Hindawi Publishing Corporation
Boundary Value Problems
Volume 2008, Article ID 963753, 22 pages
doi:10.1155/2008/963753

Research Article

A Two-Stage LGSM for Three-Point BVPs of Second-Order ODEs

Chein-Shan Liu[1, 2]

[1] *Department of Mechanical and Mechatronic Engineering, Taiwan Ocean University, Keelung 20224, Taiwan*
[2] *Department of Harbor and River Engineering, Taiwan Ocean University, Keelung 20224, Taiwan*

Correspondence should be addressed to Chein-Shan Liu, csliu@mail.ntou.edu.tw

Received 12 May 2008; Accepted 9 July 2008

Recommended by Wenming Zou

The study in this paper is a numerical integration of second-order three-point boundary value problems under two imposed nonlocal boundary conditions at $t = t_0$, $t = \xi$, and $t = t_1$ in a general setting, where $t_0 < \xi < t_1$. We construct a two-stage Lie-group shooting method for finding unknown initial conditions, which are obtained through an iterative solution of derived algebraic equations in terms of a weighting factor $r \in (0,1)$. The best r is selected by matching the target with a minimal discrepancy. Numerical examples are examined to confirm that the new approach has high efficiency and accuracy with a fast speed of convergence. Even for multiple solutions, the present method is also effective to find them.

1. Introduction

Nonlinear ordinary differential equations (ODEs) described a majority of engineering problems. When boundary conditions are imposed, the resulting problems are referred to as boundary value problems (BVPs). Naturally, the solutions of BVPs have to satisfy the boundary conditions, but in many cases this may be a difficult task when one is concerned with the numerical integrations of multipoint BVPs. There are many computational methods that have been developed for solving two-point BVPs; see, for example, Kubicek and Hlavacek [1], Cash [2, 3], Cash and Wright [4], Keller [5], Ascher et al. [6], Deeba et al. [7], Garg [8], Ha [9], Ha and Lee [10], and Cuomo and Marasco [11]. However, for the three-point BVPs only a few computational methods have been reported [12, 13].

In this paper, we propose a new method for the numerical integration of the following second-order BVP:

$$\ddot{x} = f(t, x, \dot{x}), \quad t_0 < t < t_1, \tag{1.1}$$

$$H_1(x(t_0), \dot{x}(t_0), x(\xi), \dot{x}(\xi), x(t_1), \dot{x}(t_1)) = 0, \tag{1.2}$$

$$H_2(x(t_0), \dot{x}(t_0), x(\xi), \dot{x}(\xi), x(t_1), \dot{x}(t_1)) = 0, \tag{1.3}$$

where $x(t_0)$, $\dot{x}(t_0)$, $x(\xi)$, $\dot{x}(\xi)$, $x(t_1)$, and $\dot{x}(t_1)$ are, respectively, the values of x and $\dot{x}$ at three different temporal points $t_0 < \xi < t_1$. Here, $[t_0, t_1]$ is a time interval of our problem. However, in many physical applications t may represent a spatial coordinate. Since the boundary conditions are specified at three distinct points, this problem is called a three-point boundary value problem, which is one sort of nonlocal boundary value problems. Because there are only two equations (1.2) and (1.3), we need to derive other four extra equations to solve the six unknowns of $x(t_0)$, $\dot{x}(t_0)$, $x(\xi)$, $\dot{x}(\xi)$, $x(t_1)$, and $\dot{x}(t_1)$.

For initial value problems (IVPs) the time-stepping techniques are well developed, which require the initial conditions of both x and $y = \dot{x}$ for the second-order ODEs. If they are available, then we can numerically integrate the following IVP step-by-step in a forward direction from $t = t_0$ to $t = t_1$:

$$\dot{x} = y, \tag{1.4}$$

$$\dot{y} = f(t, x, y), \tag{1.5}$$

$$x(t_0) = \alpha, \tag{1.6}$$

$$y(t_0) = A. \tag{1.7}$$

The shooting method involves a choice of the missing initial conditions in (1.6) and (1.7), which together with the numerical solutions at the midpoint $t = \xi$ and at the terminal point $t = t_1$ must satisfy the constraints in (1.2) and (1.3).

Basically, the shooting method is to assume some unknown initial conditions and to convert the BVP to an IVP. Solve the IVP and compare the solution at the boundary to the given boundary conditions. In general, the solution will not immediately satisfy the boundary conditions, and it requires many iterations to adjust the initial guess through some techniques. This iterative approach is called a shooting method. How to choose suitable initial conditions may be difficult when the guesses are carried out in an indefinite region and in a multidimensional space. The shooting method is a trial-and-error method and is often sensitive to initial guess. All that make the computation by the conventional shooting method expensive and ineffective.

Multipoint BVP has attracted much attention from researchers, due to its great challenge in the proofs of existence, nonuniqueness and positive solutions [14–19]. Gupta [20, 21] first studied the solvability of three-point BVPs of second-order ODEs. The shooting technique was used by Kwong [22] to study a certain three-point BVP of second-order ODE with a condition of $x(0) = 0$. Quasilinearization method is also used by Ahmad et al. [23] to obtain a monotone sequence converging quadratically to a solution of three-point BVP of second-order ODE.

Developing here is a new two-stage Lie-group shooting method (TSLGSM) for the three-point BVP governed by (1.1)–(1.3). Our approach of the above problem is stemmed from the group preserving scheme (GPS) developed previously by Liu [24] for initial value problems of ODEs. Recently, Liu [25–27] has extended the GPS technique to solve the two-point BVPs, and numerical results reveal that the Lie group method is a rather promising technique to effectively calculate the two-point BVPs. In the construction of Lie group method for the calculations of BVPs, Liu [25] has introduced the idea of one-step GPS by utilizing the closure property of Lie group, and hence, the new shooting method has been named the Lie-group shooting method (LGSM). Chang et al. [28] have employed the LGSM to solve a backward heat conduction problem with a high performance. Liu [29, 30] has employed the LGSM technique to accurately solve the inverse heat conduction problems of identifying nonhomogeneous heat conductivity functions and time-dependent heat conductivity functions. More interestingly, as shown by Liu [31], the Lie-group method is also useful in the inverse Sturm-Liouville problem.

The idea behind the one-step Lie-group transformation is rather promising to provide efficient numerical methods in many issues including the inverse problems and boundary value problems. The one-step GPS has been applied to the solutions of BVPs by Liu [25] but is restricted to simpler two-point boundary conditions. The present approach can be applied to the second-order three-point BVPs in a general setting, of which we can search the missing initial conditions through an iterative solution to find a suitable r in a finite range of $r \in (0,1)$.

This paper is arranged as follows. In Section 2, we give a brief sketch of the group preserving scheme for ODEs. In Section 3, we explain the mathematical basis of the construction of a one-step GPS by using the closure property of Lie group, and combine it with the midpoint rule to construct a single-parameter Lie group element in terms of a weighting factor r, and more importantly a universal one-step Lie-group element. In this section, an important Lie-group shooting equation is derived. In Section 4, we derive a new two-stage Lie-group shooting method to solve the three-point BVPs. In Section 5, we use numerical examples to demonstrate the efficiency of the new method. Finally, we draw some conclusions in Section 6.

2. Preliminaries

Although we do not know previously the symmetry group of nonlinear differential equations system, Liu [24] has embedded it into an augmented system and found an internal symmetry of the new system. That is, for an ODEs system with dimensions n,

$$\dot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}), \quad \mathbf{u} \in \mathbb{R}^n, \ t > t_0, \tag{2.1}$$

we can deal with the following $n+1$-dimensional augmented system:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X}, \tag{2.2}$$

where

$$\mathbf{X} := \begin{bmatrix} \mathbf{u} \\ \|\mathbf{u}\| \end{bmatrix} \tag{2.3}$$

is an augmented state vector, and

$$\mathbf{A} := \begin{bmatrix} \mathbf{0}_{n\times n} & \dfrac{\mathbf{f}(t,\mathbf{u})}{\|\mathbf{u}\|} \\ \dfrac{\mathbf{f}^{\{\mathrm{T}\}}(t,\mathbf{u})}{\|\mathbf{u}\|} & 0 \end{bmatrix} \tag{2.4}$$

is an element of the Lie algebra $so(n,1)$ satisfying

$$\mathbf{A}^{\{\mathrm{T}\}}\mathbf{g} + \mathbf{g}\mathbf{A} = \mathbf{0}. \tag{2.5}$$

Here,

$$\mathbf{g} = \begin{bmatrix} \mathbf{I}_n & \mathbf{0}_{n\times 1} \\ \mathbf{0}_{1\times n} & -1 \end{bmatrix} \tag{2.6}$$

is a Minkowski metric. $\mathbf{I}_n$ is the identity matrix of order n, and the superscript $\{T\}$ stands for the transpose.

It is obvious that the equation in the first row in (2.2) is the same as the original (2.1), but the inclusion of the second row in (2.2) gives us a Minkowskian structure of the augmented system for $\mathbf{X}$ satisfying the cone condition:

$$\mathbf{X}^{\{T\}}\mathbf{g}\mathbf{X} = \mathbf{u}\cdot\mathbf{u} - \|\mathbf{u}\|^2 = 0. \tag{2.7}$$

The cone condition is a natural constraint of the new system (2.2).

Accordingly, Liu [24] has developed a group-preserving scheme:

$$\mathbf{X}_{\ell+1} = \mathbf{G}(\ell)\mathbf{X}_\ell, \tag{2.8}$$

where $\mathbf{X}_\ell$ denotes the numerical value of $\mathbf{X}$ at the discrete time t_ℓ, and $\mathbf{G}(\ell) \in SO_o(n,1)$ satisfies

$$\mathbf{G}^{\{T\}}\mathbf{g}\mathbf{G} = \mathbf{g}, \tag{2.9}$$

$$\det \mathbf{G} = 1, \tag{2.10}$$

$$G_0^0 > 0, \tag{2.11}$$

where G_0^0 is the 00th component of $\mathbf{G}$. Equation (2.8) guarantees that each $\mathbf{X}_\ell$ is located on the cone satisfying the cone condition (2.7), if $\mathbf{G}$ is a proper orthochronous Lorentz group.

An exponential mapping of $\mathbf{A}(\ell)$ is given by

$$\exp\left[\Delta t \mathbf{A}(\ell)\right] = \begin{bmatrix} \mathbf{I}_n + \dfrac{(a_\ell - 1)}{\|\mathbf{f}_\ell\|^2}\mathbf{f}_\ell\mathbf{f}_\ell^{\{T\}} & \dfrac{b_\ell \mathbf{f}_\ell}{\|\mathbf{f}_\ell\|} \\ \dfrac{b_\ell \mathbf{f}_\ell^{\{T\}}}{\|\mathbf{f}_\ell\|} & a_\ell \end{bmatrix}, \tag{2.12}$$

where

$$a_\ell := \{\cosh\}\left(\frac{\Delta t\|\mathbf{f}_\ell\|}{\|\mathbf{u}_\ell\|}\right), \tag{2.13}$$

$$b_\ell := \sinh\left(\frac{\Delta t\|\mathbf{f}_\ell\|}{\|\mathbf{u}_\ell\|}\right). \tag{2.14}$$

For saving notation, we let $\mathbf{f}_\ell = \mathbf{f}(\mathbf{u}_\ell, t_\ell)$. Substituting the above $\exp[\Delta t\mathbf{A}(\ell)]$ for $\mathbf{G}(\ell)$ into (2.8) and taking its first row, we obtain

$$\mathbf{u}_{\ell+1} = \mathbf{u}_\ell + \eta_\ell \mathbf{f}_\ell = \mathbf{u}_\ell + \frac{(a_\ell - 1)\mathbf{f}_\ell\cdot\mathbf{u}_\ell + b_\ell\|\mathbf{u}_\ell\|\|\mathbf{f}_\ell\|}{\|\mathbf{f}_\ell\|^2}\mathbf{f}_\ell. \tag{2.15}$$

From $\mathbf{f}_\ell\cdot\mathbf{u}_\ell \geq -\|\mathbf{f}_\ell\|\|\mathbf{u}_\ell\|$, we can prove that

$$\eta_\ell \geq \frac{\|\mathbf{u}_\ell\|}{\|\mathbf{f}_\ell\|}\left[1 - \exp\left(-\frac{\Delta t\|\mathbf{f}_\ell\|}{\|\mathbf{u}_\ell\|}\right)\right] > 0, \quad \forall \Delta t > 0, \tag{2.16}$$

and that (2.15) is a group properties preserving scheme for all $\Delta t > 0$.

Recently, Ying and Candés [32] have introduced a phase flow method for nonlinear ordinary differential equations in (2.1) by setting

$$\frac{d\mathbf{u}}{ds} = \mathbf{f}(t, \mathbf{u}), \qquad \frac{dt}{ds} = 1. \tag{2.17}$$

Therefore, the original n-dimensional ODEs system is embedded into an $n+1$-dimensional system in the space of $(\mathbf{u}, t)$. This technique is not at all a new one, which was already appeared in many textbooks of ODEs to treat the nonautonomous ODEs system as an autonomous one. However, Ying and Candés [32] have used this technique to construct a novel and accurate approach for the phase flow maps of certain ODEs. The above augmented system is drastically different from the one in (2.2). The main features of our formulation in (2.2) are three folds: (1) a geometric cone structure, (2) a Lie-algebra structure, and then (3) a Lie-group structure of $\mathrm{SO}_o(n,1)$. Even the phase flow method may have a Lie-group structure, but it is hard to find this Lie-group mapping; more precisely, finding this Lie-group transformation of phase flow is equivalent to finding the mapping on the solution curves of the original nonlinear ODEs.

3. Two Lie-group elements

Applying scheme (2.15) to the ODEs in (2.1) with a specified initial condition $\mathbf{u}(t_0) = \mathbf{u}_{t_0,}$ we can compute the solution $\mathbf{u}(t)$ by the GPS. Assuming that the total time span $t_1 - t_0$ is divided by K steps, that is, the time-step size used in the GPS is $\Delta t = (t_1 - t_0)/K$. Starting from an initial augmented condition $\mathbf{X}_{t_0} = \mathbf{X}(t_0) = (\mathbf{u}_{t_0}^{\{\mathrm{T}\}}, \|\mathbf{u}_{t_0}\|)^{\{\mathrm{T}\}}$ and applying (2.8) step-by-step, we can obtain the value $\mathbf{X}(t_1) = (\mathbf{u}^{\{\mathrm{T}\}}(t_1), \|\mathbf{u}(t_1)\|)^{\{\mathrm{T}\}}$ at a desired time $t = t_1$ by

$$\mathbf{X}_{t_1} = \mathbf{G}_K(\Delta t) \cdots \mathbf{G}_1(\Delta t)\mathbf{X}_{t_0}. \tag{3.1}$$

Let us recall that each $\mathbf{G}_i,\ i = 1, \ldots, K$ is an element of the Lie group $\mathrm{SO}_o(n,1)$, and by the closure property of Lie group, $\mathbf{G}_K \cdots \mathbf{G}_1$ is also a Lie group. To prove this closure property, let us consider two elements $\mathbf{G}_1, \mathbf{G}_2 \in \mathrm{SO}_o(n,1)$, that is,

$$\mathbf{G}_1^{\{\mathrm{T}\}}\mathbf{g}\mathbf{G}_1 = \mathbf{g}, \qquad \mathbf{G}_2^{\{\mathrm{T}\}}\mathbf{g}\mathbf{G}_2 = \mathbf{g}. \tag{3.2}$$

Then, by using the above two equations we have

$$(\mathbf{G}_2\mathbf{G}_1)^{\{\mathrm{T}\}}\mathbf{g}\mathbf{G}_2\mathbf{G}_1 = \mathbf{G}_1^{\{\mathrm{T}\}}\mathbf{G}_2^{\{\mathrm{T}\}}\mathbf{g}\mathbf{G}_2\mathbf{G}_1 = \mathbf{G}_1^{\{\mathrm{T}\}}\mathbf{g}\mathbf{G}_1 = \mathbf{g}. \tag{3.3}$$

It means that $\mathbf{G}_2\mathbf{G}_1 \in \mathrm{SO}_o(n,1)$ if $\mathbf{G}_1, \mathbf{G}_2 \in \mathrm{SO}_o(n,1)$.

According to this argument, we can prove that $\mathbf{G}_K \cdots \mathbf{G}_1 \in \mathrm{SO}_o(n,1)$, because of $\mathbf{G}_K, \ldots, \mathbf{G}_1 \in \mathrm{SO}_o(n,1)$. Therefore in $\mathrm{SO}_o(n,1)$, there exists an element denoted by $\mathbf{G}$ which is identical to $\mathbf{G}_K \cdots \mathbf{G}_1$. Hence, from (3.1) we have

$$\mathbf{X}_{t_1} = \mathbf{G}\mathbf{X}_{t_0}. \tag{3.4}$$

This is a one-step Lie-group transformation from $\mathbf{X}_{t_0}$ to $\mathbf{X}_{t_1}$. However, it is worthwhile to point out that other numerical methods cannot share this property, since they are not of the Lie group schemes.

The exact $\mathbf{G}$ is hardly to find. However, we can approximate the exact $\mathbf{G}$ by a numerical one through some numerical methods developed below.

3.1. A Lie-group element G(r)

In above we have explored the concept of the one-step **G**. In order to increase the accuracy of our shooting method to search some unknown initial conditions of the three-point BVPs, we can calculate **G** by a midpoint rule:

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_n + \frac{(a-1)}{\|\widehat{\mathbf{f}}\|^2}\widehat{\mathbf{f}}\widehat{\mathbf{f}}^{\{\mathrm{T}\}} & \frac{b\widehat{\mathbf{f}}}{\|\widehat{\mathbf{f}}\|} \\ \frac{b\widehat{\mathbf{f}}^{\{\mathrm{T}\}}}{\|\widehat{\mathbf{f}}\|} & a \end{bmatrix}, \tag{3.5}$$

where

$$\widehat{\mathbf{u}} = r\mathbf{u}_{t_0} + (1-r)\mathbf{u}_{t_1}, \tag{3.6}$$

$$\widehat{\mathbf{f}} = \mathbf{f}(\widehat{t}, \widehat{\mathbf{u}}), \tag{3.7}$$

$$a = \{\cosh\}\left(\frac{(\mathrm{t}_1 - \mathrm{t}_0)\|\widehat{\mathbf{f}}\|}{\|\widehat{\mathbf{u}}\|}\right), \tag{3.8}$$

$$b = \sinh\left(\frac{(t_1 - t_0)\|\widehat{\mathbf{f}}\|}{\|\widehat{\mathbf{u}}\|}\right). \tag{3.9}$$

That is, we use the initial $\mathbf{u}_{t_0}$ and the final $\mathbf{u}_{t_1}$ through a suitable weighting factor r to calculate **G**, where $0 < r < 1$ is a parameter to be determined, and $\widehat{t} = rt_0 + (1-r)t_1$. The above method results in a Lie group element $\mathbf{G}(r)$ if t_0 and t_1 are fixed values.

3.2. A universal Lie-group element G($\mathbf{u}_{t_0}, \mathbf{u}_{t_1}$)

Let us define a new vector

$$\mathbf{F} := \frac{\widehat{\mathbf{f}}}{\|\widehat{\mathbf{u}}\|}, \tag{3.10}$$

and then (3.5), (3.8), and (3.9) can also be expressed as

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_n + \frac{a-1}{\|\mathbf{F}\|^2}\mathbf{F}\mathbf{F}^{\{\mathrm{T}\}} & \frac{b\mathbf{F}}{\|\mathbf{F}\|} \\ \frac{b\mathbf{F}^{\{\mathrm{T}\}}}{\|\mathbf{F}\|} a & \end{bmatrix}, \tag{3.11}$$

$$a = \{\cosh\}\left[(\mathrm{t}_1 - \mathrm{t}_0)\|\mathbf{F}\|\right], \tag{3.12}$$

$$b = \sinh\left[(t_1 - t_0)\|\mathbf{F}\|\right]. \tag{3.13}$$

From (3.4) and (3.11), it follows that

$$\mathbf{u}_{t_1} = \mathbf{u}_{t_0} + \eta\mathbf{F}, \tag{3.14}$$

$$\|\mathbf{u}_{t_1}\| = a\|\mathbf{u}_{t_0}\| + b\frac{\mathbf{F}\cdot\mathbf{u}_{t_0}}{\|\mathbf{F}\|}, \tag{3.15}$$

where

$$\eta := \frac{(a-1)\mathbf{F}\cdot\mathbf{u}_{t_0} + b\|\mathbf{u}_{t_0}\|\|\mathbf{F}\|}{\|\mathbf{F}\|^2}. \tag{3.16}$$

Equations (3.14) and (3.15) constitute $n+1$ equations, which are both required in the following calculations of three-point BVPs.

From (3.14), we have

$$\mathbf{F} = \frac{1}{\eta}(\mathbf{u}_{t_1} - \mathbf{u}_{t_0}). \tag{3.17}$$

Substituting it for $\mathbf{F}$ into (3.15), we obtain

$$\frac{\|\mathbf{u}_{t_1}\|}{\|\mathbf{u}_{t_0}\|} = a + b\frac{(\mathbf{u}_{t_1} - \mathbf{u}_{t_0})\cdot\mathbf{u}_{t_0}}{\|\mathbf{u}_{t_1} - \mathbf{u}_{t_0}\|\|\mathbf{u}_{t_0}\|}, \tag{3.18}$$

where

$$a = \{\cosh\}\left(\frac{(\mathrm{t}_1 - \mathrm{t}_0)\|\mathbf{u}_{\mathrm{t}_1} - \mathbf{u}_{\mathrm{t}_0}\|}{\mathrm{J}}\right), \tag{3.19}$$

$$b = \sinh\left(\frac{(t_1 - t_0)\|\mathbf{u}_{t_1} - \mathbf{u}_{t_0}\|}{\eta}\right) \tag{3.20}$$

are obtained by inserting (3.17) for $\mathbf{F}$ into (3.12) and (3.13).

Let

$$\cos\theta := \frac{(\mathbf{u}_{t_1} - \mathbf{u}_{t_0})\cdot\mathbf{u}_{t_0}}{\|\mathbf{u}_{t_1} - \mathbf{u}_{t_0}\|\|\mathbf{u}_{t_0}\|}, \tag{3.21}$$

$$S := (t_1 - t_0)\|\mathbf{u}_{t_1} - \mathbf{u}_{t_0}\|, \tag{3.22}$$

and from (3.18)–(3.20) it follows that

$$\frac{\|\mathbf{u}_{t_1}\|}{\|\mathbf{u}_{t_0}\|} = \{\cosh\}\left(\frac{S}{\eta}\right) + \cos\theta\sinh\left(\frac{S}{\eta}\right). \tag{3.23}$$

By defining

$$Z := \exp\left(\frac{S}{\eta}\right), \tag{3.24}$$

from (3.23) we obtain a quadratic equation for Z:

$$(1+\cos\theta)Z^2 - \frac{2\|\mathbf{u}_{t_1}\|}{\|\mathbf{u}_{t_0}\|}Z + 1 - \cos\theta = 0. \tag{3.25}$$

On the other hand, by inserting (3.17) for $\mathbf{F}$ into (3.16) we obtain

$$\|\mathbf{u}_{t_1} - \mathbf{u}_{t_0}\|^2 = (a-1)(\mathbf{u}_{t_1} - \mathbf{u}_{t_0})\cdot\mathbf{u}_{t_0} + b\|\mathbf{u}_{t_0}\|\|\mathbf{u}_{t_1} - \mathbf{u}_{t_0}\|. \tag{3.26}$$

Dividing both sides by $\|\mathbf{u}_{t_0}\|\|\mathbf{u}_{t_1}-\mathbf{u}_{t_0}\|$ and using (3.19)–(3.20) and (3.24) we obtain another quadratic equation for Z:

$$(1+\cos\theta)Z^2-2\left(\cos\theta+\frac{\|\mathbf{u}_{t_1}-\mathbf{u}_{t_0}\|}{\|\mathbf{u}_{t_0}\|}\right)Z+\cos\theta-1=0. \tag{3.27}$$

From (3.25) and (3.27), the solution of Z is found to be

$$Z=\frac{(\cos\theta-1)\|\mathbf{u}_{t_0}\|}{\cos\theta\|\mathbf{u}_{t_0}\|+\|\mathbf{u}_{t_1}-\mathbf{u}_{t_0}\|-\|\mathbf{u}_{t_1}\|}. \tag{3.28}$$

Then from (3.24), we obtain

$$\eta=\frac{(t_1-t_0)\|\mathbf{u}_{t_1}-\mathbf{u}_{t_0}\|}{\ln Z}. \tag{3.29}$$

Through the above discussions and from (3.10) and (3.14), we arrive at an important result.

Theorem 3.1. *For the ODEs in* (2.1), *between any two points* $\mathbf{u}_{t_0}$ *and* $\mathbf{u}_{t_1}$, *there exists a Lie-group shooting equation, which is given by*

$$\mathbf{u}_{t_1}=\mathbf{u}_{t_0}+\frac{\eta}{\|\hat{\mathbf{u}}\|}\hat{\mathbf{f}}, \tag{3.30}$$

where $\hat{\mathbf{u}}$ *and* $\hat{\mathbf{f}}$ *are defined by* (3.6) *and* (3.7), *and* η *is defined by* (3.29).

Because η *is uniquely determined by* $\mathbf{u}_{t_0}$ *and* $\mathbf{u}_{t_1}$ *as can be seen from* (3.28) *and* (3.29), *the above Lie-group shooting equation in terms of* t_0, t_1, r *and the vector field* $\mathbf{f}$ *can be used to solve the three-point BVPs. On the other hand, the Lie-group element defined by* (3.11) *is a universal one, because it is independent on the ODEs in* (2.1).

3.3. A simple demonstration of LGSM

The majority of the Lie-group shooting method (LGSM) is coined into a main algebraic equation (3.30). For its application to the second-order two-point BVPs, we refer the reader going to the details in [25]. This equation is the first time that a universal algebraic equation to connect the vector field $\mathbf{f}$ and boundary values $\mathbf{u}_{t_0}$ and $\mathbf{u}_{t_1}$ is derived. In the past open literature, there appeared no such a similar algebraic equation.

In order to explore how the present LGSM work, let us consider a simple two-point boundary value problem investigated by Liu [25]:

$$u''=\frac{3}{2}u^2, \tag{3.31}$$

$$u(0)=4, \qquad u(1)=1. \tag{3.32}$$

$\mathbf{u}_{t_0}$ and $\mathbf{u}_{t_1}$ are, respectively, given by

$$\mathbf{u}_{t_0}=\begin{bmatrix}4\\u'(t_0)\end{bmatrix}, \qquad \mathbf{u}_{t_1}=\begin{bmatrix}1\\u'(t_1)\end{bmatrix}, \tag{3.33}$$

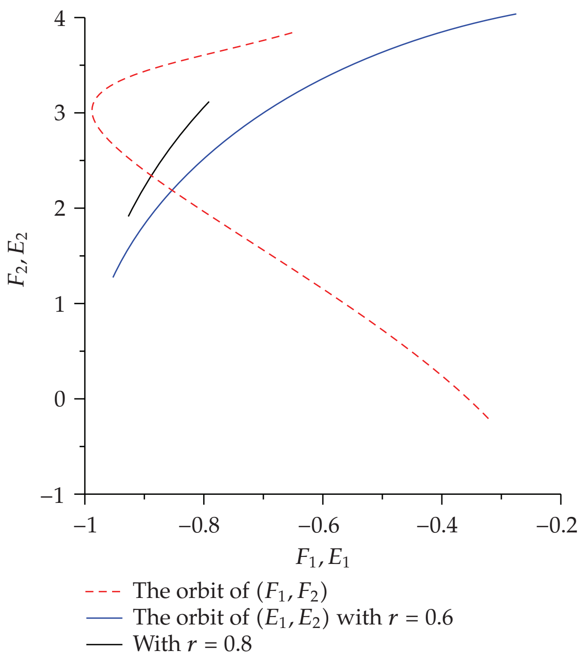

Figure 1: A simple example to demonstrate the orbits of two Lie-groups used in this paper for a two-point boundary value problem.

where $t_0 = 0$ and $t_1 = 1$, but $u'(t_0)$ and $u'(t_1)$ are two unknown values. Now, we can identify $\mathbf{G}(r) = \mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1})$ for some values of r, $u'(t_0)$, and $u'(t_1)$. In view of (3.5) and (3.11), it suffices to identify $\mathbf{F} = \widehat{\mathbf{f}}/\|\widehat{\mathbf{u}}\|$ for some values of r, $u'(t_0)$, and $u'(t_1)$. Because there are many parameters in the above equation, we fix $u'(t_0) = -8$, $r = 0.8$, and $r = 0.6$ for $\widehat{\mathbf{f}}/\|\widehat{\mathbf{u}}\|$, whose two components are denoted by E_1 and E_2 in Figure 1 for simple notations. Now, we let $u'(t_1)$ run from -10 to 10, and the calculated orbit of $\mathbf{F} = (F_1, F_2)$ is plotted in Figure 1 by the dashed-line, while the orbits of (E_1, E_2) are plotted in Figure 1 by a thick line for $r = 0.8$ and a thin line for $r = 0.6$. At the intersection points, we have $\mathbf{G}(r) = \mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1})$; however, we should stress that the representation of $\mathbf{G}(r)$ by (3.5) is obtained by an approximation, even $\mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1})$ has an exact form. Therefore, the above equality of $\mathbf{G}(r)$ and $\mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1})$ should be understood as $\mathbf{G}(r) \sim \mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1})$, and we need to solve it to find some missing initial conditions.

4. Algebraic equations to solve six unknowns

The three-point BVPs considered here give information at an initial time $t = t_0$, at a middle time $t = \xi$, and at a final time $t = t_1$. However, the time-stepping scheme of GPS developed in Section 2 requires a complete information at the starting time $t = t_0$. Some effort is then required to reconcile the time-stepping scheme to the three-point BVPs presented here.

Let $y = dx/dt$. We obtain

$$\dot{x} = y, \tag{4.1}$$

$$\dot{y} = f(t, x, y), \tag{4.2}$$

$$x(t_0) = \alpha, \qquad x(\xi) = \beta, \qquad x(t_1) = \gamma, \tag{4.3}$$

$$y(t_0) = A, \qquad y(\xi) = B, \qquad y(t_1) = C, \tag{4.4}$$

where α, β, γ, A, B, and C are six unknown constants.

Let

$$\mathbf{u} := \begin{bmatrix} x \\ y \end{bmatrix}, \qquad \mathbf{f} := \begin{bmatrix} y \\ f(t,x,y) \end{bmatrix}. \tag{4.5}$$

Applying Theorem 3.1 to (4.1) and (4.2) in the interval of $t_0 \le t \le \xi$ and using the first two columns in (4.3) and (4.4), we obtain

$$\beta = \alpha + \frac{\eta_1}{\rho_1}\left[rA + (1-r)B\right], \tag{4.6}$$

$$B = A + \frac{\eta_1}{\rho_1}\widehat{f}_1, \tag{4.7}$$

where

$$\widehat{f}_1 := f(rt_0 + (1-r)\xi, r\alpha + (1-r)\beta, rA + (1-r)B), \tag{4.8}$$

$$\rho_1 := \sqrt{[r\alpha + (1-r)\beta]^2 + [rA + (1-r)B]^2}, \tag{4.9}$$

$$\eta_1 = \frac{(\xi - t_0)\sqrt{(\beta-\alpha)^2 + (B-A)^2}}{\ln Z_1}, \tag{4.10}$$

$$Z_1 = \frac{(\cos\theta_1 - 1)\sqrt{\alpha^2 + A^2}}{\cos\theta_1\sqrt{\alpha^2 + A^2} + \sqrt{(\beta-\alpha)^2 + (B-A)^2} - \sqrt{\beta^2 + B^2}}, \tag{4.11}$$

$$\cos\theta_1 = \frac{\alpha(\beta-\alpha) + A(B-A)}{\sqrt{(\beta-\alpha)^2 + (B-A)^2}\sqrt{\alpha^2 + A^2}}. \tag{4.12}$$

Similarly, applying Theorem 3.1 to (4.1) and (4.2) in the interval of $\xi \le t \le t_1$ and using the last two columns in (4.3) and (4.4), we obtain

$$\gamma = \beta + \frac{\eta_2}{\rho_2}[rB + (1-r)C], \tag{4.13}$$

$$C = B + \frac{\eta_2}{\rho_2}\widehat{f}_2, \tag{4.14}$$

where

$$\widehat{f}_2 := f(r\xi + (1-r)t_1, r\beta + (1-r)\gamma, rB + (1-r)C), \tag{4.15}$$

$$\rho_2 := \sqrt{[r\beta + (1-r)\gamma]^2 + [rB + (1-r)C]^2}, \tag{4.16}$$

$$\eta_2 = \frac{(t_1 - \xi)\sqrt{(\gamma-\beta)^2 + (C-B)^2}}{\ln Z_2}, \tag{4.17}$$

$$Z_2 = \frac{(\cos\theta_2 - 1)\sqrt{\beta^2 + B^2}}{\cos\theta_2\sqrt{\beta^2 + B^2} + \sqrt{(\gamma-\beta)^2 + (C-B)^2} - \sqrt{\gamma^2 + C^2}}, \tag{4.18}$$

$$\cos\theta_2 = \frac{\beta(\gamma-\beta) + B(C-B)}{\sqrt{(\gamma-\beta)^2 + (C-B)^2}\sqrt{\beta^2 + B^2}}. \tag{4.19}$$

At the same time, (1.2) and (1.3) can be written as

$$H_1(\alpha, A, \beta, B, \gamma, C) = 0, \tag{4.20}$$

$$H_2(\alpha, A, \beta, B, \gamma, C) = 0. \tag{4.21}$$

Therefore, we have six equations (4.6), (4.7), (4.13), (4.14), (4.20), and (4.21) to solve six unknowns α, β, γ, A, B, and C.

For a specified r and a given vector field f, (4.6), (4.7), (4.13), (4.14), (4.20), and (4.21) can be used to generate a new set of α, β, γ, A, B, and C, by starting from an initial guess of these values. We repeat the above process until $(\alpha, \beta, \gamma, A, B, C)$ converge according to a given stopping criterion ϵ. If α and A are available, we can return to (1.4)–(1.7) and integrate them by a forward integration scheme as the GPS derived in Section 2, or other available numerical integrators, like as the fourth-order Runge-Kutta method (RK4).

A suitable r can be determined as follows. For the obtained α and A, we can integrate (1.4) and (1.5) to $t = \xi$ to obtain β and B, and then to $t = t_1$ to obtain γ and C, and then we pick up the best r such that the following result holds:

$$\min_{r\in(0,1)} |H_1(\alpha, A, \beta, B, \gamma, C)| + |H_2(\alpha, A, \beta, B, \gamma, C)|. \tag{4.22}$$

5. Numerical examples

In order to assess the performance of the newly developed method, let us investigate the following examples. We first treat two linear cases but with different boundary conditions at the last two points of time. Then, we consider nonlinear cases, which are subjecting to complex boundary conditions and may have multiple solutions.

5.1. Example 1

Let us consider the following three-point BVP [33]:

$$\ddot{x} = -2, \qquad x(0) = 0, \qquad x(1) = \alpha x(\xi). \tag{5.1}$$

The exact solution is

$$x(t) = \frac{1 - \alpha\xi^2}{1 - \alpha\xi} t - t^2. \tag{5.2}$$

We fix $\alpha = 3$ and $\xi = 0.5$, and take $[0.1, 0.5]$ to be the range of r for picking up the best r as shown in Figure 2(a) at the minimal point. In Figure 2(b), we compare the numerical result with exact solution. It can be seen that the numerical error of x is in the order of 10^{-5} as shown in Figure 2(c).

5.2. Example 2

For the following three-point BVP [34]:

$$\ddot{x} = -\cos t, \qquad x(0) = 0, \qquad 3x(1/3) + 2\dot{x}(1) = 0, \tag{5.3}$$

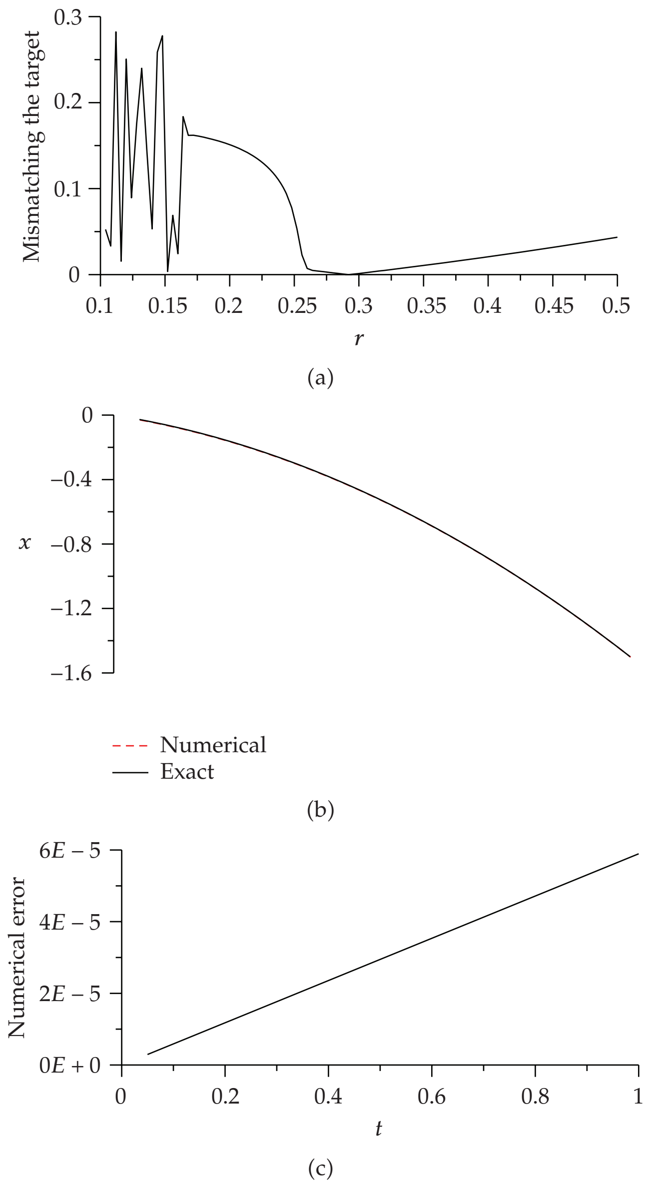

Figure 2: For example 1: (a) showing the error of mismatching, (b) comparing numerical and exact solutions, and (c) displaying the error.

the exact solution is

$$x(t) = \frac{2}{3} t \sin 1 - t \cos \frac{1}{3} + t + \cos t - 1. \tag{5.4}$$

We first take $(0, 1)$ to be the range of r, where the profile of mismatching the target is plotted in Figure 3(a). Then, we use a finer range with $r \in [0.81225, 0.81228]$ to search our numerical solution. Under this condition, we find that the initial slopes A are, respectively, 0.61602 in exact and 0.61611 in numerical, showing that the present TSLGSM can provide very accurate estimation of unknown initial condition of slope. In Figure 3(b), we compare the numerical result with exact solution, where the numerical error of x is in the order of 10^{-5} as can be seen from Figure 3(c).

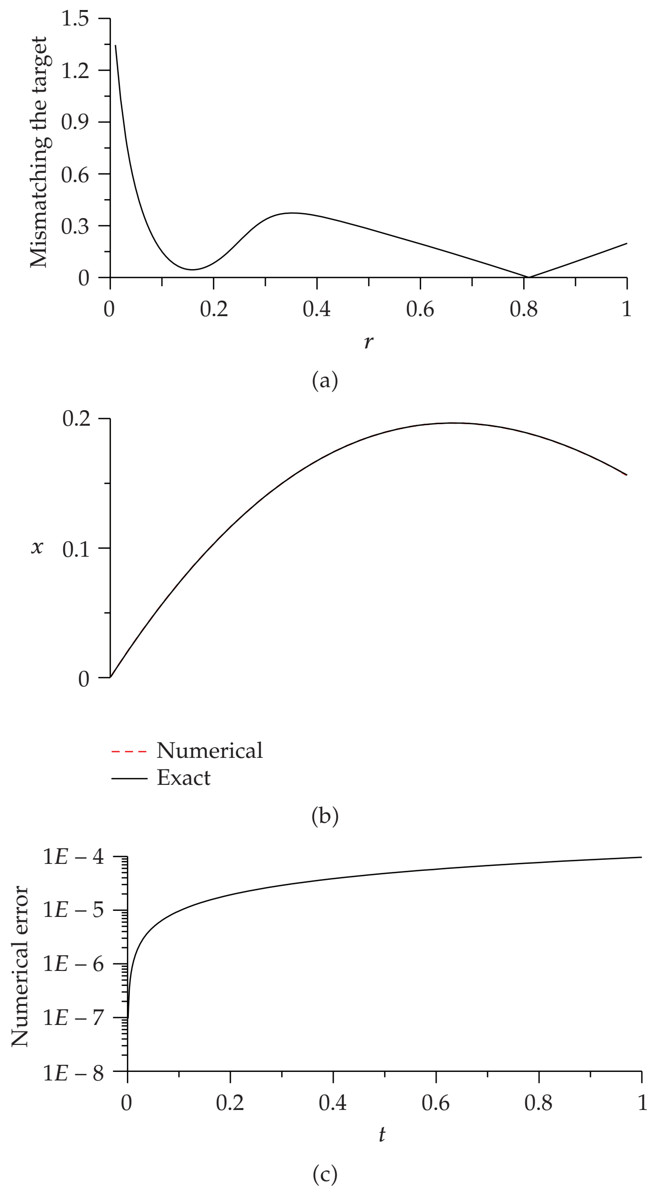

Figure 3: For example 2: (a) showing the error of mismatching, (b) comparing numerical and exact solutions, and (c) displaying the error.

5.3. Example 3

For the following nonlinear three-point BVP:

$$\ddot{x} = -2.25x - (x - 1.5\sin t)^3 + 2\sin t, \tag{5.5}$$

$$x(0) = 0, \qquad x(1) = \frac{1.59941\sin 1 - 0.00004\sin 3}{1.59941\sin 0.5 - 0.00004\sin 1.5}x(0.5), \tag{5.6}$$

the exact solutions are

$$x(t) = 1.59941\sin t - 0.00004\sin 3t, \tag{5.7}$$

$$y(t) = 1.59941\cos t - 0.00012\cos 3t. \tag{5.8}$$

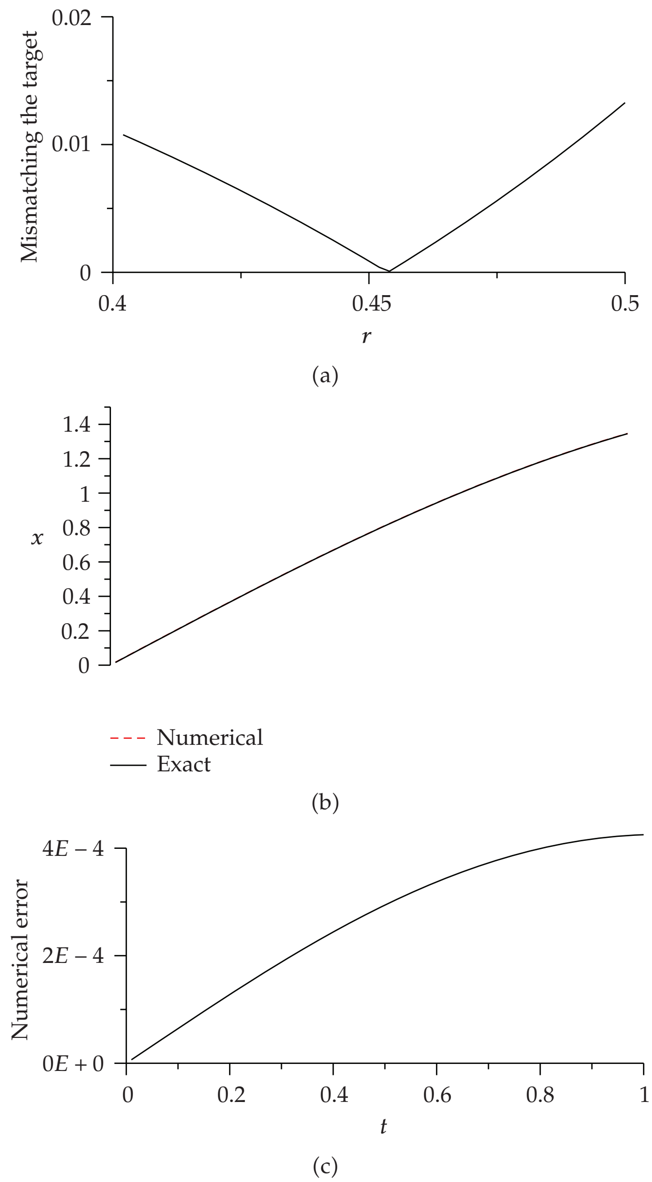

Figure 4: For example 3: (a) showing the error of mismatching, (b) comparing numerical and exact solutions, and (c) displaying the error.

We attempt to search a missing initial condition $y(0) = A$, such that in the numerical solutions of

$$\dot{x} = y, \qquad x(0) = 0, \tag{5.9}$$

$$\dot{y} = -2.25x - (x - 1.5\sin t)^3 + 2\sin t, \qquad y(0) = A, \tag{5.10}$$

$x(0.5)$ and $x(1)$ can match the second equation in (5.6).

We take $[0.4, 0.5]$ as the range of r, where the root of r is located. The iterative process is converged through 32 iterations under a tolerance error of $\epsilon = 10^{-5}$. The mismatching of target is plotted in Figure 4(a) with respect to r.

In Figure 4(b), we compare the exact solution with numerical result calculated by TSLGSM using a time-step size $\Delta t = 0.01$ second. It can be seen that the numerical error of x is in the order of 10^{-4} as shown in 4(c).

5.4. Example 4

For the following three-point BVP:

$$\ddot{x} = \frac{1}{8}(32 + 2t^3 - x\dot{x}), \tag{5.11}$$

$$x(1) = 17, \qquad x(2) + x(3) = \frac{79}{3}, \tag{5.12}$$

the exact solution is

$$x(t) = t^2 + \frac{16}{t}. \tag{5.13}$$

For this example it is hard to directly use the algebraic equations in Section 4 to solve the unknowns β, γ, A, B, and C, and instead of, from (4.6), (4.7), (4.13), (4.14), and (5.12) a matrix equation follows:

$$\begin{bmatrix} 1 & 0 & -\dfrac{r\eta_1}{\rho_1} & -\dfrac{(1-r)\eta_1}{\rho_1} & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 1 & -1 & 0 & \dfrac{r\eta_2}{\rho_2} & \dfrac{(1-r)\eta_2}{\rho_2} \\ 0 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \\ A \\ B \\ C \end{bmatrix} = \begin{bmatrix} 17 \\ \dfrac{\eta_1}{\rho_1}\widehat{f}_1 \\ 0 \\ \dfrac{\eta_2}{\rho_2}\widehat{f}_2 \\ \dfrac{79}{3} \end{bmatrix}. \tag{5.14}$$

Then for each r in the range of $[0.3, 0.4]$, we iteratively solve the above equation by using the conjugate gradient method to find the inverse of the system matrix.

In Figure 5(a), we compare the numerical result with exact solution. It can be seen that the numerical error of x is in the order of 10^{-3} as shown in Figure 5(b).

Instead of (5.12), we now consider a more complex boundary values problem with

$$x(1) + x(2) = 29, \tag{5.15}$$

$$x(2) + x(3) + \dot{x}(3) = \frac{275}{9}. \tag{5.16}$$

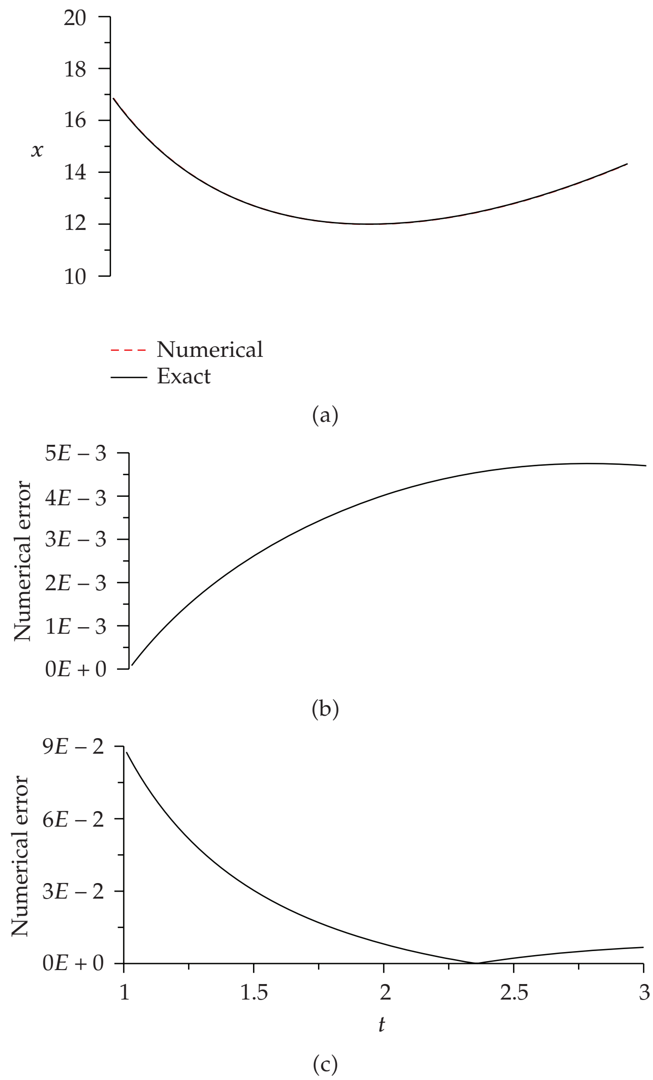

Figure 5: For example 4: (a) comparing numerical and exact solutions, (b) and (c) displaying the errors for two different boundary conditions.

From (4.6), (4.7), (4.13), (4.14), (5.15), and (5.16), it follows that

$$\begin{bmatrix} 1 & -1 & 0 & \dfrac{r\eta_1}{\rho_1} & \dfrac{(1-r)\eta_1}{\rho_1} & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 & \dfrac{r\eta_2}{\rho_2} & \dfrac{(1-r)\eta_2}{\rho_2} \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ A \\ B \\ C \end{bmatrix} = \begin{bmatrix} 0 \\ \dfrac{\eta_1}{\rho_1}\widehat{f}_1 \\ 0 \\ \dfrac{\eta_2}{\rho_2}\widehat{f}_2 \\ 29 \\ \dfrac{275}{9} \end{bmatrix}. \tag{5.17}$$

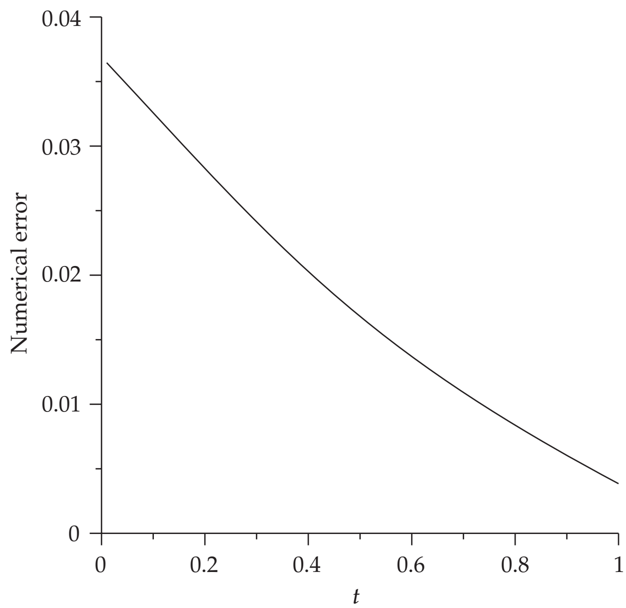

Figure 6: Showing the numerical error for example 5.

Similarly, the conjugate gradient method is used to solve the above equations system to obtain the unknowns. In Figure 5(c), we plot the numerical error of x, which is in the order of 10^{-2}. This case is more difficult than the previous one, because both the initial values of $x(1)$ and $\dot{x}(1)$ are unknown.

5.5. Example 5

We consider the following three-point BVP:

$$\ddot{x} = x^2 + \frac{\dot{x}^2}{(\alpha\pi)^2} - 1 - (\alpha\pi)^2 \sin(\alpha\pi t), \tag{5.18}$$

$$x(0) + x(0.5) + \dot{x}(0.5) = C_1, \tag{5.19}$$

$$x(0.5) + x(1) + +\dot{x}(1) = C_2, \tag{5.20}$$

where C_1 and C_2 can be computed from the exact solution $x(t) = \sin(\alpha\pi t)$.

In Figure 6, we plot the numerical error of x, which is in the order of 10^{-2}. This case is difficult, because both the initial values of $x(0)$ and $\dot{x}(0)$ are unknown and are subjecting to rather complex boundary conditions.

5.6. Example 6

For this three-point BVP, we adopt an example from Kwong and Wong [35]:

$$\ddot{x} + \frac{x^2}{1+x} = 0, \tag{5.21}$$

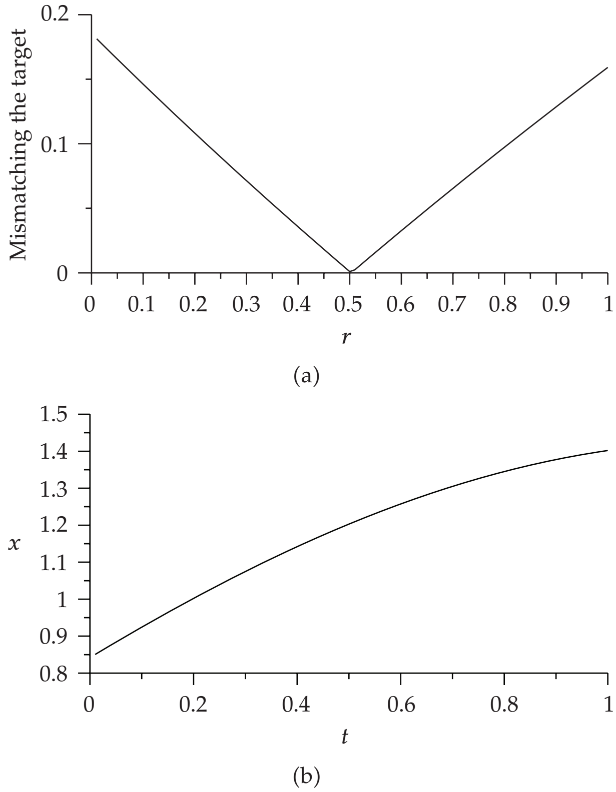

Figure 7: For example 6: (a) showing the error of mismatching, and (b) showing a numerical solution.

with the following boundary conditions:

$$x(0) - \dot{x}(0) = 0, \tag{5.22}$$

$$x(1) - \frac{1}{3}x(0.5) = b. \tag{5.23}$$

When b is smaller than a critical value b^*, Kwong and Wong [35] have proven that there exist two positive solutions. However, we only find one solution under $b = 1$.

We take $(0, 1)$ as the range of r, where the root of r is located, of which the process is converged through about 10 iterations under a tolerance error of $\epsilon = 10^{-5}$. The mismatching of target is plotted in Figure 7(a) with respect to r. In Figure 7(b), we plot the numerical solution of x, of which the error of $x(0) - \dot{x}(0)$ is found to be very small with -1.1×10^{-16}. This case is difficult, because both the initial values of $x(0)$ and $\dot{x}(0)$ are unknown.

5.7. Example 7

Henderson [36] has identified double solutions of the following three-point BVP:

$$-f(x) = \begin{cases} c_1, & x < \dfrac{b}{\xi}, \\ c_3 x + c_4, & x \in \left[\dfrac{b}{\xi}, c\right], \\ c_2, & x > c, \end{cases} \tag{5.24}$$

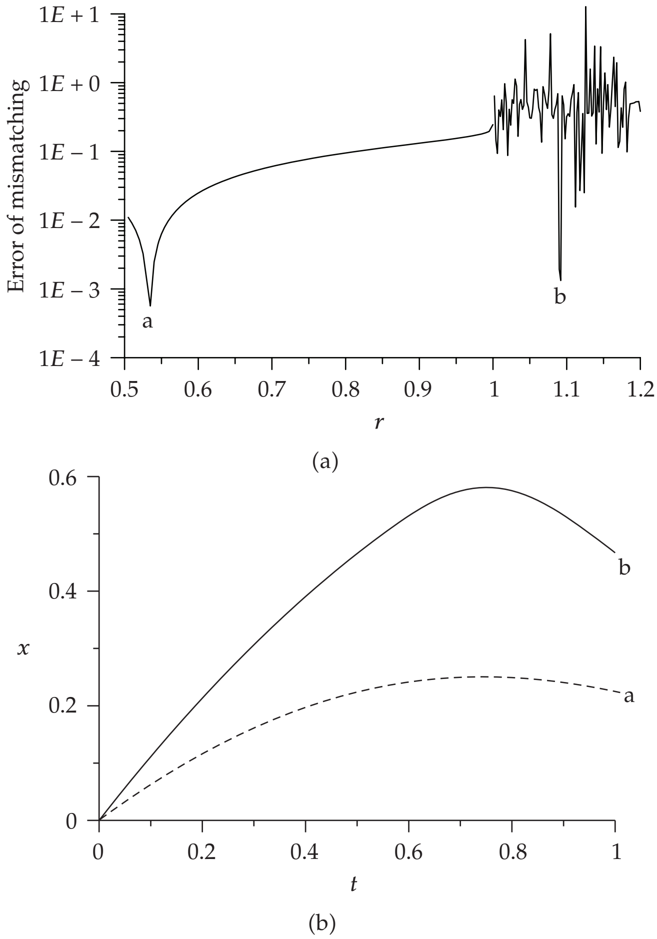

Figure 8: For example 7: (a) showing the error of mismatching in a larger range, and (b) showing two numerical solutions.

where

$$c_1 = \frac{b}{\xi} + \frac{(1-\xi)a}{d[d(1-d)+\xi(d-\xi)]},$$
$$c_2 = 1 + \frac{2c}{\xi(1-\xi)},$$
$$c_3 = \frac{c_2 - c}{c - b/\xi},$$
$$c_4 = c_2 - cc_3. \tag{5.25}$$

Under the following conditions:

$$x(0) = 0, \qquad x(\xi) = x(1), \tag{5.26}$$

$$0 < a < \frac{d[d(1-d)+\xi(d-\xi)]b}{\xi(1-\xi)} < \frac{d[d(1-d)+\xi(d-\xi)]c}{1-\xi}, \tag{5.27}$$

the three-point BVP in (1.1) with the above f has at least two positive solutions.

We consider $\xi = 0.5$, $a = 0.1$, $b = 0.25$, $c = 0.6$, and $d = 0.5$, which satisfy the above conditions. In Figure 8(a), we plot the error of mismatching the target with respect to r in the range of $(0, 1.2)$. It can be seen that there are two minimal points as marked by points a and b. When the ranges for minima are identified, we can pick up more correct value of r by searching the minima in a finer range. When the missing initial conditions are available, we can use the RK4 method to integrate (1.4)–(1.7). There appear two solutions which are marked by a and b in Figure 8(b). For the smaller solution marked by a, it has an exact value of the initial slope with $A = 0.675$, and our numerical value of A is 0.67463 by searching the target in the range of $r \in [0.532, 0.534]$, and the error of mismatching the target is small up to 1.17038×10^{-6} by using $r = 0.53352$. It can be seen that the TSLGSM can provide very accurate numerical result. About the larger solution marked by b, it is interesting that the numerical solution is found outside the range of $r \in (0, 1)$. Correspondingly, the error of mismatching the target is slightly larger with a value of 1.93613×10^{-3} by using $r = 1.09$. In the range of $r > 1$, the present numerical method converges slower than that in the range of $r \in (0, 1)$.

6. Conclusions

In this paper, there were two important points deserved further notifications. The first was the construction of a one-step Lie-group element $\mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1})$, which is a universal Lie-group transformation between initial conditions and final conditions in the augmented Minkowski space, independent on the ODEs we consider. Then, another one was the use of a midpoint rule to construct another Lie-group element $\mathbf{G}(r)$. In order to estimate the missing initial conditions for three-point boundary value problems, we have employed the equation $\mathbf{G}(\mathbf{u}_{t_0}, \mathbf{u}_{t_1}) = \mathbf{G}(r)$ in two-stage of two consecutive intervals to derive four extra algebraic equations, which together with the two given nonlocal boundary conditions lead to totally six equations to solve the six unknowns. Therefore, we can solve them iteratively in a compact space of $r \in (0, 1)$. Numerical examples were examined to ensure that the new approach has a fast convergent speed on the selection of r in a preselected range smaller than $(0, 1)$, which usually required only a few number of iterations. The numerical solution could match the specified boundary conditions very well. The present TSLGSM is also workable to find multiple solutions if the considered equation has. Through this study, it can be concluded that the new TSLGSM is accurate, effective and stable, suggesting it to be useful in a numerical solution of three-point BVPs of second-order ODEs.

Acknowledgment

Taiwan's National Science Council project NSC-96-2221-E-019-027-MY3 granted to the author is highly appreciated.

References

[1] M. Kubicek and V. Hlavacek, *Numerical Solution of Nonlinear Boundary Value Problems with Applications*, Prentice-Hall, New York, NY, USA, 1983.

[2] J. R. Cash, "Numerical integration of nonlinear two-point boundary value problems using iterated deferred corrections—I: a survey and comparison of some one-step formulae," *Computers & Mathematics with Applications*, vol. 12, no. 10, part 1, pp. 1029–1048, 1986.

[3] J. R. Cash, "On the numerical integration of nonlinear two-point boundary value problems using iterated deferred corrections—II: the development and analysis of highly stable deferred correction formulae," *SIAM Journal on Numerical Analysis*, vol. 25, no. 4, pp. 862–882, 1988.

[4] J. R. Cash and R. W. Wright, "Continuous extensions of deferred correction schemes for the numerical solution of nonlinear two-point boundary value problems," *Applied Numerical Mathematics*, vol. 28, no. 2–4, pp. 227–244, 1998.
[5] H. B. Keller, *Numerical Methods for Two-Point Boundary Value Problems*, Dover, New York, NY, USA, 1992.
[6] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, vol. 13 of *Classics in Applied Mathematics*, SIAM, Philadelphia, Pa, USA, 1995.
[7] E. Deeba, S. A. Khuri, and S. Xie, "An algorithm for solving boundary value problems," *Journal of Computational Physics*, vol. 159, no. 2, pp. 125–138, 2000.
[8] V. K. Garg, "Improved shooting techniques for linear boundary value problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 22, no. 1, pp. 87–99, 1980.
[9] S. N. Ha, "A nonlinear shooting method for two-point boundary value problems," *Computers & Mathematics with Applications*, vol. 42, no. 10-11, pp. 1411–1420, 2001.
[10] S. N. Ha and C. R. Lee, "Numerical study for two-point boundary value problems using Green's functions," *Computers & Mathematics with Applications*, vol. 44, no. 12, pp. 1599–1608, 2002.
[11] S. Cuomo and A. Marasco, "A numerical approach to nonlinear two-point boundary value problems for ODEs," *Computer & Mathematics with Applications*, vol. 55, no. 11, pp. 2476–2489, 2008.
[12] M. Cakir and G. M. Amiraliyev, "Numerical solution of a singularly perturbed three-point boundary value problem," *International Journal of Computer Mathematics*, vol. 84, no. 10, pp. 1465–1481, 2007.
[13] A. L. Marhoune, "A three-point boundary value problem with an integral two-space-variables condition for parabolic equations," *Computers & Mathematics with Applications*, vol. 53, no. 6, pp. 940–947, 2007.
[14] B. Calvert and C. P. Gupta, "Existence and uniqueness of solutions to a super-linear three-point boundary-value problem," *Electronic Journal of Differential Equations*, vol. 2005, no. 19, pp. 1–21, 2005.
[15] Y. Gao and M. Pei, "Solvability for two classes of higher-order multi-point boundary value problems at resonance," *Boundary Value Problems*, vol. 2008, Article ID 723828, 14 pages, 2008.
[16] Y. An and R. Ma, "Global behavior of the components for the second order m-point boundary value problems," *Boundary Value Problems*, vol. 2008, Article ID 254593, 10 pages, 2008.
[17] Y. Sun and X. Zhang, "Existence of symmetric positive solutions for an m-point boundary value problem," *Boundary Value Problems*, vol. 2007, Article ID 79090, 14 pages, 2007.
[18] W. Jiang and F. Li, "Several existence theorems of monotone positive solutions for third-order multipoint boundary value problems," *Boundary Value Problems*, vol. 2007, Article ID 17951, 9 pages, 2007.
[19] C. Bai, "Existence of positive solutions for fourth-order three-point boundary value problems," *Boundary Value Problems*, vol. 2007, Article ID 68758, 10 pages, 2007.
[20] C. P. Gupta, "Solvability of a three-point nonlinear boundary value problem for a second order ordinary differential equation," *Journal of Mathematical Analysis and Applications*, vol. 168, no. 2, pp. 540–551, 1992.
[21] C. P. Gupta, "A note on a second order three-point boundary value problem," *Journal of Mathematical Analysis and Applications*, vol. 186, no. 1, pp. 277–281, 1994.
[22] M. K. Kwong, "The shooting method and multiple solutions of two/multi-point BVPs of second-order ODE," *Electronic Journal of Qualitative Theory of Differential Equations*, vol. 2006, no. 6, pp. 1–14, 2006.
[23] B. Ahmad, R. Ali Khan, and P. W. Eloe, "Generalized quasilinearization method for a second order three point boundary-value problem with nonlinear boundary conditions," *Electronic Journal of Differential Equations*, vol. 2002, no. 90, pp. 1–12, 2002.
[24] 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.
[25] 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.
[26] C.-S. Liu, "Efficient shooting methods for the second-order ordinary differential equations," *Computer Modeling in Engineering & Sciences*, vol. 15, no. 2, pp. 69–86, 2006.
[27] C.-S. Liu, "The Lie-group shooting method for singularly perturbed two-point boundary value problems," *Computer Modeling in Engineering & Sciences*, vol. 15, no. 3, pp. 179–196, 2006.

[28] J. R. Chang, C.-S. Liu, and C. W. Chang, "A new shooting method for quasi-boundary regularization of backward heat conduction problems," *International Journal of Heat and Mass Transfer*, vol. 50, no. 11-12, pp. 2325–2332, 2007.
[29] C.-S. Liu, "An LGSM to identify nonhomogeneous heat conductivity functions by an extra measurement of temperature," *International Journal of Heat and Mass Transfer*, vol. 51, no. 9-10, pp. 2603–2613, 2008.
[30] J.-S. Liu, "An LGEM to identify time-dependent heat conductivity function by an extra measurement of temperature gradient," *Computers, Materials & Continua*, vol. 7, no. 2, pp. 81–95, 2008.
[31] C.-S. Liu, "Solving an inverse Sturm-Liouville problem by a Lie-group method," *Boundary Value Problems*, vol. 2008, Article ID 749865, 18 pages, 2008.
[32] L. Ying and E. J. Candès, "The phase flow method," *Journal of Computational Physics*, vol. 220, no. 1, pp. 184–215, 2006.
[33] R. Ma, "Positive solutions of a nonlinear three-point boundary-value problem," *Electric Journal of Differential Equations*, vol. 1998, no. 34, pp. 1–8, 1998.
[34] Z. Zhao, "Positive solutions for singular three-point boundary-value problems," *Electronic Journal of Differential Equations*, vol. 2007, no. 156, pp. 1–8, 2007.
[35] M. K. Kwong and J. S. W. Wong, "The shooting method and nonhomogeneous multipoint BVPs of second-order ODE," *Boundary Value Problems*, vol. 2007, Article ID 64012, 16 pages, 2007.
[36] J. Henderson, "Double solutions of three-point boundary-value problems for second-order differential equations," *Electronic Journal of Differential Equations*, vol. 2004, no. 115, pp. 1–7, 2004.