2a_{12}}{a_{11}(a_{21}G_{12} + a_{22}G_{22}) - a_{12}(a_{21}G_{11} + a_{22}G_{21})}, \\ u_2^0 = \frac{b_2a_{11} - b_1(a_{21}G_{11} + a_{22}G_{21})}{a_{11}(a_{21}G_{12} + a_{22}G_{22}) - a_{12}(a_{21}G_{11} + a_{22}G_{21})}, \\ u_1^f = G_{11}u_1^0 + G_{12}u_2^0 + G_{13}u_3^0, \\ u_2^f = G_{21}u_1^0 + G_{22}u_2^0 + G_{23}u_3^0, \\ u_3^f = G_{31}u_1^0 + G_{32}u_2^0 + G_{33}u_3^0.\end{gathered} \tag{63}$$

The above five equations can be used to iteratively solve the five unknowns of u_1^0, u_2^0, u_1^f, u_2^f, and u_3^f. We note that $u_3^0 = 0$ by the last boundary condition in (61).

As a demonstrative case, we take $F(x, y(x)) = h(x) - y(x)$, that is, $f(x, u(x)) = h(x) - [u(x) - k_0]$, $p = 3$, and $y(x) = e^x \cos x$ to be the exact solution, where $h(x)$

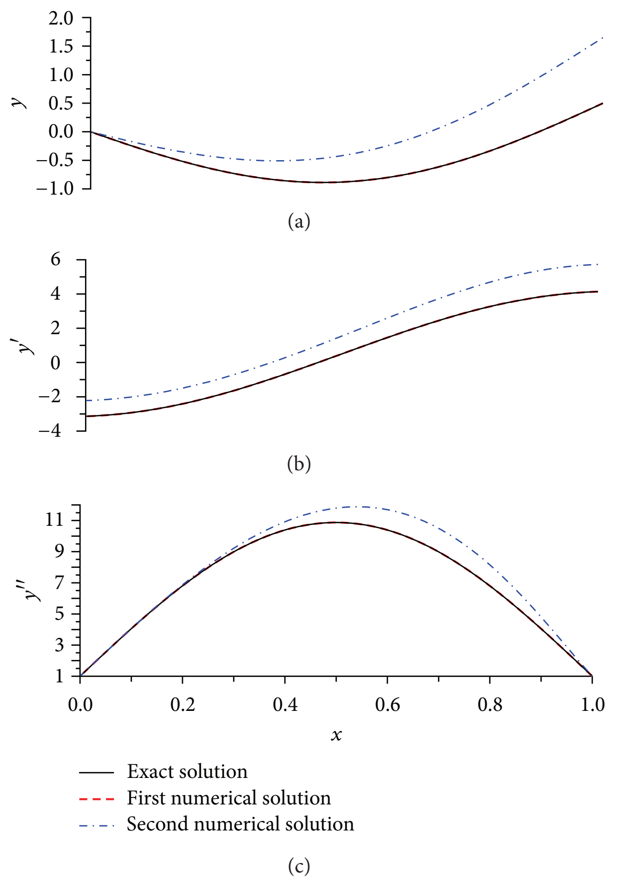


FIGURE 6: Example 3 comparing the first numerical solution and the second numerical solution with the exact solution.

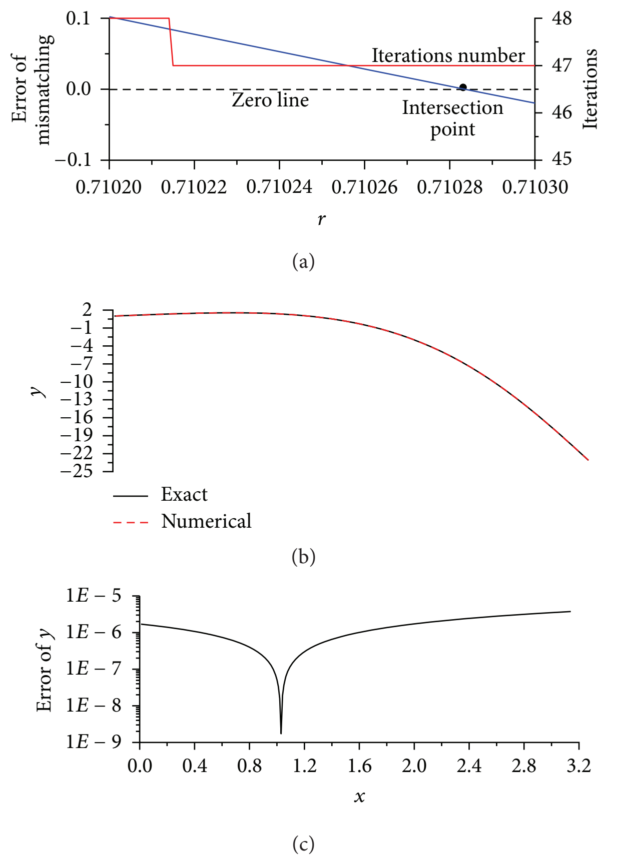


FIGURE 7: Example 4 solved by the SL(3, ℝ) method, (a) showing the error of mismatching and iterations number, (b) comparing the numerical and exact solutions, and (c) showing the numerical error of y.

can be computed from (60) by inserting the above $y(x)$. The boundary conditions are given by

$$y(0)+y'(0)=2, \qquad y(\pi)+y'(\pi)=-2e^{\pi},$$
$$y''(0)=0,$$
$$\text{that is, } u(0)+u'(0)=k_0+2, \qquad u(\pi)+u'(\pi)=k_0-2e^{\pi},$$
$$u''(0)=0. \tag{64}$$

We take $k_0 = 25$. If the target equation $u_1^f + u_2^f - k_0 + 2e^{\pi} = 0$ is satisfied, then we obtain the numerical solution. When we plot the mismatching error with respect to r in a finer range $[0.7102, 0.7103]$ in Figure 7, we find that there exists one intersection point at $r = 0.710283575975$. We can match the right-end boundary condition very precisely with an error being -3.694×10^{-10}. Because there are many equations to be solved iteratively, the number of iterations as shown in Figure 7(a) is between 45 and 48, which is higher than the previous three examples. In Figure 7(b), we compare the numerical solution of $y(x)$ with the exact solution $y(x) = e^x \cos x$, whose numerical error as shown in Figure 7(c) is quite accurate in the order of 10^{-6}.

Alternatively, we consider a nonlinear perturbation of the above example under the same boundary conditions but with

$$F(x, y(x)) = h(x) - y(x) + \frac{1}{2}y^2(x),$$
$$h(x) = -2(p-1)e^{x}(\cos x + \sin x)\left|2e^{x}\sin x\right|^{p-2}, \tag{65}$$

where we also fix $p = 3$. For this problem, we do not have a closed-form solution. However, we take $k_0 = 26$, and by taking $r = 0.6863523$ and $r = 0.799$, we can obtain two numerical solutions as shown in Figure 8. The numerical solution as shown by the dashed line is quite unstable. For the purpose of comparison, we also plot the numerical solutions obtained in the last example in (64) by the dashed-dotted lines in Figure 8. It can be seen that the solid lines are somewhat perturbed from the ones of the dashed-dotted lines, but the unstable ones are quite different from the above two solutions.

4. Power-Law Fluids

In this section, we consider the boundary layer problems of power-law fluid in (6). We use the SL(3, ℝ) Lie-group

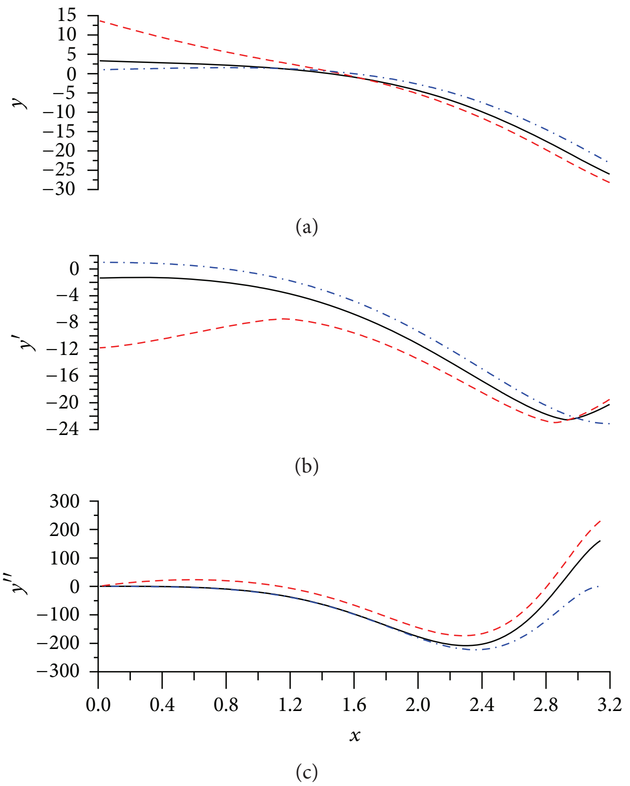


Figure 8: A nonlinear perturbation of Example 4, displaying two solutions of (a) y, (b) y', and (c) y''.

shooting method developed in Section 2.3 by adjusting the value of u_3^0. However, before that we need to treat the difficulty mentioned in Section 2.3 if we use the boundary conditions in (36).

First, we need to point out that for this boundary layer problem we only consider that the function $f(x)$ is convex, that is, $f''(x) \geq 0$. So the term b defined in (30) is positive. Then, from (28), (23), and (32), we have

$$u_1^f = \cosh\left(\ell\sqrt{b}\right)u_1^0 + \frac{1}{\sqrt{b}}\sinh\left(\ell\sqrt{b}\right)u_2^0, \quad (66)$$

$$u_2^f = \sqrt{b}\sinh\left(\ell\sqrt{b}\right)u_1^0 + \cosh\left(\ell\sqrt{b}\right)u_2^0, \quad (67)$$

$$u_3^f = c\sinh\left(\ell\sqrt{b}\right)u_1^0 + \frac{c}{\sqrt{b}}\left[\cosh\left(\ell\sqrt{b}\right) - 1\right]u_2^0 + u_3^0, \quad (68)$$

where $\ell = \eta_\infty$.

Let $z := \ell\sqrt{b}$. From (67), we can derive a scalar equation to solve z:

$$u_2^0\ell\cosh z + u_1^0 z\sinh z - \ell u_2^f = 0. \quad (69)$$

Because $u_1^0 = k_0 - C_0$, $u_2^0 = \xi$, and $u_2^f = 1$ are given from the boundary conditions, and $\ell = \eta_\infty$ is selected, we can apply the Newton method to solve the above equation, whose solution is denoted by z_0. Hence, we have

$$b = \left(\frac{z_0}{\ell}\right)^2, \quad (70)$$

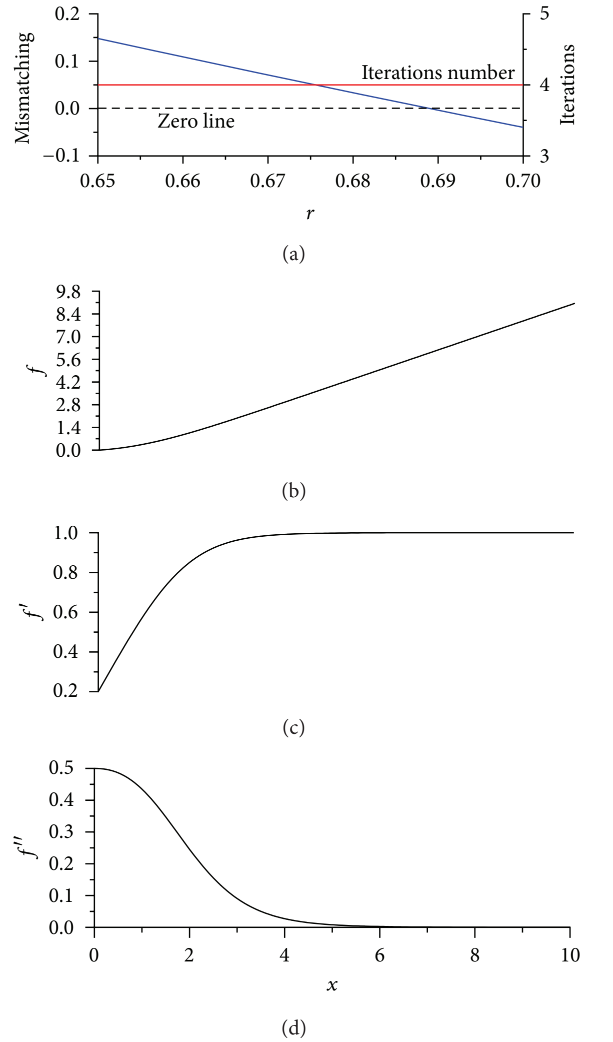


Figure 9: The boundary layer problem of a power-law fluid with $N = 0.8$ solved by the $SL(3,\mathbb{R})$ method, (a) showing the error of mismatching and iterations number and showing the numerical solutions of (b) f, (c) f', and (d) f''.

by the definition of $z = \ell\sqrt{b}$. Furthermore, by (30), we have

$$ru_3^0 + (1-r)u_3^f = \phi_p\left(b\hat{u}_1\right). \quad (71)$$

From (68) and (71), we can solve

$$\begin{aligned} u_3^0 &= \phi_p\left(b\hat{u}_1\right) - (1-r) \\ &\quad \times \left[c\sinh\left(\ell\sqrt{b}\right)u_1^0 + \frac{c}{\sqrt{b}}\left[\cosh\left(\ell\sqrt{b}\right) - 1\right]u_2^0\right], \\ u_3^f &= \phi_p\left(b\hat{u}_1\right) \\ &\quad + r\left[c\sinh\left(\ell\sqrt{b}\right)u_1^0 + \frac{c}{\sqrt{b}}\left[\cosh\left(\ell\sqrt{b}\right) - 1\right]u_2^0\right], \end{aligned} \quad (72)$$

where $c = -(\hat{u}_1 - k_0)\phi_q(\hat{u}_3)/(\sqrt{b}\hat{u}_1)$. When b is solved, u_1^f is determined by (66); hence, (72) can be used iteratively to

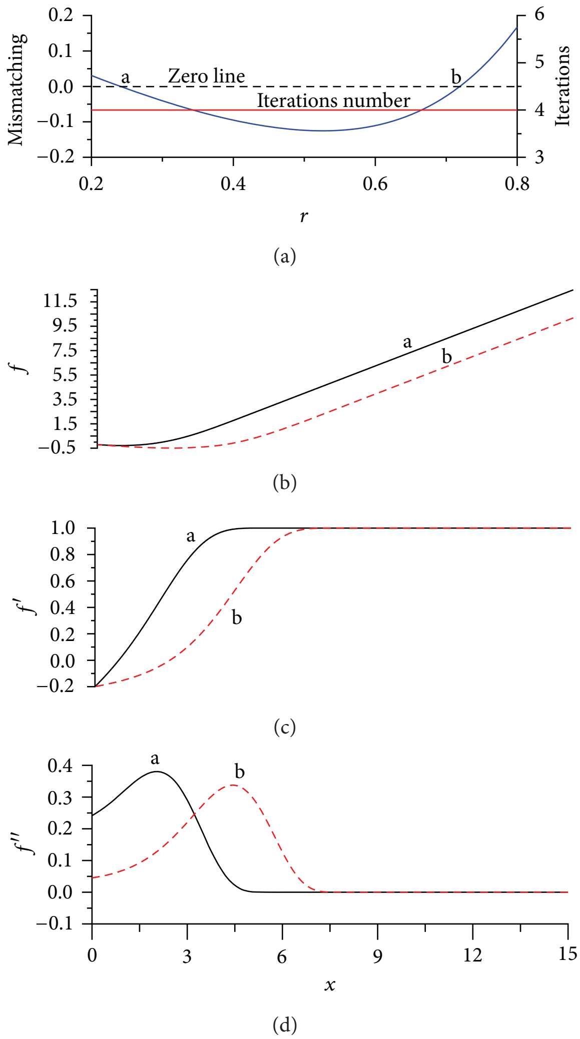


FIGURE 10: The boundary layer problem of power-law fluid with $N = 1.2$ solved by the $SL(3, \mathbb{R})$ method, (a) showing the error of mismatching and iterations number and showing two different numerical solutions of (b) f, (c) f', and (d) f''.

solve the two unknowns of u_3^0 and u_3^f. We can satisfy the target equation $u_2^f = 1$ by selecting the best value of r.

Example 5. We fix $N = 0.8$, $C_0 = 0$, $\xi = 0.2$, $\eta_\infty = 10$, and $k_0 = 1$, and the convergence criterion is $\varepsilon = 10^{-5}$. Although under this stringent convergence criterion the iteration process to find u_3^0 is convergent very fast as shown in Figure 9(a), where for $r \in [0.65, 0.7]$ the iteration numbers are all to be four. In Figure 9(a), we plot the mismatching error with respect to r in the same range. It can be seen that the mismatching error curve is intersected with the zero line at a point near to 0.68. Then, through a finer tuning of the value to $r = 0.689064$, we can match the right-end boundary condition very precisely with an error in the order of 10^{-7}. The unknown initial value of $u_3(0) = v(0) = 0.4998652$ (or $f''(0) = 0.4203065$) is obtained. The numerical results of f, f', and f'' are, respectively, plotted in Figures 9(b)–9(d).

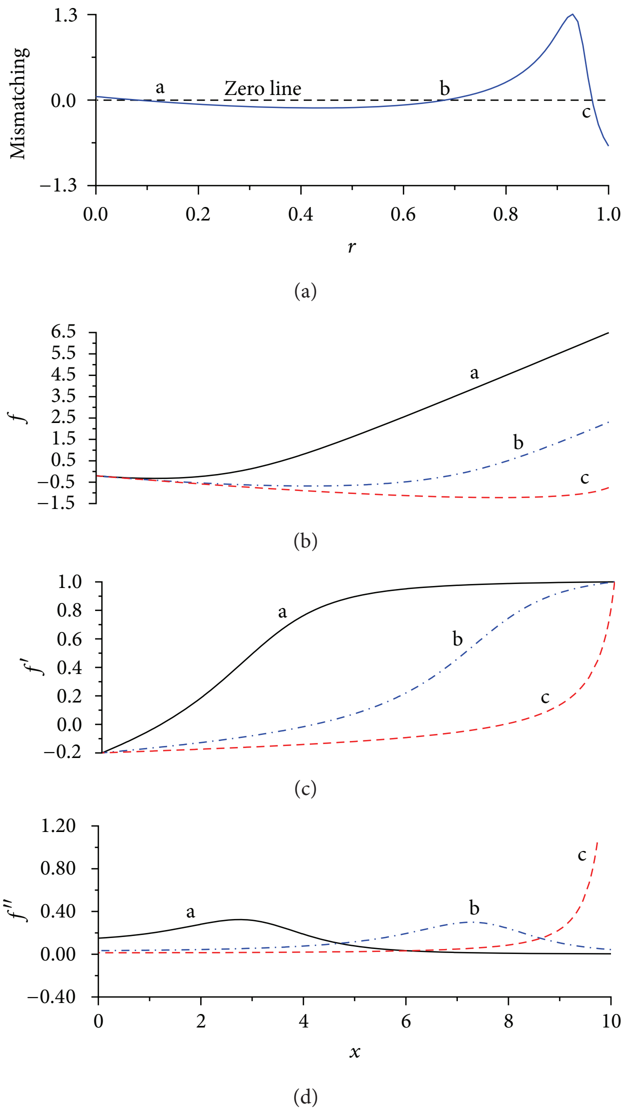


FIGURE 11: The boundary layer problem of a power-law fluid with $N = 0.3$ solved by the $SL(3, \mathbb{R})$ method, (a) showing the error of mismatching and iterations number and showing three different numerical solutions of (b) f, (c) f', and (d) f''.

Example 6. We fix $N = 1.2$, $C_0 = 0.2$, $\xi = -0.2$, $\eta_\infty = 15$, and $k_0 = 1$, and the convergence criterion is $\varepsilon = 10^{-5}$. As shown in Figure 10(a), where for $r \in [0.2, 0.8]$ the iteration numbers are all to be four, the mismatching error curve is intersected with the zero line at two points a and b. Then, through a finer tuning of the values to $r = 0.2408982$ and $r = 0.7191693$, we can match the right-end boundary condition very precisely with the errors in the order of 10^{-8}, and thus we obtain two numerical solutions as compared in Figures 10(b)–10(d). For the first solution, the unknown initial value of $u_3(0) = v(0) = 0.18159807$ (or $f''(0) = 0.24131965$) is obtained, while for the second solution the unknown initial value of $u_3(0) = v(0) = 0.024415987$ (or $f''(0) = 0.04533098$) is obtained. The numerical results of

$v(0) = 0.18159807$ and $v(0) = 0.024415987$ are very close to that obtained by Liu [14].

Example 7. We fix $N = 0.3$, $C_0 = 0.2$, $\xi = -0.2$, $\eta_\infty = 10$, and $k_0 = 0.5$. As shown in Figure 11(a), the mismatching error curve is intersected with the zero line at three points a, b, and c. Then, through a finer tuning of the values to $r = 0.081054$, $r = 0.6812354$, and $r = 0.96791285$, we can obtain three corresponding numerical solutions as compared in Figures 11(b)–11(d). For the first solution, the unknown initial value of $f''(0) = 0.150134848$ is obtained, and for the second solution the unknown initial value is $f''(0) = 0.0337905$, while that for the third solution the unknown initial value is $f''(0) = 0.01283368$. For the last solution, f'' grows rapidly after $x = 8$.

5. Conclusions

In the present paper, we have offered a rather accurate and simple method with only a few iterations to find the unknown left boundary conditions by applying the $SL(3,\mathbb{R})$ Lie-group shooting method to the third-order p-Laplacian boundary value problems. Also, as an application, we have solved the boundary layer problems of power-law fluids by the present method. The $SL(3,\mathbb{R})$ Lie-group shooting method allows us to express the missing left-end boundary conditions by the closed-form functions of $r \in [0, 1]$, where the best r is determined iteratively by matching the right-end boundary conditions. Because the iterations to find the missing left-end boundary conditions are convergent very fast, the Lie-group shooting method based on $SL(3,\mathbb{R})$ is quite computationally efficient. The new method was effective to find the multiple solutions, although for the highly nonlinear case with multiple unknown left boundary conditions.

Acknowledgments

The paper was supported by the Project NSC-99-2221-E-002-074-MY3, and the 2011 Outstanding Research Award from Taiwan's National Science Council and the 2011 Taiwan Research Front Award from Thomson Reuters, granted to the author, are highly appreciated.

References

[1] W. R. Schowalter, "The application of boundary layer theory to power-law pseudoplastic fluids: similar solutions," *American Institute of Chemical Engineering*, vol. 6, pp. 24–28, 1960.

[2] A. Acrivos, M. J. Shah, and E. E. Petersen, "Momentum and heat transfer in laminar boundary layer ows of non-Newtonian uids past external surfaces," *American Institute of Chemical Engineering*, vol. 6, pp. 312–318, 1960.

[3] T. G. Howell, D. R. Jeng, and K. J. De Witt, "Momentum and heat transfer on a continuous moving surface in a power law fluid," *International Journal of Heat and Mass Transfer*, vol. 40, no. 8, pp. 1853–1861, 1997.

[4] A. Nachman and A. Callegari, "A nonlinear singular boundary value problem in the theory of pseudoplastic fluids," *SIAM Journal on Applied Mathematics*, vol. 38, no. 2, pp. 275–281, 1980.

[5] M. N. Ozisik, *Basic Heat Transfer*, McGraw-Hill, New York, NY, USA, 1979.

[6] H. Schlichting, *Boundary Layer Theory*, McGraw-Hill, New York, NY, USA, 1979.

[7] J. J. Shu and G. Wilks, "Heat transfer in the flow of a cold, two-dimensional vertical liquid jet against a hot, horizontal plate," *International Journal of Heat and Mass Transfer*, vol. 39, no. 16, pp. 3367–3379, 1996.

[8] L. C. Zheng and X. X. Zhang, "Skin friction and heat transfer in power-law fluid laminar boundary layer along a moving surface," *International Journal of Heat and Mass Transfer*, vol. 45, no. 13, pp. 2667–2672, 2002.

[9] C.-S. Liu, "Cone of non-linear dynamical system and group preserving schemes," *International Journal of Non-Linear Mechanics*, vol. 36, no. 7, pp. 1047–1068, 2001.

[10] C.-S. Liu, "The Lie-group shooting method for nonlinear two-point boundary value problems exhibiting multiple solutions," *Computer Modeling in Engineering & Sciences*, vol. 13, no. 2, pp. 149–163, 2006.

[11] 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.

[12] C.-S. Liu and J. R. Chang, "The Lie-group shooting method for multiple-solutions of Falkner-Skan equation under suction-injection conditions," *International Journal of Non-Linear Mechanics*, vol. 43, pp. 844–851, 2008.

[13] C.-S. Liu, C.-W. Chang, and J.-R. Chang, "A new shooting method for solving boundary layer equations in fluid mechanics," *Computer Modeling in Engineering & Sciences*, vol. 32, no. 1, pp. 1–15, 2008.

[14] C.-S. Liu, "The Lie-group shooting method for boundary-layer problems with suction/injection/reverse ow conditions for power-law fluids," *International Journal of Non-Linear Mechanics*, vol. 46, pp. 1001–1008, 2011.

[15] C.-S. Liu, W. Yeih, and S. N. Atluri, "An enhanced fictitious time integration method for non-linear algebraic equations with multiple solutions: boundary layer, boundary value and eigenvalue problems," *Computer Modeling in Engineering & Sciences*, vol. 59, no. 3, pp. 301–323, 2010.

[16] C.-S. Liu, "Computing the eigenvalues of the generalized Sturm-Liouville problems based on the Lie-group $SL(2,\mathbb{R})$," *Journal of Computational and Applied Mathematics*, vol. 236, no. 17, pp. 4547–4560, 2012.

[17] C.-S. Liu, "An $SL(3,\mathbb{R})$ shooting method for solving the Falkner-Skan boundary layer equation," *International Journal of Non-Linear Mechanics*, vol. 49, pp. 145–151, 2013.

[18] C. Yang and J. Yan, "Positive solutions for third-order Sturm-Liouville boundary value problems with p-Laplacian," *Computers & Mathematics with Applications*, vol. 59, no. 6, pp. 2059–2066, 2010.

[19] R. Glowinski and J. Rappaz, "Approximation of a nonlinear elliptic problem arising in a non-Newtonian fluid flow model in glaciology," *Mathematical Modelling and Numerical Analysis*, vol. 37, no. 1, pp. 175–186, 2003.

[20] C. Liu, "Weak solutions for a viscous p-Laplacian equation," *Electronic Journal of Differential Equations*, vol. 2003, no. 63, pp. 1–11, 2003.

[21] I. Ly and D. Seck, "Isoperimetric inequality for an interior free boundary problem with p-Laplacian operator," *Electronic Journal of Differential Equations*, vol. 2005, no. 119, pp. 1–12, 2005.

[22] M. Ramaswamy and R. Shivaji, "Multiple positive solutions for classes of p-Laplacian equations," *Differential and Integral Equations*, vol. 17, no. 11-12, pp. 1255–1261, 2004.

[23] S. Oruganti, J. Shi, and R. Shivaji, "Diffusive logistic equation with constant yield harvesting. I. Steady states," *Transactions of the American Mathematical Society*, vol. 354, no. 9, pp. 3601–3619, 2002.

[24] S. Oruganti, J. Shi, and R. Shivaji, "Logistic equation with p-Laplacian and constant yield harvesting," *Abstract and Applied Analysis*, vol. 9, pp. 723–727, 2004.

[25] L. C. Evans and W. Gangbo, "Differential equations methods for the Monge-Kantorovich mass transfer problem," *Memoirs of the American Mathematical Society*, vol. 137, no. 653, 1999.