 CMES, vol.15, no.2, pp.69-86, 2006

Efficient Shooting Methods for the Second-Order Ordinary Differential Equations

Chein-Shan Liu[1]

Abstract: In this paper we will study the numerical integrations of second order boundary value problems under the imposed conditions at $t=0$ and $t=T$ in a general setting. We can construct a compact space shooting method for finding the unknown initial conditions. The key point is based on the construction of a one-step Lie group element $\mathbf{G}(\mathbf{u}_0,\mathbf{u}_T)$ and the establishment of a mid-point Lie group element $\mathbf{G}(r)$. Then, by imposing $\mathbf{G}(\mathbf{u}_0,\mathbf{u}_T)=\mathbf{G}(r)$ we can search the missing initial conditions through an iterative solution of the weighting factor $r\in(0,1)$. Numerical examples were examined to convince that the new approach has high efficiency and accuracy with a fast convergence speed by solving r with a half-interval method. Even under a large span of the boundary coordinate, the new method is also applicable by requiring only a few iterations. The method is also extended to the BVP with general boundary conditions.

keyword: One-step group preserving scheme, Boundary value problem, Shooting method, Estimation of missing initial condition.

1 Introduction

A lot of engineering problems can be described by the nonlinear ordinary differential equations (ODEs). When boundary conditions are imposed, the resulting problems are referred to as the boundary value problems (BVPs). Naturally, the solutions of BVPs have to satisfy the boundary conditions, but in many cases this can be a difficult task when one is concerned with the numerical integrations of BVPs. There are many computational methods that have been developed for solving BVPs (Kubicek and Hlavacek, 1983; Cash, 1986, 1988; Cash and Wright, 1998; Keller, 1992; Ascher, Mattheij and Russell, 1995).

In this paper we propose new methods for the integrations of the following second order BVP:

$$\ddot{x}=f(t,x,\dot{x}),\;\; 0<t<T, \tag{1}$$

$$a_0x(0)+b_0\dot{x}(0)=c_0, \tag{2}$$

$$a_Tx(T)+b_T\dot{x}(T)=c_T, \tag{3}$$

where a_0,b_0,c_0 and a_T,b_T,c_T are given constants, and $[0,T]$ is a time interval of our problem. However, in many applications t may represent a spatial coordinate. Since the boundary conditions are specified at two distinct points, this problem is also called the two-point boundary value problem.

If $f(t,x,\dot{x})$ is a linear function of $(x,\dot{x})$, e.g., $f(t,x,\dot{x})=p(t)\dot{x}+q(t)x+r(t)$, we can find the solution of a linear BVP with the assistance of a linear structure of the governed equation and the use of the solutions of two special initial value problems (IVPs). Suppose that $u(t)$ is the unique solution of the following IVP:

$$\ddot{u}=p(t)\dot{u}+q(t)u+r(t), \tag{4}$$

$$u(0)=1,\;\; \dot{u}(0)=0, \tag{5}$$

and that $v(t)$ is the unique solution of the following IVP:

$$\ddot{v}=p(t)\dot{v}+q(t)v+r(t), \tag{6}$$

$$v(0)=0,\;\; \dot{v}(0)=1. \tag{7}$$

Then, a linear combination of $u(t)$ and $v(t)$ is a solution of Eqs. (1)-(3) with f replaced by the above $f(t,x,\dot{x})=p(t)\dot{x}+q(t)x+r(t)$:

$$x(t)=\alpha_1u(t)+\alpha_2v(t), \tag{8}$$

$$\alpha_1=\frac{c_0[a_Tv(T)+b_T\dot{v}(T)]-b_0c_T}{a_0[a_Tv(T)+b_T\dot{v}(T)]-b_0[a_Tu(T)+b_T\dot{u}(T)]}, \tag{9}$$

[1] Department of Mechanical and Mechatronic Engineering, Taiwan Ocean University, Keelung, Taiwan. E-mail: csliu@mail.ntou.edu.tw

$$\alpha_2 = \frac{a_0 c_T - c_0[a_T u(T) + b_T \dot{u}(T)]}{a_0[a_T v(T) + b_T \dot{v}(T)] - b_0[a_T u(T) + b_T \dot{u}(T)]}. \quad (10)$$

The above technique for the linear BVP is known as a linear shooting method (LSM). Unfortunately, the LSM technique cannot be applied to the nonlinear BVP.

From Eqs. (8), (5) and (7) we can get the initial condition $\dot{x}(0) = \alpha_2$. However, we will point out that the initial condition $\dot{x}(0) = \alpha_2$ may deviate from the exact one to a great extent, which will be demonstrated through an example with exact solution given in Section 5.4, where one can see that the LSM is not so good even for the linear BVP. To remedy, there had some improvements reported in the literature (Garg, 1980; Randez, 1993).

The time-stepping techniques developed for the IVPs require the initial conditions of both x and $y = \dot{x}$ for the second order ODEs, such that we can numerically integrate the following IVP step-by-step in a forward direction from $t = 0$ to $t = T$:

$$\dot{x} = y, \quad (11)$$

$$\dot{y} = f(t, x, y), \quad (12)$$

$$x(0) = \alpha, \ \ y(0) = A. \quad (13)$$

The shooting method involves a choice of the missing initial conditions in Eq. (13), which must satisfy the constraint $a_0\alpha + b_0 A = c_0$ in Eq. (2), and the numerical solution at the terminal point must satisfy the constraint $a_T x(T) + b_T y(T) = c_T$ in Eq. (3).

Basically, the shooting method is to assume some unknown initial conditions and to convert the BVP to the IVP. Solve the IVP and compare the solution at the boundary to the known 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 a suitable initial condition may be difficult when the guesses are carried out in an indefinite range. The shooting method is a trial-and-error method and is often very sensitive to the initial guess. All that makes the computation expensive.

Our approach of the above second order BVP is based on the group preserving scheme (GPS) developed by Liu (2001). The GPS method is very effective to deal with ODEs with special structures as shown by Liu (2005) for stiff equations and by Liu (2006a) for ODEs with constraints. Then, Liu (2006b) has developed a one-step GPS method, which is named the Lie-group shooting method, to calculate the multiple solutions of second order ODEs. About the partial differential equations (PDEs), Liu (2006c) has developed the numerical line method together with the GPS to calculate the solutions of Burgers equation. The same strategy is also used by Liu and Ku (2005) to solve the Landau-Lifshitz equation, where an effective combination of GPS and Runge-Kutta method is employed to enhance the stability and accuracy of numerical solutions. On the other hand, in order to effectively solve the backward in time problems of parabolic PDEs, a past cone structure and a backward group preserving scheme have been successfully developed by the author, such that the new numerical methods can be used to solve the backward in time Burgers equation by Liu (2006d), and the backward in time heat conduction equation by Liu, Chang and Chang (2006). Recently, the so-called one-step estimation method based on the Lie-group is developed by Liu (2006e) for the numerical estimation of temperature-dependent heat conductivity, and then Liu (2006f) extends it to a simultaneous estimation of temperature-dependent heat conductivity and heat capacity for the one-dimensional heat conduction problem.

The idea based on 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 (2006b), but restricted to the simpler boundary conditions. The present approach can be applied to the second order BVPs in a rather general setting, of which we can search the missing initial condition through an iterative solution of r in a compact space of $r \in (0,1)$.

This paper is arranged as follows. In the next section we give a brief sketch of the group preserving scheme for ODEs. In Section 3 we explain the construction of a one-step GPS by using the closure property of the Lie group, and combine it with the mid-point rule to construct a single-parameter Lie group in terms of the weighting factor r. In Section 4 we derive a new shooting method to solve BVPs. In Section 5 we use numerical examples to demonstrate the efficiency of the new method. It is known that the solutions of BVPs may be non-unique. In order to treat this sort BVPs we are derived another method in Section 6 and use two numerical examples to test our method. In Section 7 we extend our estimation method of initial conditions to the general boundary conditions. Numerical examples of mixed type boundary conditions are also investigated. Finally, we draw some conclusions in Section 8.

2 Preliminaries

Although we do not know previously the symmetry group of nonlinear differential equations system, Liu (2001) 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:

$$\dot{\mathbf{u}} = \mathbf{f}(\mathbf{u},t), \ \ \mathbf{u} \in \mathbb{R}^n, \ \ t \in \mathbb{R}, \tag{14}$$

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

$$\frac{d}{dt}\mathbf{X} := \frac{d}{dt}\begin{bmatrix} \mathbf{u} \\ \|\mathbf{u}\| \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{n\times n} & \frac{\mathbf{f}(\mathbf{u},t)}{\|\mathbf{u}\|} \\ \frac{\mathbf{f}^{\mathrm{T}}(\mathbf{u},t)}{\|\mathbf{u}\|} & 0 \end{bmatrix}\begin{bmatrix} \mathbf{u} \\ \|\mathbf{u}\| \end{bmatrix}. \tag{15}$$

It is obvious that the first row in Eq. (15) is the same as the original equation (14), but the inclusion of the second row in Eq. (15) gives us a Minkowskian structure of the augmented system for $\mathbf{X}$ satisfying the cone condition:

$$\mathbf{X}^{\mathrm{T}}\mathbf{g}\mathbf{X} = \mathbf{u}\cdot\mathbf{u} - \|\mathbf{u}\|^2 = 0, \tag{16}$$

where

$$\mathbf{g} = \begin{bmatrix} \mathbf{I}_n & \mathbf{0}_{n\times 1} \\ \mathbf{0}_{1\times n} & -1 \end{bmatrix} \tag{17}$$

is a Minkowski metric. $\mathbf{I}_n$ is the identity matrix of order n, and the superscript T stands for the transpose. The cone condition is a natural constraint of the system (15).

Therefore we have an $n+1$-dimensional augmented system:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} \tag{18}$$

with a constraint (16), where

$$\mathbf{A} := \begin{bmatrix} \mathbf{0}_{n\times n} & \frac{\mathbf{f}(\mathbf{u},t)}{\|\mathbf{u}\|} \\ \frac{\mathbf{f}^{\mathrm{T}}(\mathbf{u},t)}{\|\mathbf{u}\|} & 0 \end{bmatrix} \tag{19}$$

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

$$\mathbf{A}^{\mathrm{T}}\mathbf{g} + \mathbf{g}\mathbf{A} = \mathbf{0}. \tag{20}$$

Accordingly, Liu (2001) has developed a group-preserving numerical scheme:

$$\mathbf{X}_{\ell+1} = \mathbf{G}(\ell)\mathbf{X}_\ell, \tag{21}$$

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

$$\mathbf{G}^{\mathrm{T}}\mathbf{g}\mathbf{G} = \mathbf{g}, \tag{22}$$

$$\det \mathbf{G} = 1, \tag{23}$$

$$G_0^0 > 0, \tag{24}$$

where G_0^0 is the 00th component of $\mathbf{G}$.

The Lie group generated from $\mathbf{A} \in so(n,1)$ is known as a proper orthochronous Lorentz group. An exponential mapping of $\mathbf{A}(\ell)$ is given by

$$\exp[\Delta t\mathbf{A}(\ell)] = \begin{bmatrix} \mathbf{I}_n + \frac{(a_\ell-1)}{\|\mathbf{f}_\ell\|^2}\mathbf{f}_\ell\mathbf{f}_\ell^{\mathrm{T}} & \frac{b_\ell\mathbf{f}_\ell}{\|\mathbf{f}_\ell\|} \\ \frac{b_\ell\mathbf{f}_\ell^{\mathrm{T}}}{\|\mathbf{f}_\ell\|} & a_\ell \end{bmatrix}, \tag{25}$$

where

$$a_\ell := \cosh\left(\frac{\Delta t\|\mathbf{f}_\ell\|}{\|\mathbf{u}_\ell\|}\right), \tag{26}$$

$$b_\ell := \sinh\left(\frac{\Delta t\|\mathbf{f}_\ell\|}{\|\mathbf{u}_\ell\|}\right). \tag{27}$$

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 Eq. (21) and taking its first row, we obtain

$$\mathbf{u}_{\ell+1} = \mathbf{u}_\ell + \eta_\ell\mathbf{f}_\ell = \mathbf{u}_\ell + \frac{(a_\ell-1)\mathbf{f}_\ell\cdot\mathbf{u}_\ell + b_\ell\|\mathbf{u}_\ell\|\|\mathbf{f}_\ell\|}{\|\mathbf{f}_\ell\|^2}\mathbf{f}_\ell. \tag{28}$$

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

$$\eta_\ell \geq \frac{\|\mathbf{u}_\ell\|}{\|\mathbf{f}_\ell\|}\left[1 - \exp\left(-\frac{\Delta t\|\mathbf{f}_\ell\|}{\|\mathbf{u}_\ell\|}\right)\right] > 0, \ \ \forall \Delta t > 0, \tag{29}$$

and that Eq. (28) is a group properties preserving scheme for all $\Delta t > 0$.

3 Two Lie-group elements

Applying the scheme (28) to the ODEs in Eq. (14) with a specified initial condition $\mathbf{u}(0)=\mathbf{u}_0$ we can compute the solution $\mathbf{u}(t)$ by the GPS. Assuming that the total time T is divided by K steps, that is, the time stepsize used in the GPS is $\Delta t = T/K$. Starting from an initial augmented condition $\mathbf{X}_0=\mathbf{X}(0)=(\mathbf{u}_0^{\mathrm{T}}, \|\mathbf{u}_0\|)^{\mathrm{T}}$ we want to calculate the value $\mathbf{X}(T)=(\mathbf{u}^{\mathrm{T}}(T), \|\mathbf{u}(T)\|)^{\mathrm{T}}$ at a desired time $t=T$.

By applying Eq. (21) step-by-step we can obtain

$$\mathbf{X}_T=\mathbf{G}_K(\Delta t)\cdots\mathbf{G}_1(\Delta t)\mathbf{X}_0, \tag{30}$$

where $\mathbf{X}_T$ approximates the exact $\mathbf{X}(T)$.

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

$$\mathbf{G}_1^T\mathbf{g}\mathbf{G}_1=\mathbf{g},\ \ \mathbf{G}_2^T\mathbf{g}\mathbf{G}_2=\mathbf{g}. \tag{31}$$

Then, by using the above two equations we have

$$(\mathbf{G}_2\mathbf{G}_1)^T\mathbf{g}\mathbf{G}_2\mathbf{G}_1=\mathbf{G}_1^T\mathbf{G}_2^T\mathbf{g}\mathbf{G}_2\mathbf{G}_1=\mathbf{G}_1^T\mathbf{g}\mathbf{G}_1=\mathbf{g}. \tag{32}$$

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

According to this argument we can prove that $\mathbf{G}_K\cdots\mathbf{G}_1\in SO_o(n,1)$, because of $\mathbf{G}_K,\cdots,\ \mathbf{G}_1\in SO_o(n,1)$. Therefore in $SO_o(n,1)$, there exists an element denoted by $\mathbf{G}$ which is identical to $\mathbf{G}_K\cdots\mathbf{G}_1$. Hence, from Eq. (30) we have

$$\mathbf{X}_T=\mathbf{G}\mathbf{X}_0. \tag{33}$$

This is a one-step Lie-group transformation from $\mathbf{X}_0$ to $\mathbf{X}_T$. However, it is worthwhile to point out that the 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 *The Lie group element* $\mathbf{G}(r)$

In above we have explored the concept of the one-step $\mathbf{G}$. In order to increase the accuracy of our shooting method to search some unknown initial conditions of the BVPs, we can calculate $\mathbf{G}$ by a mid-point rule:

$$\mathbf{G}=\begin{bmatrix} \mathbf{I}_n+\frac{(a-1)}{\|\hat{\mathbf{f}}\|^2}\hat{\mathbf{f}}\hat{\mathbf{f}}^{\mathrm{T}} & \frac{b\hat{\mathbf{f}}}{\|\hat{\mathbf{f}}\|} \\ \frac{b\hat{\mathbf{f}}^{\mathrm{T}}}{\|\hat{\mathbf{f}}\|} & a \end{bmatrix}, \tag{34}$$

where

$$\hat{\mathbf{u}}=r\mathbf{u}_0+(1-r)\mathbf{u}_T, \tag{35}$$

$$\hat{\mathbf{f}}=\mathbf{f}(\hat{\mathbf{u}}), \tag{36}$$

$$a=\cosh\left(\frac{T\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{u}}\|}\right), \tag{37}$$

$$b=\sinh\left(\frac{T\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{u}}\|}\right). \tag{38}$$

That is, we use the initial $\mathbf{u}_0$ and the final $\mathbf{u}_T$ through a suitable weighting factor r to calculate $\mathbf{G}$, where $0<r<1$ is a parameter. The above method results in a Lie group element $\mathbf{G}(r)$ if T is a fixed value.

3.2 *The Lie group element* $\mathbf{G}(\mathbf{u}_0,\mathbf{u}_T)$

Let us define a new vector

$$\mathbf{F}:=\frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{u}}\|}, \tag{39}$$

and then Eqs. (34), (37) and (38) can also be expressed as

$$\mathbf{G}=\begin{bmatrix} \mathbf{I}_n+\frac{a-1}{\|\mathbf{F}\|^2}\mathbf{F}\mathbf{F}^{\mathrm{T}} & \frac{b\mathbf{F}}{\|\mathbf{F}\|} \\ \frac{b\mathbf{F}^{\mathrm{T}}}{\|\mathbf{F}\|} & a \end{bmatrix}, \tag{40}$$

$$a=\cosh(T\|\mathbf{F}\|), \tag{41}$$

$$b=\sinh(T\|\mathbf{F}\|). \tag{42}$$

From Eqs. (33) and (40) it follows that

$$\mathbf{u}_T=\mathbf{u}_0+\eta\mathbf{F}, \tag{43}$$

$$\|\mathbf{u}_T\| = a\|\mathbf{u}_0\| + b\frac{\mathbf{F}\cdot\mathbf{u}_0}{\|\mathbf{F}\|}, \tag{44}$$

where

$$\eta := \frac{(a-1)\mathbf{F}\cdot\mathbf{u}_0 + b\|\mathbf{u}_0\|\|\mathbf{F}\|}{\|\mathbf{F}\|^2}. \tag{45}$$

Eqs. (43) and (44) constitute $n+1$ equations, which are both required in the following calculations of BVPs.
From Eq. (43) we have

$$\mathbf{F} = \frac{1}{\eta}(\mathbf{u}_T - \mathbf{u}_0). \tag{46}$$

Substituting it for **F** into Eq. (44) we obtain

$$\frac{\|\mathbf{u}_T\|}{\|\mathbf{u}_0\|} = a + b\frac{(\mathbf{u}_T-\mathbf{u}_0)\cdot\mathbf{u}_0}{\|\mathbf{u}_T-\mathbf{u}_0\|\|\mathbf{u}_0\|}, \tag{47}$$

where

$$a = \cosh\left(\frac{T\|\mathbf{u}_T-\mathbf{u}_0\|}{\eta}\right), \tag{48}$$

$$b = \sinh\left(\frac{T\|\mathbf{u}_T-\mathbf{u}_0\|}{\eta}\right) \tag{49}$$

are obtained by inserting Eq. (46) for **F** into Eqs. (41) and (42).
Let

$$\cos\theta := \frac{[\mathbf{u}_T-\mathbf{u}_0]\cdot\mathbf{u}_0}{\|\mathbf{u}_T-\mathbf{u}_0\|\|\mathbf{u}_0\|}, \tag{50}$$

$$S := T\|\mathbf{u}_T-\mathbf{u}_0\|, \tag{51}$$

and from Eqs. (47)-(49) it follows that

$$\frac{\|\mathbf{u}_T\|}{\|\mathbf{u}_0\|} = \cosh\left(\frac{S}{\eta}\right) + \cos\theta\sinh\left(\frac{S}{\eta}\right). \tag{52}$$

By defining

$$Z := \exp\left(\frac{S}{\eta}\right), \tag{53}$$

from Eq. (52) we obtain a quadratic equation for Z:

$$(1+\cos\theta)Z^2 - \frac{2\|\mathbf{u}_T\|}{\|\mathbf{u}_0\|}Z + 1 - \cos\theta = 0. \tag{54}$$

The solution is found to be

$$Z = \frac{\frac{\|\mathbf{u}_T\|}{\|\mathbf{u}_0\|} + \sqrt{\left(\frac{\|\mathbf{u}_T\|}{\|\mathbf{u}_0\|}\right)^2 - (1-\cos^2\theta)}}{1+\cos\theta}, \tag{55}$$

and then from Eq. (53) we obtain

$$\eta = \frac{T\|\mathbf{u}_T-\mathbf{u}_0\|}{\ln Z}. \tag{56}$$

Through the above discussions we can arrive at an important result.

Theorem 1: Between any two points $(\mathbf{u}_0, \|\mathbf{u}_0\|)$ and $(\mathbf{u}_T, \|\mathbf{u}_T\|)$ on the cone, there exists a Lie group element $\mathbf{G} \in SO_o(n,1)$ mapping $(\mathbf{u}_0, \|\mathbf{u}_0\|)$ onto $(\mathbf{u}_T, \|\mathbf{u}_T\|)$, which is given by

$$\begin{bmatrix} \mathbf{u}_T \\ \|\mathbf{u}_T\| \end{bmatrix} = \mathbf{G}(\mathbf{u}_0, \mathbf{u}_T)\begin{bmatrix} \mathbf{u}_0 \\ \|\mathbf{u}_0\| \end{bmatrix}, \tag{57}$$

where

$$\mathbf{G}(\mathbf{u}_0,\mathbf{u}_T) = \begin{bmatrix} \mathbf{I}_n + \frac{a-1}{\|\overline{\mathbf{F}}\|^2}\overline{\mathbf{F}}\overline{\mathbf{F}}^{\mathrm{T}} & \frac{b\overline{\mathbf{F}}}{\|\overline{\mathbf{F}}\|} \\ \frac{b\overline{\mathbf{F}}^{\mathrm{T}}}{\|\overline{\mathbf{F}}\|} & a \end{bmatrix}, \tag{58}$$

$$a = \cosh(\|\overline{\mathbf{F}}\|), \tag{59}$$

$$b = \sinh(\|\overline{\mathbf{F}}\|), \tag{60}$$

$$\overline{\mathbf{F}} = \ln Z\frac{\mathbf{u}_T-\mathbf{u}_0}{\|\mathbf{u}_T-\mathbf{u}_0\|}. \tag{61}$$

Because Z is uniquely determined by $\mathbf{u}_0$ and $\mathbf{u}_T$ as can be seen from Eqs. (55) and (50), the above **G** is a Lie group in terms of $\mathbf{u}_0$ and $\mathbf{u}_T$ and is independent on T, which is denoted by $\mathbf{G}(\mathbf{u}_0,\mathbf{u}_T)$. In the below we will use $\mathbf{G}(\mathbf{u}_0,\mathbf{u}_T) = \mathbf{G}(r)$ to derive the governing equations for solving the BVPs.

4 Boundary value problems

Up to this point we have only considered the solutions of differential equations for which the initial conditions are known. However, in many engineering applications the differential equations do not specify the initial conditions, but rather some given boundary conditions.

Let us consider the following second order boundary value problems:

$$\ddot{x} = f(t, x, \dot{x}), \ \ 0 < t < T, \tag{62}$$

$$x(0) = \alpha, \ \ x(T) = \beta. \tag{63}$$

The conditions that a solution to Eqs. (62) and (63) exists should be checked before any numerical scheme is applied; otherwise, a list of meaningless output may be generated. The general conditions are stated in the following theorem (Burden, 1993).

Theorem 2: Suppose the function f in Eq. (62) is continuous and that $\partial f/\partial x$ and $\partial f/\partial \dot{x}$ are continuous in a domain $\mathbb{D} = \{(t, x, \dot{x})|0 \le t \le T, -\infty < x < \infty, -\infty < \dot{x} < \infty\}$. If $\partial f/\partial x > 0$ and $|\partial f/\partial \dot{x}| \le M$ in $\mathbb{D}$, where M is a constant, then the boundary value problem has a unique solution in $\mathbb{D}$.

The BVPs constructed here require information at the initial time $t = 0$ and at a final time $t = T$. However, the time-stepping scheme developed in Section 2 only requires the information at the starting time $t = 0$. Some effort is then required to reconcile the time-stepping scheme with the BVPs presented here.

Let $y = dx/dt$. We obtain

$$\dot{x} = y, \tag{64}$$

$$\dot{y} = f(t, x, y), \tag{65}$$

$$x(0) = \alpha, \ \ x(T) = \beta, \tag{66}$$

$$y(0) = A, \ \ y(T) = B, \tag{67}$$

where A and B are two unknown constants, while α and β are two given constants.

Let

$$\mathbf{u} := \begin{bmatrix} x \\ y \end{bmatrix}. \tag{68}$$

From Eqs. (43), (66) and (67) it follows that

$$\mathbf{F} := \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \frac{1}{\eta} \begin{bmatrix} \beta - \alpha \\ B - A \end{bmatrix}. \tag{69}$$

Starting from an initial guess of (A, B) we use the following equation to calculate η:

$$\eta = \frac{T\sqrt{(\alpha-\beta)^2 + (A-B)^2}}{\ln Z}, \tag{70}$$

in which Z is calculated by

$$Z = \frac{\frac{\sqrt{\beta^2+B^2}}{\sqrt{\alpha^2+A^2}} + \sqrt{\frac{\beta^2+B^2}{\alpha^2+A^2} - (1-\cos^2\theta)}}{1+\cos\theta}, \tag{71}$$

where

$$\cos\theta = \frac{\alpha(\beta-\alpha) + A(B-A)}{\sqrt{(\alpha-\beta)^2 + (A-B)^2}\sqrt{\alpha^2+A^2}}. \tag{72}$$

The above three equations are obtained from Eqs. (56), (55) and (50) by inserting Eq. (68) for $\mathbf{u}$.

When comparing Eq. (69) with Eq. (39), and with the aid of Eqs. (35), (36) and (64)-(67) we obtain

$$A = \frac{1}{\eta\rho}[\rho^2(\beta-\alpha) - (1-r)\eta^2 \hat{f}], \tag{73}$$

$$B = \frac{1}{\eta\rho}[\rho^2(\beta-\alpha) + r\eta^2 \hat{f}], \tag{74}$$

where

$$\hat{f} := f(rT, r\alpha + (1-r)\beta, rA + (1-r)B), \tag{75}$$

$$\rho := \sqrt{[r\alpha + (1-r)\beta]^2 + [rA + (1-r)B]^2}. \tag{76}$$

The above derivation of the governing equations (70)-(76) is stemed from by letting the two $\mathbf{F}$ in Eqs. (39) and (46) be equal, which is essentially identical to the specification of $\mathbf{G}(\mathbf{u}_0, \mathbf{u}_T) = \mathbf{G}(r)$ in terms of the Lie group elements $\mathbf{G}(\mathbf{u}_0, \mathbf{u}_T)$ and $\mathbf{G}(r)$.

For a specified r and a given vector field f, Eqs. (73) and (74) can be used to generate a new (A, B). We repeat the above process in Eqs. (70)-(76) until (A, B) converges according to a given stopping criterion:

$$\sqrt{(A_{i+1} - A_i)^2 + (B_{i+1} - B_i)^2} \le \varepsilon_1. \tag{77}$$

If A is available, we can return to Eqs. (64)-(66) but with merely integrating the following equations by a forward integration scheme as that given in Section 2:

$$\dot{x} = y, \tag{78}$$

$$\dot{y} = f(t,x,y), \tag{79}$$

$$x(0) = \alpha, \tag{80}$$

$$y(0) = A. \tag{81}$$

A suitable r can be determined as follows. Let $x_r(T)$ denote the above solution of x at T. We start from $r = 1/2$ to determine A by Eqs. (70)-(77) and then numerically integrate Eqs. (78)-(81) from $t = 0$ to $t = T$, and compare the end value of $x(T)$ with the exact β. If $|x_{1/2}(T) - \beta|$ is smaller than a given tolerance error ε_2, then the process of finding solution is finished. If the result cannot be accepted, we need to calculate the values of $x(T)$ corresponding to two different $r_1 < 0.5$ and $r_2 > 0.5$, which are denoted by $x_1(T)$ and $x_2(T)$, respectively. If $[x_1(T) - \beta][x_{1/2}(T) - \beta] < 0$, then there exists one root of r between r_1 and 0.5, which renders $x_r(T) - \beta = 0$; otherwise, this root is located between $(0.5, r_2)$. Then, we apply the half-interval method to find a suitable r, which requires us to calculate Eqs. (78)-(81) at each of the calculation of $x_r(T) - \beta$, until $|x_r(T) - \beta|$ is small enough to satisfy the criterion of $|x_r(T) - \beta| \le \varepsilon_2$, where ε_2 is a given error tolerance.

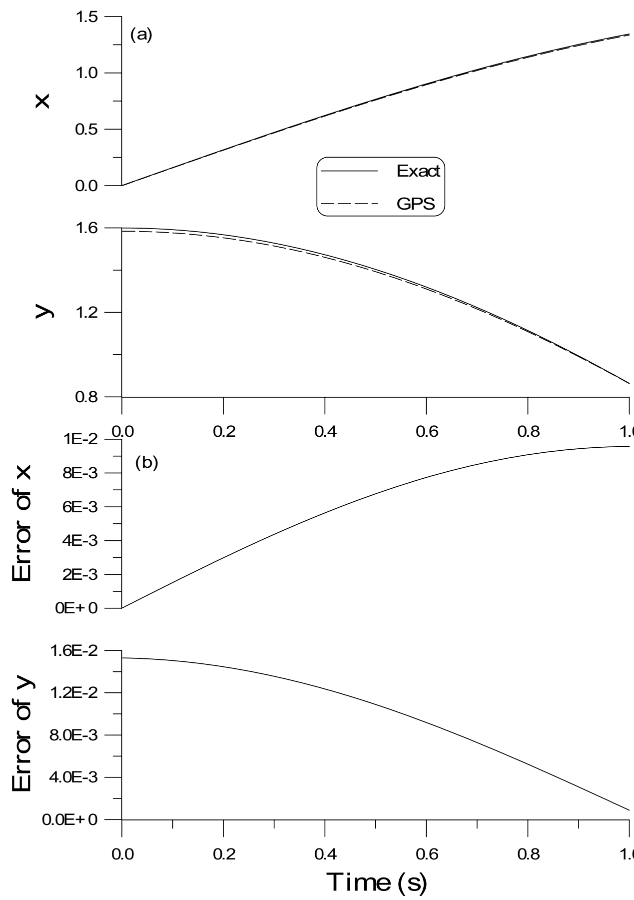

Figure 1 : By using only $r = 0.5$ we compare the numerical solutions and exact solutions for Example 1 in (a), and (b) the numerical errors.

5 Numerical examples

In order to assess the performance of the newly developed method let us investigate the following examples.

5.1 Example 1

For the following BVP:

$$\ddot{x} = -2.25x - (x - 1.5\sin t)^3 + 2\sin t,$$
$$x(0) = 0, \;\; x(1) = 1.59941\sin 1 - 0.00004\sin 3, \tag{82}$$

the exact solutions are

$$x(t) = 1.59941\sin t - 0.00004\sin 3t, \tag{83}$$

$$y(t) = 1.59941\cos t - 0.00012\cos 3t. \tag{84}$$

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

$$\dot{x} = y, \;\; x(0) = 0, \tag{85}$$

$$\dot{y} = -2.25x - (x - 1.5\sin t)^3 + 2\sin t, \;\; y(0) = A, \tag{86}$$

$x(1)$ can match the exact value $x(1) = 1.59941\sin 1 - 0.00004\sin 3$.

We first temporarily take $r = 0.5$ as a trial value in the estimation of A, where under the criterion in Eq. (77) with $\varepsilon_1 = 10^{-15}$ it requires 13 iterations leading to a $A = 1.583997$ with an absolute error 1.5293×10^{-2} with the exact $A = 1.59929$. In Fig. 1 we compare the numerical results with the exact solutions. It can be seen that y is slightly different from the exact one. However, the numerical errors of x and y are in the order of 10^{-2}.

In order to increase the accuracy of the numerical solution, we may need to search another r such that the new A can supply a more accurate initial condition of y. We try another $r = 0.3$ and find that for $r = 0.3$ the numerically integrating result of $x(1)$ is larger than the exact terminal

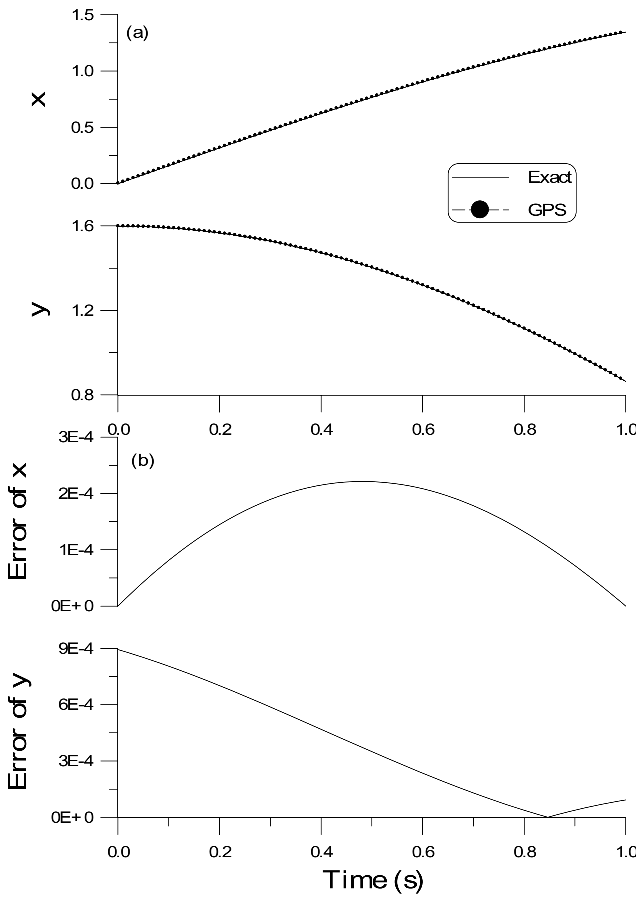

Figure 2 : By iteratively solving r we compare the numerical solutions and exact solutions for Example 1 in (a), and (b) the numerical errors.

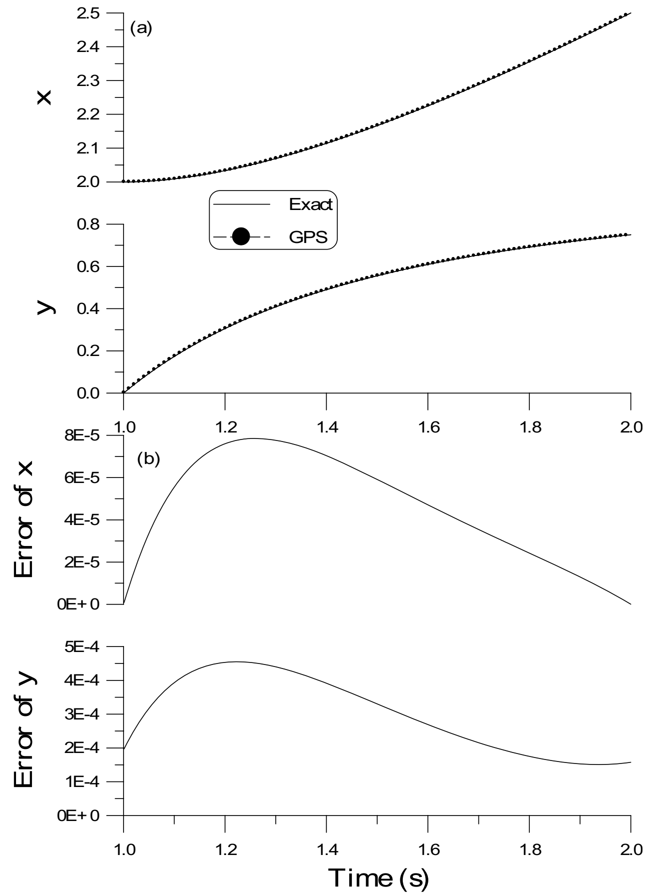

Figure 3 : Comparing the numerical solutions and exact solutions for Example 2 in (a), and (b) the numerical errors.

condition given in Eq. (82), while for $r = 0.5$ the numerically integrating result of $x(1)$ is smaller than the exact terminal condition; hence, the accurate solution of r is between 0.3 and 0.5.

Therefore, we take $[0.3, 0.5]$ as the range of r, where the root of r is located, and apply a half-interval method to search an accurate r with an initial $(A, B) = (1, 2)$, which is converged through 48 iterations under a tolerance error of $\varepsilon_2 = 10^{-15}$.

The final value of x matches very well with the exact value with an error 4.441×10^{-16}. It is indeed a very good shooting technique rendering a fulfillment of the final condition of x. At the same time, the estimated $A = 1.5984$ is rather accurate when compared with the exact $y(0) = 1.5993$, and the error of the end value of y is 9.1954×10^{-5}. In Fig. 2 we compare the exact solutions with the numerical results calculated by GPS using a time stepsize $\Delta t = 0.001$ sec. It can be seen that the numerical errors of x and y are both in the order of 10^{-4}. Unless otherwise specified, the $\Delta t = 0.001$ sec will be used in all the calculations below, and the numerical scheme used for the integrations of ODEs is the GPS in Section 2.

5.2 *Example 2*

Let us consider the following BVP (Ha, 2001):

$$\ddot{x} = 2x^3 - 6x - 2t^3, \;\; x(1) = 2, \;\; x(2) = 2.5. \tag{87}$$

The exact solutions are

$$x(t) = t + \frac{1}{t}, \tag{88}$$

$$y(t) = 1 - \frac{1}{t^2}. \tag{89}$$

We are going to search a missing initial condition $y(1) = A$, such that in the numerical solutions of

$$\dot{x} = y, \ \ x(1) = 2, \tag{90}$$

$$\dot{y} = 2x^3 - 6x - 2t^3, \ \ y(1) = A, \tag{91}$$

$x(2)$ can match the exact value $x(2) = 2.5$.

We take $[0.5, 0.56]$ to be the range of r. In the estimation of A, the criterion in Eq. (77) with $\varepsilon_1 = 10^{-15}$ was used. The initial (A,B) is taken to be $(A,B) = (5,2)$. Then we use a half-interval method to search an accurate r, which is converged through 26 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of x matches very well with the exact value with an error 5.9286×10^{-9}. In Fig. 3 we compare the numerical results with the exact solutions. It can be seen that the numerical error of x is in the order of 10^{-5} and y is in the order of 10^{-4}.

5.3 *Example 3*

Let us consider the following BVP (Ha and Lee, 2002):

$$\ddot{x} = x^2 - 2x + 1 + 2\pi^2 \cos(2\pi t) - (\sin \pi t)^4, \quad x(0) = 1, \ \ x(1) = 1. \tag{92}$$

The exact solutions are

$$x(t) = 1 + (\sin \pi t)^2, \tag{93}$$

$$y(t) = \pi \sin(2\pi t). \tag{94}$$

For this example, we let $[0.24, 0.25]$ be the range of r. In the estimation of A, the criterion in Eq. (77) with $\varepsilon_1 = 10^{-12}$ was used. The initial (A,B) is taken to be $(A,B) = (1,2)$. Then we use a half-interval method to search an accurate r, which is converged through 24 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of x matches very well with the exact value with an error 5.615×10^{-8}. In Fig. 4 we compare the numerical results with the exact solutions. It can be seen that the numerical error of x is in the order of 10^{-3} and y is in the order of 10^{-2}.

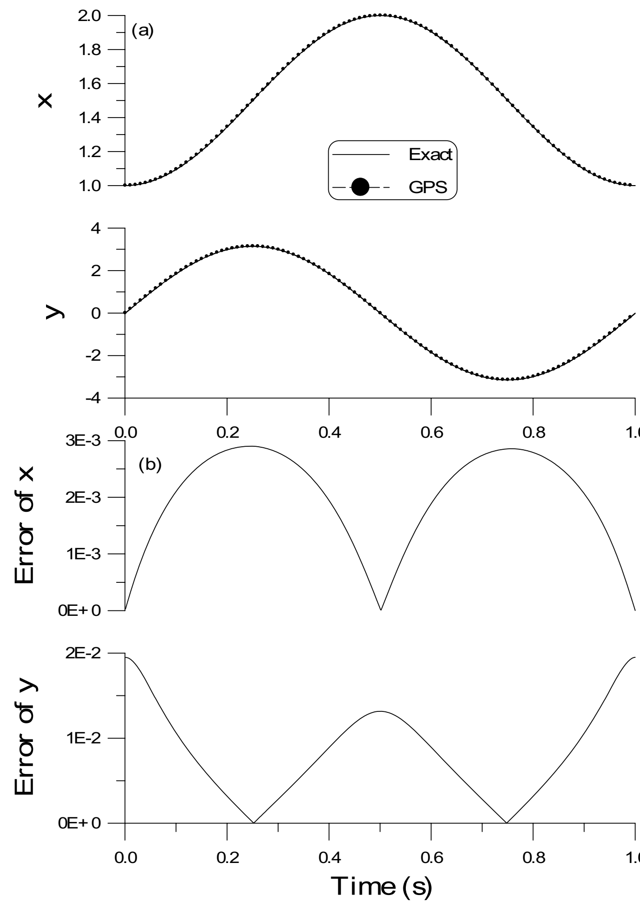

Figure 4 : Comparing the numerical solutions and exact solutions for Example 3 in (a), and (b) the numerical errors.

5.4 *Example 4*

Let us consider the following linear BVP:

$$\ddot{x} = \sqrt{t} \cos t + \left(\frac{1}{4t^2} - 1 \right) x - \frac{1}{t}\dot{x}, \ \ x(1) = 1, \ \ x(6) = -0.5. \tag{95}$$

The exact solution is

$$x(t) = \left[\frac{0.0588713}{\sqrt{t}} + \frac{\sqrt{t}}{4} \right] \cos t + \left[\frac{0.740071}{\sqrt{t}} + \frac{t^{3/2}}{4} \right] \sin t. \tag{96}$$

As mentioned in Section 1, we can find the solution of linear BVP by

$$x(t) = u(t) - \frac{0.5 + u(6)}{v(6)} v(t), \tag{97}$$

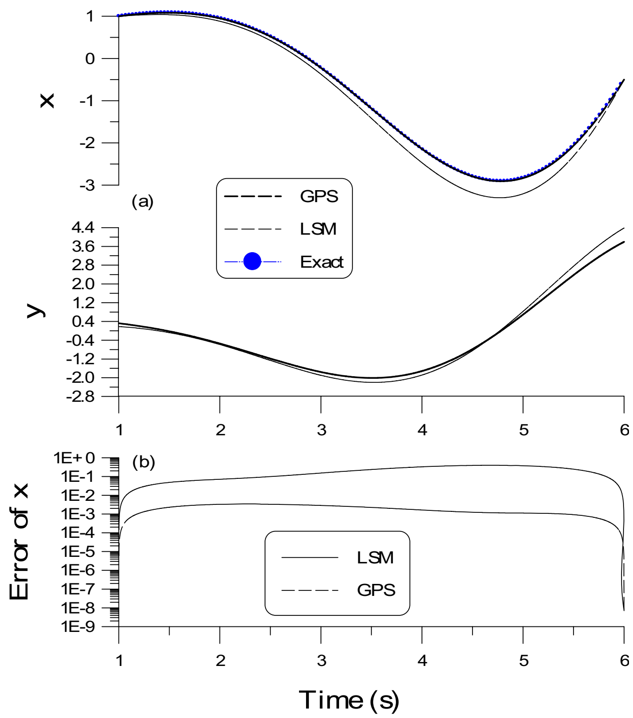

Figure 5 : Comparing the numerical solutions by GPS and LSM with the exact solutions for Example 4 in (a), and (b) the numerical errors of x.

where $u(t)$ and $v(t)$ both satisfy the differential equation but subjecting to the initial conditions with $u(1) = 1$, $\dot{u}(1) = 0$ and $v(1) = 0$, $\dot{v}(1) = 1$. From Eq. (97) it can be seen that $x(t)$ can match not only the differential equation but also the two boundary conditions $x(1) = 1$ and $x(6) = -0.5$ in Eq. (95). This method is called a linear shooting method (LSM).

In contrast, we are going to find a missing initial condition $y(0) = A$, such that in the numerical solutions of

$$\dot{x} = y, \ \ x(1) = 1, \tag{98}$$

$$\dot{y} = \sqrt{t}\cos t + \left(\frac{1}{4t^2} - 1\right)x - \frac{1}{t}y, \ \ y(1) = A, \tag{99}$$

$x(6)$ can match the exact value of $x(6) = -0.5$.

In our estimation we let $[0.75, 0.78]$ as the range of r, and $\varepsilon_1 = 10^{-10}$ was used. The initial (A,B) is taken to be $(A,B) = (-1,0)$. Then, we use a half-interval method to search an accurate r, which is converged through 23 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of x matches very well with the exact value with an error 1.3191×10^{-8}. In Fig. 5 we compare the numerical results with the exact solution. It can be seen that the numerical error of x is in the order of 10^{-3}. At the same figure we also plotted the result computed by the LSM, where we use GPS to integrate the solutions of u and v. Even the LSM solution can match the boundary conditions very well; however, its solution is not good when compared with the exact solution given in Eq. (97). The main reason is that the LSM method gives no correction of the initial slope of x, which makes that $\dot{x}(1) = -[0.5+u(6)]/v(6) = 0.18574$ is not accurate when compared with the exact $\dot{x}(1) = 0.33084$. The error 1.451×10^{-1} is much larger than our 6.37×10^{-3}.

This example shows that our estimation method of initial condition is effective even for a large time span.

5.5 *Example 5*

In this example we consider

$$\ddot{u} = -\delta e^{u}, \ \ u(0) = 0, \ \ u(1) = 0, \tag{100}$$

which is referred as the Bratu problem (Aris, 1975). It was noted by Ascher, Mattheij and Russel (1995) that the function

$$u(t) = -2\ln\left[\cosh\left(\frac{0.5(t-0.5)\phi}{0.25\cosh\phi}\right)\right] \tag{101}$$

is a solution of Eq. (100), if ϕ is the solution of $\phi = \sqrt{2\delta}\cosh(\phi/4)$.

From the boundary conditions we have $\alpha = \beta = 0$, which make $\cos\theta$ in Eq. (72) to be -1, and thus Z in Eq. (71) cannot be defined. When apply the method in Section 4 to this problem we consider a translation $x(t) = u(t)+1$, such that we have

$$\ddot{x} = -\delta e^{x-1}, \ \ x(0) = 1, \ \ x(1) = 1. \tag{102}$$

We only consider $\delta = 1$. In the estimation of the initial condition $y(0) = A$, we let $[0.45, 0.48]$ as the range for an iterative solution of r, $\varepsilon_1 = 10^{-10}$ was used, and the initial (A,B) is taken to be $(A,B) = (3,-3)$. Then, we use a half-interval method to search an accurate r, which is converged through 19 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of x matches very well with the exact value with an error 2.6731×10^{-10}. In Table 1 we compare the numerical solutions with the exact solutions at some points, where Error 1 is the

Table 1 : The comparison of numerical solutions with the exact solutions for Example 5.

t	Exact solution	Numerical solution	Error 1	Error 2
0.1	0.04985	0.04984	0.00001	0.00269
0.2	0.08919	0.08918	0.00001	0.00202
0.3	0.11761	0.11760	0.00001	0.00015
0.4	0.13479	0.13479	0	0.00220
0.5	0.14054	0.14054	0	0.00301
0.6	0.13479	0.13479	0	0.00220
0.7	0.11761	0.11762	0.00001	0.00015
0.8	0.08919	0.08920	0.00001	0.00202
0.9	0.04985	0.04985	0	0.00269

error obtained by taking the absolute of the difference between our numerical solution with exact solution, while the Error 2 is obtained by Deeba, Khuri and Xie (2000) using the Adomian decomposition method (Adomian, 1994). It can be seen that the accuracy is largely improved from the third order to the fifth order when apply our method to this problem.

6 Non-unique solutions

The numerical method in Section 4 is obtained by equating the two $\mathbf{F}$ in Eqs. (39) and (46), which results in

$$\mathbf{u}_T = \mathbf{u}_0 + \eta \frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{u}}\|}, \tag{103}$$

where η is fully determined by $\mathbf{u}_0$ and $\mathbf{u}_T$ through Eq. (56). In this section we derive the governing equations for BVPs by another method, and give numerical examples to test the new method.

6.1 An alternative way to derive the algebraic equations

Inserting Eq. (34) into Eq. (33) and taking the first row we obtain

$$\mathbf{u}_T = \mathbf{u}_0 + \hat{\eta}\hat{\mathbf{f}}, \tag{104}$$

$$\hat{\eta} = \frac{(a-1)\hat{\mathbf{f}}\cdot\mathbf{u}_0 + b\|\mathbf{u}_0\|\,\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{f}}\|^2}, \tag{105}$$

where a and b were defined by Eqs. (37) and Eqs. (38). When comparing the above two equations with Eqs. (103) and (56), it can be seen that they have different representations on η and $\hat{\eta}$. η depends only on $\mathbf{u}_0$ and $\mathbf{u}_T$ as just mentioned; however, $\hat{\eta}$ is slightly complex depending on $\mathbf{u}_0$, $\mathbf{u}_T$, $\hat{\mathbf{f}}$ as well as the parameter r. Eq. (103) is linear on $\hat{\mathbf{f}}$, but Eq. (104) is nonlinear on $\hat{\mathbf{f}}$.

From Eqs. (64)-(68), (35), (36) and (104) it follows that

$$\beta = \alpha + \hat{\eta}[rA + (1-r)B], \tag{106}$$

$$B = A + \hat{\eta}\hat{f}, \tag{107}$$

where $\hat{f}$ is defined by Eq. (75). Solving the above equations we obtain

$$A = \frac{\beta-\alpha}{\hat{\eta}} - \hat{\eta}(1-r)\hat{f}, \tag{108}$$

$$B = \frac{\beta-\alpha}{\hat{\eta}} + r\hat{\eta}\hat{f}. \tag{109}$$

For a specified r and a given vector field f, Eqs. (108) and (109) can be used to obtain (A,B) iteratively with

$$\hat{f} = f(rT, r\alpha + (1-r)\beta, rA + (1-r)B), \tag{110}$$

$$\|\mathbf{u}_0\| = \sqrt{\alpha^2 + A^2}, \tag{111}$$

$$\|\hat{\mathbf{u}}\| = \sqrt{[r\alpha + (1-r)\beta]^2 + [rA + (1-r)B]^2}, \tag{112}$$

$$\|\hat{\mathbf{f}}\| = \sqrt{[rA + (1-r)B]^2 + \hat{f}^2}, \tag{113}$$

$$\hat{\mathbf{f}} \cdot \mathbf{u}_0 = \alpha[rA + (1-r)B] + A\hat{f}, \tag{114}$$

$$a = \cosh\left(\frac{T\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{u}}\|}\right), \tag{115}$$

$$b = \sinh\left(\frac{T\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{u}}\|}\right), \tag{116}$$

$$\hat{\eta} = \frac{(a-1)\hat{\mathbf{f}} \cdot \mathbf{u}_0 + b\|\mathbf{u}_0\|\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{f}}\|^2}. \tag{117}$$

The other procedures to obtain the solution of (A,B) are similar to that in Section 4.

6.2 Example 6

Let us consider the following BVP (Ha, 2001):

$$\ddot{x} = \frac{3}{2}x^2, \ \ x(0) = 4, \ \ x(1) = 1. \tag{118}$$

The exact solutions are

$$x(t) = \frac{4}{(1+t)^2}, \tag{119}$$

$$y(t) = \frac{-8}{(1+t)^3}. \tag{120}$$

It needs to stress that the solution of Eq. (118) is not unique. In addition the one in Eq. (119), there exists another solution:

$$x(t) = c_1^2\left(\frac{1-\mathrm{cn}(c_1t-c_2,k^2)}{1+\mathrm{cn}(c_1t-c_2,k^2)} - \frac{1}{\sqrt{3}}\right), \tag{121}$$

where $\mathrm{cn}(\xi,k)$ is the modulus k Jacobi elliptic function. In the above case we have $c_1 = 4.30310990$, $c_1 = 2.3346196$, and $k = \sqrt{2+\sqrt{3}}/2$.

In this problem the vector field $f = 3x^2/2$ cannot satisfy the unique conditions in Theorem 2, since $\partial f/\partial x = 3x$ may be negative, for example the solution in Eq. (121). On the other hand, since f may be zero when x passes the zero axis, we consider a translation of x in Eq. (118) by $z = x+10$, such that one has

$$\ddot{z} = \frac{3}{2}[z^2 - 20z + 100], \ \ z(0) = 14, \ \ z(1) = 11. \tag{122}$$

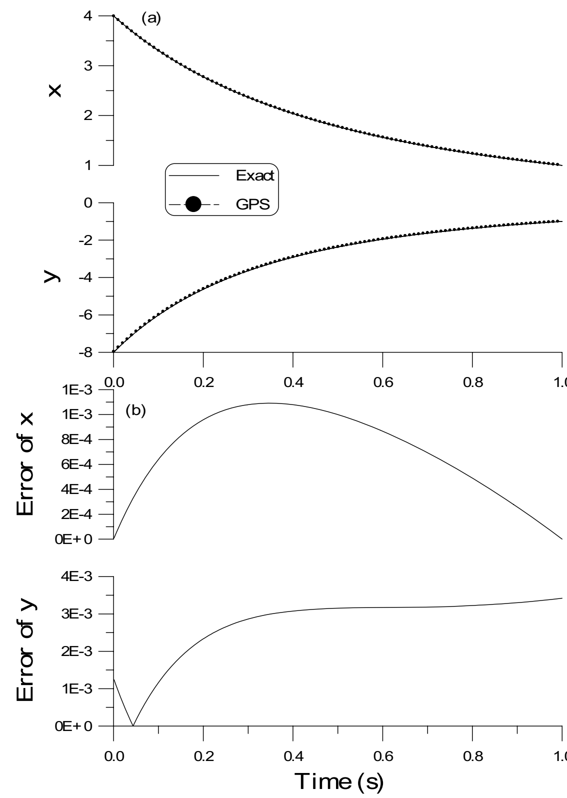

Figure 6 : Comparing the numerical solutions and exact solutions for Example 6 in (a), and (b) the numerical errors.

Then we apply the method in Section 6.1 on the above equation, and then obtain x and y by $x = z - 10$ and $y = \dot{z}$.

In the estimation of A, we take $[0.57, 0.585]$ as the range for an iterative solution of r, and $\varepsilon_1 = 10^{-10}$ was used in the criterion (77). The initial (A,B) is taken to be $(A,B) = (-10,-4)$. Then we use a half-interval method to search an accurate r, which is converged through 21 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of x matches very well with the exact value with an error 5.9227×10^{-8}. In Fig. 6 we compare the numerical results with the exact solutions. It can be seen that the numerical errors of x and y are both in the order of 10^{-3}.

6.3 Example 7

For the following BVP (Ha, 2001):

$$\ddot{x} = \frac{1}{8}(32 + 2t^3 - x\dot{x}), \ \ x(1) = 17, \ \ x(3) = \frac{43}{3}, \tag{123}$$

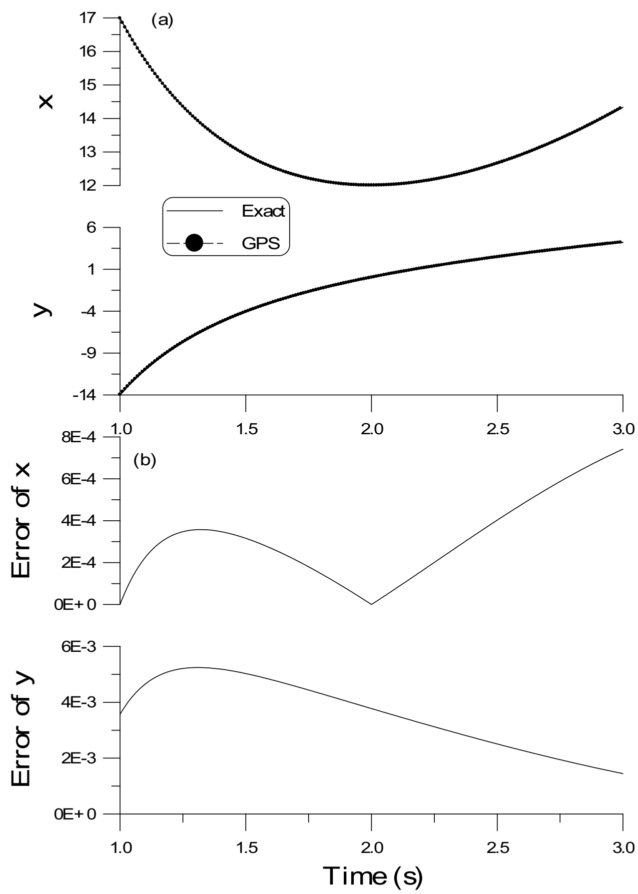

Figure 7 : Comparing the numerical solutions and exact solutions for Example 7 in (a), and (b) the numerical errors.

the exact solutions are

$$x(t) = t^2 + \frac{16}{t}, \tag{124}$$

$$y(t) = 2t - \frac{16}{t^2}. \tag{125}$$

In this problem the vector field $f = (32 + 2t^3 - x\dot{x})/8$ cannot satisfy the unique conditions in Theorem 2, since $\partial f/\partial x = -\dot{x}/8 = -y/8$ may be negative when y is positive. We take two $r = 0.25$ and $r = 0.3$ as a range of r, where the root is located. In the estimation of A, the criterion in Eq. (77) with $\varepsilon_1 = 10^{-10}$ was used. The initial (A,B) is taken to be $(A,B) = (-10,10)$. Then we use a half-interval method to search an accurate r, which is converged through 22 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of x can match the exact value with an error 7.4188×10^{-4}. In Fig. 7 we compare the numerical results with the exact solutions. It can be seen that the numerical error of x is in the order of 10^{-4} and that of y is in the order of 10^{-3}.

7 General boundary conditions

At this moment we have only considered the solution of a specific two-point boundary values problem in Eqs. (78)-(81). However, in a practical application we may encounter two-point boundary values problem with the following more general boundary conditions:

$$a_0 x(0) + b_0 y(0) = c_0, \tag{126}$$

$$a_T x(T) + b_T y(T) = c_T. \tag{127}$$

Because they are two-point boundary values, (a_0, b_0) cannot both be zero, and this is also true for (a_T, b_T).

There are nine cases should be considered, which are obtained by considering the following permutations:

$$\begin{bmatrix} (a_0 \neq 0, b_0 = 0) \\ (a_0 = 0, b_0 \neq 0) \\ (a_0 \neq 0, b_0 \neq 0) \end{bmatrix} \times \begin{bmatrix} (a_T \neq 0, b_T = 0) \\ (a_T = 0, b_T \neq 0) \\ (a_T \neq 0, b_T \neq 0) \end{bmatrix}. \tag{128}$$

In the below we study them seperately.

7.1 Initial conditions estimation technique

For case (i) with $[a_0 \neq 0, b_0 = 0, a_T \neq 0, b_T = 0]$, $x(0) = \alpha$ and $x(T) = \beta$ are two known parameters, and the other two unknown parameters $y(0) = A$ and $y(T) = B$ can be estimated by the method in Section 4.

For case (ii) with $[a_0 \neq 0, b_0 = 0, a_T = 0, b_T \neq 0]$, $x(0) = \alpha$ and $y(T) = B$ are two known parameters, and the other two unknown parameters $y(0) = A$ and $x(T) = \beta$ can be estimated by the following method. We rearrange Eq. (109) to

$$\beta = \alpha + \hat{\eta} B - r\hat{\eta}^2 \hat{f}. \tag{129}$$

Starting from an initial guess of (A, β) we use Eqs. (108)-(117), where Eq. (109) is replaced by Eq. (129), to generate the new (A, β) until they converge according to the following criterion:

$$\sqrt{(A_{i+1} - A_i)^2 + (\beta_{i+1} - \beta_i)^2} \leq \varepsilon_1. \tag{130}$$

Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $y(T) - c_T/b_T = 0$.

For case (iii) with $[a_0 \neq 0, b_0 = 0, a_T \neq 0, b_T \neq 0]$, $x(0) = \alpha$ is a known parameter, and the other three unknown parameters $y(0) = A$, $y(T) = B$ and $x(T) = \beta$ can be estimated by the following method. Substituting

$$\beta = \frac{c_T - b_T B}{a_T} \tag{131}$$

into Eqs. (108)-(117), and starting from an initial guess of (A,B) we can use these equations to generate the new (A,B) until they converge according to the stopping criterion (77). If B is available, we can return to Eq. (131) to calculate β. Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $a_T x(T) + b_T y(T) - c_T = 0$.

For case (iv) with $[a_0 = 0, b_0 \neq 0, a_T \neq 0, b_T = 0]$, $y(0) = A$ and $x(T) = \beta$ are two known parameters, and the other two unknown parameters $x(0) = \alpha$ and $y(T) = B$ can be estimated by the following method. We rearrange Eq. (108) to

$$\alpha = \beta - \hat{\eta} A - (1-r)\hat{\eta}^2 \hat{f}. \tag{132}$$

Starting from an initial guess of (α, B) we use Eqs. (108)-(117), where Eq. (108) is replaced by Eq. (132), to generate the new (α, B) until they converge according to the following criterion:

$$\sqrt{(\alpha_{i+1} - \alpha_i)^2 + (B_{i+1} - B_i)^2} \leq \varepsilon_1. \tag{133}$$

Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $x(T) - c_T/a_T = 0$.

For case (v) with $[a_0 = 0, b_0 \neq 0, a_T = 0, b_T \neq 0]$, $y(0) = A$ and $y(T) = B$ are two known parameters, and the other two unknown parameters $x(0) = \alpha$ and $x(T) = \beta$ can be estimated by the following method. We rearrange Eqs. (108) and (109) to

$$\alpha = \alpha + A - B + \hat{\eta} \hat{f}, \tag{134}$$

$$\beta = \alpha + \hat{\eta}[rA + (1-r)B]. \tag{135}$$

Starting from an initial guess of (α, β) we use Eqs. (108)-(117), where Eqs. (108) and (109) are replaced by Eqs. (134) and (135), to generate the new (α, β) until they converge according to the following criterion:

$$\sqrt{(\alpha_{i+1} - \alpha_i)^2 + (\beta_{i+1} - \beta_i)^2} \leq \varepsilon_1. \tag{136}$$

Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $y(T) - c_T/b_T = 0$.

For case (vi) with $[a_0 = 0, b_0 \neq 0, a_T \neq 0, b_T \neq 0]$, $y(0) = A$ is a known parameter, and the other three parameters $x(0) = \alpha$, $x(T) = \beta$ and $y(T) = B$ are unknown, which can be estimated by the following method. Substituting Eq. (131) for β into Eqs. (108)-(117), where Eq. (109) is replaced by Eq. (132), and starting from an initial guess of (α, B) we can use these equations to generate the new (α, B) until they converge according to the stopping criterion (133). If B is available, we can return to Eq. (131) to calculate β. Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $a_T x(T) + b_T y(T) - c_T = 0$.

For case (vii) with $[a_0 \neq 0, b_0 \neq 0, a_T \neq 0, b_T = 0]$, $x(T) = \beta$ is a known parameter, and the other three unknown parameters $x(0) = \alpha$, $y(0) = A$ and $y(T) = B$ can be estimated by the following method. Substituting

$$\alpha = \frac{c_0 - b_0 A}{a_0} \tag{137}$$

into Eqs. (108)-(117), and starting from an initial guess of (A,B) we can use these equations to generate the new (A,B) until they converge according to the stopping criterion (77). If A is available, we can return to Eq. (137) to calculate α. Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $x(T) - c_T/a_T = 0$.

For case (viii) with $[a_0 \neq 0, b_0 \neq 0, a_T = 0, b_T \neq 0]$, $y(T) = B$ is a known parameter, and the other three unknown parameters $x(0) = \alpha$, $x(T) = \beta$ and $y(0) = A$ can be estimated by the following method. Substituting Eq. (137) for α into Eqs. (108)-(117), where Eq. (109) is replaced by Eq. (129), and starting from an initial guess of (A, β) we can use these equations to generate the new (A, β) until they converge according to the stopping criterion (130). If A is available, we can return to Eq. (137) to calculate α. Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $y(T) - c_T/b_T = 0$.

For case (ix) with $[a_0 \neq 0, b_0 \neq 0, a_T \neq 0, b_T \neq 0]$, there are four unknown parameters $x(0) = \alpha$, $x(T) = \beta$, $y(0) =$

A and $y(T) = B$, which can be estimated by the following method. Substituting Eq. (137) for α and Eq. (131) for β into Eqs. (108)-(117), and starting from an initial guess of (A,B) we can use these equations to generate the new (A,B) until they converge according to the stopping criterion (77). If A and B are available, we can return to Eqs. (137) and (131) to calculate α and β. Then we use the same method in Section 4 to calculate (x,y) until they match the target equation: $a_T x(T) + b_T y(T) - c_T = 0$.

In summary, the nine cases can be grouped into four types according to the unknown variables. While the cases (i), (iii), (vii) and (ix) are with (A,B) as unknowns, the case (v) is with (α,β) as unknowns. While the cases (ii) and (viii) are with (A,β) as unknowns, the cases (iv) and (vi) are with (α,B) as unknowns. On the other hand, the nine cases can be grouped into three types according to the target equations. The cases (i), (iv) and (vii) are with $x(T) - c_T/a_T = 0$ as a target, the cases (ii), (v) and (viii) are with $y(T) - c_T/b_T = 0$ as a target, and the cases (iii), (vi) and (ix) are with $a_T x(T) + b_T y(T) - c_T = 0$ as a target.

7.2 Example 8

Let us consider the following BVP (Chen and Liu, 1998):

$$\ddot{x} - \varepsilon(x^4 - 1) + 1 = 0, \quad x(0) = 1, \quad \dot{x}(1) = 0. \tag{138}$$

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

$$\dot{x} = y, \quad x(0) = 1, \tag{139}$$

$$\dot{y} = \varepsilon(x^4 - 1) - 1, \quad y(0) = A, \tag{140}$$

$y(1)$ can match the exact value of $y(1) = 0$.

This problem is of case (ii) in Section 7.1, and we apply the estimation technique specified there to search A. We first consider $\varepsilon = 0.1$, and take two $r = 0.82$ and $r = 0.83$ as a range of r, where the root is located. In the estimation of (A,β), the criterion in Eq. (130) with $\varepsilon_1 = 10^{-10}$ was used. The initial (A,β) is taken to be $(A,\beta) = (-3,1)$. Then we use a half-interval method to search an accurate r, which is converged through 18 iterations under an error tolerance of $\varepsilon_2 = 10^{-15}$. The final value of y can match the exact value with an error 5.627×10^{-9}. In Fig. 8 we compare the numerical results with the following solutions (Chen and Liu, 1998):

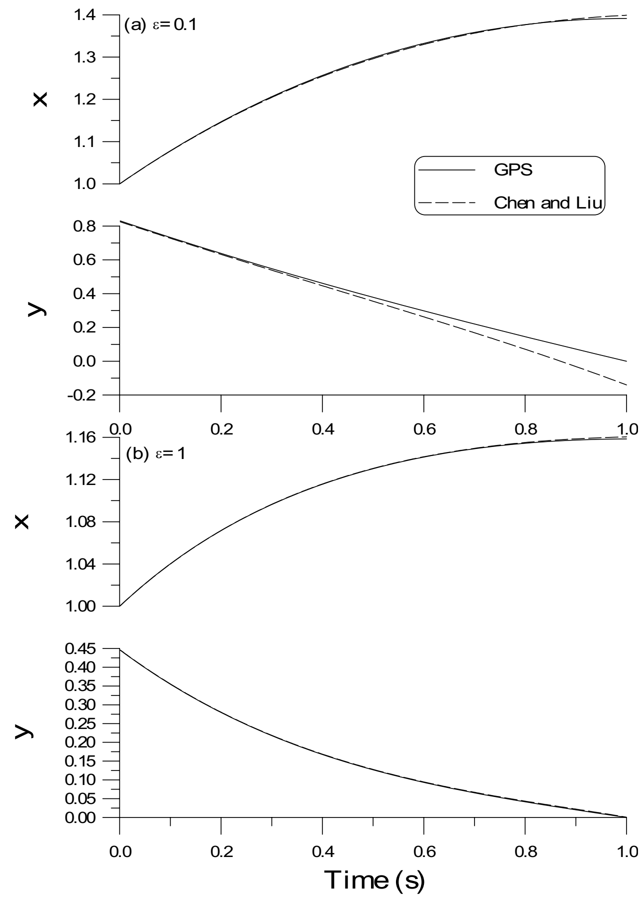

Figure 8 : Comparing the numerical solutions of GPS and that obtained by Chen and Liu (1998) for Example 8 in (a) with $\varepsilon = 0.1$, and (b) with $\varepsilon = 1$.

$$x(t) = 1.0 + 0.8267t - 0.5t^2 + 0.0551t^3 + 0.0175t^4, \tag{141}$$

$$y(t) = 0.8267 - t + 0.1653t^2 - 0.07t^3 - 0.062t^4. \tag{142}$$

It can be seen that the numerical result of $y(1) = -0.14$ obtained from Eq. (142) cannot match the condition $y(1) = 0$ and the numerical error is in the first order.

Next we consider $\varepsilon = 1$, and take $[0.6, 0.65]$ as the range for an iterative solution of r, where the root is located. By our estimation the final value B can match the exact value $y(1) = 0$ with an error 4.5912×10^{-8}. In Fig. 8 we compare the numerical results with the following solutions (Chen and Liu, 1998):

$$x(t) = 1 + 0.4469t - 0.5t^2 + 0.2979t^3 - 0.0668t^4 - 0.0566t^5 + 0.05579t^6 - 0.0118t^7 - 0.0178t^8 + 0.0186t^9 - 0.0058t^{10} - 0.0041t^{11} + 0.0058t^{12} - 0.0026t^{13} - 0.0006t^{14} + 0.0016t^{15}, \quad (143)$$

$$y(t) = 0.4469 - t + 0.8937t^2 - 0.267t^3 - 0.2832t^4 + 0.3344t^5 - 0.0827t^6 - 0.1427t^7 + 0.1676t^8 - 0.0584t^9 - 0.0451t^{10} + 0.0696t^{11} - 0.0335t^{12} - 0.0086t^{13} + 0.0237t^{14} - 0.0142t^{15}. \quad (144)$$

It can be seen that the numerical result of $y(1) = 0.0005$ obtained from Eq. (144) cannot match the condition $y(1) = 0$ very well, even Chen and Liu (1998) used a rather complex polynominal expansion method to obtain the solutions up to fifteen degree of t as given above.

7.3 Example 9

Let us consider the following BVP (Kubicek and Hlavacek, 1983):

$$\ddot{x} + \frac{1}{t}\dot{x} = -\delta e^x, \;\; \dot{x}(0) = 0, \;\; x(1) = 0. \quad (145)$$

This problem is of case (iv) in Section 7.1 and is singular at the zero point $t = 0$. We apply the estimation technique in Section 4 to this problem for searching the missing initial condition $x(0) = \alpha$. For this case (α, B) are unknowns and when applied Eqs. (70)-(76) on the solution of (α, B), Eq. (73) is replaced by

$$\alpha = \beta - \frac{\rho A}{\eta} - (1-r)\hat{f}. \quad (146)$$

The closed form solution of Eq. (145) is

$$x(t) = \ln \frac{8b}{\delta(1+bt^2)^2}, \quad (147)$$

where the integration constant b is determined by

$$\frac{8b}{\delta(1+b)^2} = 1. \quad (148)$$

It can be seen that for a given δ in the range of $0 < \delta < 2$, two distinct real roots of b in Eq. (148) exist and correspondingly, there are two solutions in Eq. (147). For $\delta = 2$, there is only one solution corresponding to $b = 1$.

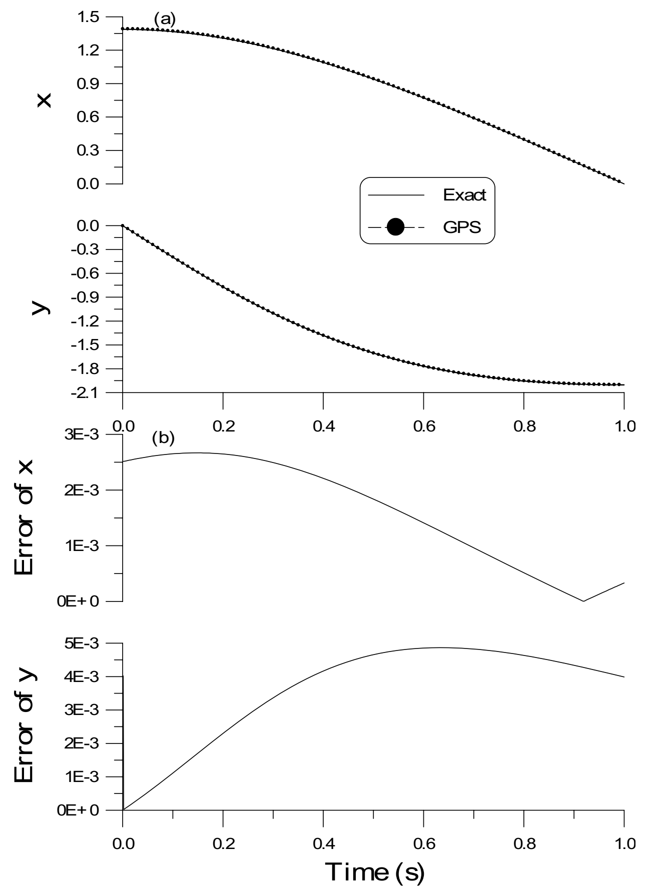

Figure 9 : Comparing the numerical solutions and exact solutions for Example 9 with $\delta = 2$ in (a), and (b) the numerical errors.

In Fig. 9 we compare our solutions with the exact solutions by taking $\delta = 2$ and $b = 1$. In our calculation we have fixed $r = 0.6013$, which leads to an error of the final value of x in the order of 3.314×10^{-4} when compared with the exact $x = 0$. It can be seen that the numerical errors of x and y are both in the order of 10^{-3}. Next, in Fig. 10 we compare our solution with the exact solution by taking $\delta = 1.5$ and b being calculated by Eq. (148):

$$b_1 = \frac{1}{2}\left[\frac{8}{\delta} - 2 + \sqrt{\left(\frac{8}{\delta} - 2\right)^2 - 4}\right],$$

$$b_2 = \frac{1}{2}\left[\frac{8}{\delta} - 2 - \sqrt{\left(\frac{8}{\delta} - 2\right)^2 - 4}\right].$$

Corresponding to b_1 and b_2 there are two exact solutions.

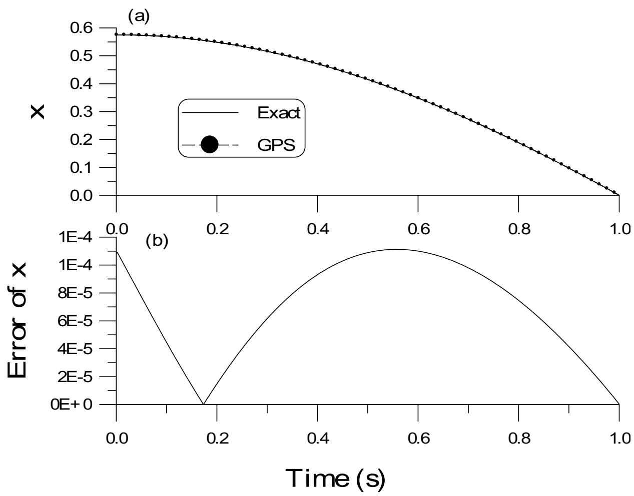

Figure 10 : Comparing the numerical solution and exact solution for Example 9 with $\delta = 1.5$ in (a), and (b) the numerical error of x.

The solution corresponding to b_2 is compared with our calculation, where we were fixed $r = 0.535822$, which leads to an error of the final value of x in the order of 3.306×10^{-7} when compared with the exact $x = 0$. It can be seen that the numerical error of x is in the order of 10^{-4}.

8 Conclusions

In this paper there were two important points deserved a further notify. The first was the construction of a one-step group $\mathbf{G}(\mathbf{u}_0, \mathbf{u}_T)$ and the full use of Eqs. (43) and (44), which are the Lie group transformation between initial conditions and final conditions in the augmented Minkowski space. Then, another one was the use of a mid-point rule to construct another Lie group element $\mathbf{G}(r)$. In order to estimate the missing initial conditions for the boundary value problems, we have employed the equation $\mathbf{G}(\mathbf{u}_0, \mathbf{u}_T) = \mathbf{G}(r)$ to derive algebraic equations. 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 convergence speed on the solution of r in a preselected range smaller than $(0,1)$ by using the half-interval method, which usually required only a small number of iterations. The numerical solution could match the specified terminal boundary conditions with a high accuracy. Some other numerical examples for the general boundary conditions of BVPs were also worked out, which show that the new methods are applicable even under a large span of the boundary coordinate. Through this study, it can be concluded that the new shooting method is accurate, effective and stable. Its numerical implementation is very simple and the computation speed is very fast.

Acknowledgement: The author would like to acknowledge the research sponsor by the National Science Council of Taiwan under the grant NSC 95-2221-E-019-002.

References

Adomian, G. (1994): Solving Frontier Problems of Physics: The Decomposition Method. Kluwer, Dordrecht.

Aris, R. (1975): The Mathematical Theory of Diffusion and Reaction in Premeable Catalyst. Clarendon Press, Oxford.

Ascher, U. M.; Mattheij, R. M. M.; Russell, R. D. (1995): Numerical Solution of Boundary Value Problems for Ordinary Differential Equations. SIAM, Philadelphia.

Burden, R. L.; Faires, J. D. (1993): Numerical Analysis. PWS, Boston.

Cash, J. R. (1986): On the numerical integration of nonlinear two-point boundary value problems using iterated deferred corrections, Part 1: a survey and comparison of some one-step formulae. *Comput. Math. Appl.*, vol. 12, pp. 1029-1048.

Cash, J. R. (1988): On the numerical integration of nonlinear two-point boundary value problems using iterated deferred corrections, Part 2: the development and analysis of highly stable deferred correction formulae. *SIAM J. Numer. Anal.*, vol. 25, pp. 862-882.

Cash, J. R.; Wright, R. W. (1998): Continuous extensions of deferred correction schemes for the numerical solution of nonlinear two-point boundary value problems. *Appl. Numer. Math.*, vol. 28, pp. 227-244.

Chen, C. L.; Liu, Y. C. (1998): Differential transformation technique for steady nonlinear heat conduction problems. *Appl. Math. Comp.*, vol. 95, pp. 155-164.

Deeba, E.; Khuri, S. A.; Xie, S. (2000): An algorithm for solving boundary value problems. *J. Comp. Phys.*, vol. 159, pp. 125-138.

Garg, V. K. (1980): Improved shooting techniques for linear boundary value problems. *Comput. Meth. Appl. Mech. Eng.*, vol. 22, pp. 87-99.

Ha, S. N. (2001): A nonlinear shooting method for two-point boundary value problems. *Comp. Math. Appl.*, vol. 42, pp. 1411-1420.

Ha, S. N.; Lee, C. R. (2002): Numerical study for two-point boundary value problems using Green's functions. *Comp. Math. Appl.*, vol. 44, pp. 1599-1608.

Kubicek, M.; Hlavacek, V. (1983): Numerical Solution of Nonlinear Boundary Value Problems with Applications. Prentice-Hall, New York.

Keller, H. B. (1992): Numerical Methods for Two-point Boundary Value Problems. Dover, New York.

Liu, C.-S. (2001): Cone of non-linear dynamical system and group preserving schemes. *Int. J. Non-Linear Mech.*, vol. 36, pp. 1047-1068.

Liu, C.-S. (2005): Nonstandard group-preserving schemes for very stiff ordinary differential equations. *CMES: Computer Modeling in Engineering & Sciences*, vol. 9, pp. 255-272.

Liu, C.-S. (2006a): Preserving constraints of differential equations by numerical methods based on integrating factors. *CMES: Computer Modeling in Engineering & Sciences*, vol. 12, pp. 83-108.

Liu, C.-S. (2006b): The Lie-group shooting method for nonlinear two-point boundary value problems exhibiting multiple solutions. *CMES: Computer Modeling in Engineering & Sciences*, vol. 13, pp. 149-163.

Liu, C.-S. (2006c): A group preserving scheme for Burgers equation with very large Reynolds number. *CMES: Computer Modeling in Engineering & Sciences*, vol. 12, pp. 197-211.

Liu, C.-S. (2006d): An efficient backward group preserving scheme for the backward in time Burgers equation. *CMES: Computer Modeling in Engineering & Sciences*, vol. 12, pp. 55-65.

Liu, C.-S. (2006e): One-step GPS for the estimation of temperature-dependent thermal conductivity. *Int. J. Heat Mass Transf.*, vol. 49, pp. 3084-3093.

Liu, C.-S. (2006f): An efficient simultaneous estimation of temperature-dependent thermophysical properties. *CMES: Computer Modeling in Engineering & Sciences*, vol. 14, pp. 77-90.

Liu, C.-S.; Chang, C.-W.; Chang, J.-R. (2006): Past cone dynamics and backward group preserving schemes for backward heat conduction problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 12, pp. 67-81.

Liu, C.-S.; Ku, Y.-L. (2005): A combination of group preserving scheme and Runge-Kutta method for the integration of Landau-Lifshitz equation. *CMES: Computer Modeling in Engineering & Sciences*, vol. 9, pp. 151-178.

Randez, L. (1993): Optimizing the numerical integration of initial value problems in shooting methods for linear boundary value problems. *SIAM J. Sci. Comput.*, vol. 14, pp. 860-871.