APPLYING THE LIE GROUP METHODS ON ENGINEERING BOUNDARY VALUE PROBLEM AND INVERSE PROBLEM

Chein-Shan Liu

Department of Mechanical and Mechatronic Engineering,

National Taiwan Ocean University

May 8, 2006

Dedicated to the Department of Mechanical Engineering,
National Taiwan University

1. The Construction of Lie Group for ODEs

2. Future cone and Past cone

3. Lie Group Shooting Method for BVPs (ODEs and PDEs)

4. One-Step Estimation Method for Inverse Problems

5. Conclusions

Hamiltonian system: Symplectic group

Rigid body dynamics: Rotation group

For General Dynamical Systems does there exist Lie Group ?

The paper by Liu

Chein-Shan Liu, 2001, Cone of nonlinear dynamical system and group preserving schemes, Int. J. Non-Linear Mechanics, vol.36, pp.1047-1068.

gives a definite answer to this problem.

1. Future cone dynamics

Let us consider the n-dimensional ODEs:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t), \quad \mathbf{x} \in \mathbb{R}^n, \quad t \in \mathbb{R}^+, \tag{1}$$

and by using

$$\|\mathbf{x}\| = \sqrt{\mathbf{x} \cdot \mathbf{x}}$$

to conduct the following calculation:

$$\frac{d}{dt}\|\mathbf{x}\| = \frac{\mathbf{x} \cdot \dot{\mathbf{x}}}{\sqrt{\mathbf{x} \cdot \mathbf{x}}} = \frac{\mathbf{x} \cdot \mathbf{f}}{\|\mathbf{x}\|} \tag{2}$$

Eqs. (1) and (2) can be written as

$$\frac{d}{dt}\begin{bmatrix} \mathbf{x} \\ \\ \|\mathbf{x}\| \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{n\times n} & \frac{\mathbf{f}(\mathbf{x},t)}{\|\mathbf{x}\|} \\ \\ \frac{\mathbf{f}^{\mathrm{T}}(\mathbf{x},t)}{\|\mathbf{x}\|} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \\ \|\mathbf{x}\| \end{bmatrix}. \tag{3}$$

The first row in Eq. (3) is the same as the original equation (1), but the inclusion of the second row in Eq. (3) gives us a

Minkowskian structure of the augmented state variable of

$$\mathbf{X} := \begin{bmatrix} \mathbf{X}^s \\ \\ \mathbf{X}^0 \end{bmatrix} = \begin{bmatrix} \mathbf{x} \\ \\ \|\mathbf{x}\| \end{bmatrix}.$$

See Fig. 1: the cone condition

$$\mathbf{X}^{\mathrm{T}}\mathbf{g}\mathbf{X} = 0, \tag{4}$$

where

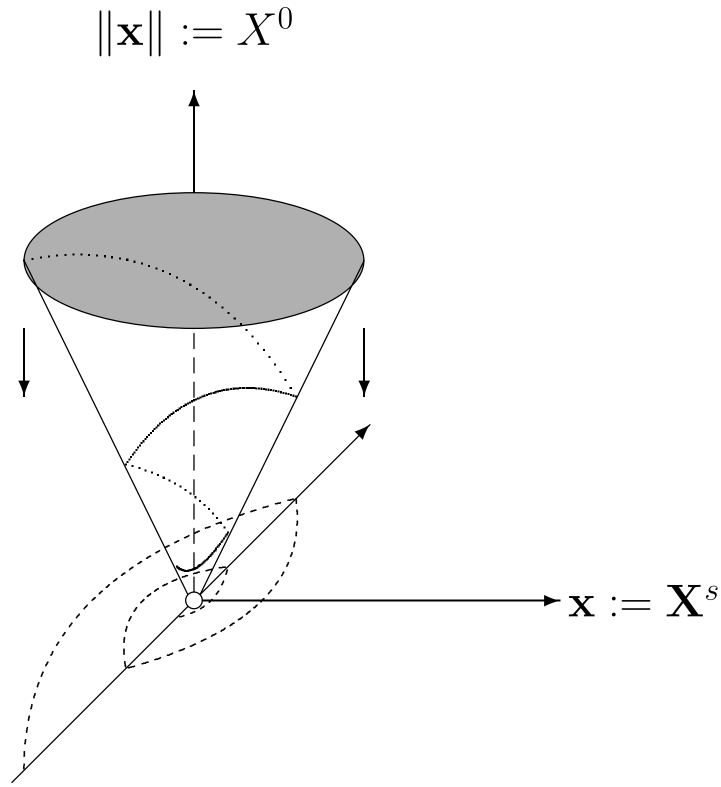
$$\mathbf{g} = \begin{bmatrix} \mathbf{I}_n & \mathbf{0}_{n\times 1} \\ \\ \mathbf{0}_{1\times n} & -1 \end{bmatrix} \tag{5}$$

is a Minkowski metric.

$$\mathbf{X}^{\mathrm{T}}\mathbf{g}\mathbf{X} = \begin{bmatrix} \mathbf{x}^{\mathrm{T}} & \|\mathbf{x}\| \end{bmatrix} \begin{bmatrix} \mathbf{I}_n & \mathbf{0}_{n\times 1} \\ \\ \mathbf{0}_{1\times n} & -1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \\ \|\mathbf{x}\| \end{bmatrix}$$

$$
= \begin{bmatrix} \mathbf{x}^{\mathrm{T}} & -\|\mathbf{x}\| \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \\ \|\mathbf{x}\| \end{bmatrix}
$$

$$
= \mathbf{x} \cdot \mathbf{x} - \|\mathbf{x}\|^2 = \|\mathbf{x}\|^2 - \|\mathbf{x}\|^2 = 0.
$$



It seems that **more** can be said.

By Eq. (3) we have an $n+1$-dimensional augmented system:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} \tag{6}$$

with a constraint (4), where

$$\mathbf{A} := \begin{bmatrix} \mathbf{0}_{n\times n} & \frac{\mathbf{f}(\mathbf{x},t)}{\|\mathbf{x}\|} \\ \frac{\mathbf{f}^{\mathrm{T}}(\mathbf{x},t)}{\|\mathbf{x}\|} & 0 \end{bmatrix}, \tag{7}$$

satisfying

$$\mathbf{A}^{\mathrm{T}}\mathbf{g} + \mathbf{g}\mathbf{A} = \mathbf{0}, \tag{8}$$

is a Lie algebra $so(n,1)$ of the proper orthochronous Lorentz group $SO_o(n,1)$.

This fact prompts us to devise the so-called group-preserving scheme, whose discretized mapping $\mathbf{G}$ exactly preserves the following properties:

$$\mathbf{G}^{\mathrm{T}}\mathbf{g}\mathbf{G} = \mathbf{g}, \tag{9}$$

$$\det \mathbf{G} = 1, \tag{10}$$

$$G_0^0 > 0, \tag{11}$$

where G_0^0 is the 00th component of $\mathbf{G}$. Such $\mathbf{G}$ is a proper orthochronous Lorentz group denoted by $SO_o(n, 1)$.

Remarkably, the original n-dimensional dynamical system (1) in $\mathbb{E}^n$ can be embedded naturally into an augmented $n+1$-dimensional dynamical system (6) in $\mathbb{M}^{n+1}$. That two systems are mathematically equivalent. Although the dimension of the new system is raised by one, it has been shown that we can develop the group preserving scheme (GPS):

$$\mathbf{X}_{\ell+1} = \mathbf{G}(\ell)\mathbf{X}_{\ell}, \tag{12}$$

where $\mathbf{X}_\ell$ denotes the numerical value of $\mathbf{X}$ at the discrete time

t_ℓ, and $\mathbf{G}(\ell) \in SO_o(n, 1)$ is the group value at time t_ℓ.

After that a new field today being called the Lie Group Integrator or Geometrical Integrator is thus brought out. (As I know that there are at least two Ph.D dissertations, one book, and over twenty papers according to this new idea.)

My contributions are given the general dynamical systems or ODEs a geometric picture of **future cone**, a **Lie algebra**, a **Lie group**, and its many **Group Preserving Schemes** (GPS).

Development of **GPS + Runge-Kutta Method**.

Stiff ODEs:

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

ODEs with constraints:

Chein-Shan Liu, 2006, Preserving constraints of differential equations by numerical methods based on integrating factors, CMES: Computer Modeling in Engineering & Sciences, vol.12, pp. 83-107.

2. Past cone dynamics

Time has two directions: **past** and **future**.

The time dynamics that goes to future is known as a **forward problem**, and that goes to past is called a **backward problem**.

Corresponding to the initial value problems (IVPs) governed by Eq. (1) with a specified initial value $\mathbf{x}(0)$ at zero time, for many systems in engineering applications, the final value problems (FVPs) may happen due to one wants to retrieve the past histories of states exhibited in the physical models. These time backward problems can be described as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t), \quad \mathbf{x} \in \mathbb{R}^n, \quad t \in \mathbb{R}^-. \tag{13}$$

With a specified final value $\mathbf{x}(0)$ at $t = 0$, we intend to recover the past values of $\mathbf{x}$ in the past time of $t < 0$.

For the past dynamics does there have the similar structures as that for the forward problem which governed by ODEs.

After five years this problem is solved by a definite answer:

Chein-Shan Liu, 2006, An efficient backward group preserving scheme for the backward in time Burgers equation, CMES: Computer Modeling in Engineering & Sciences, vol.12, pp. 55-65.

Chein-Shan Liu, Chih-Wen Chang and Jiang-Ren Chang, 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.

Here, I summarize the basic ingredients:

$$\mathbf{X} := \begin{bmatrix} \mathbf{x} \\ -\|\mathbf{x}\| \end{bmatrix},$$

$$\dot{\mathbf{X}} = \mathbf{B}\mathbf{X} \tag{14}$$

with a constraint (4), where

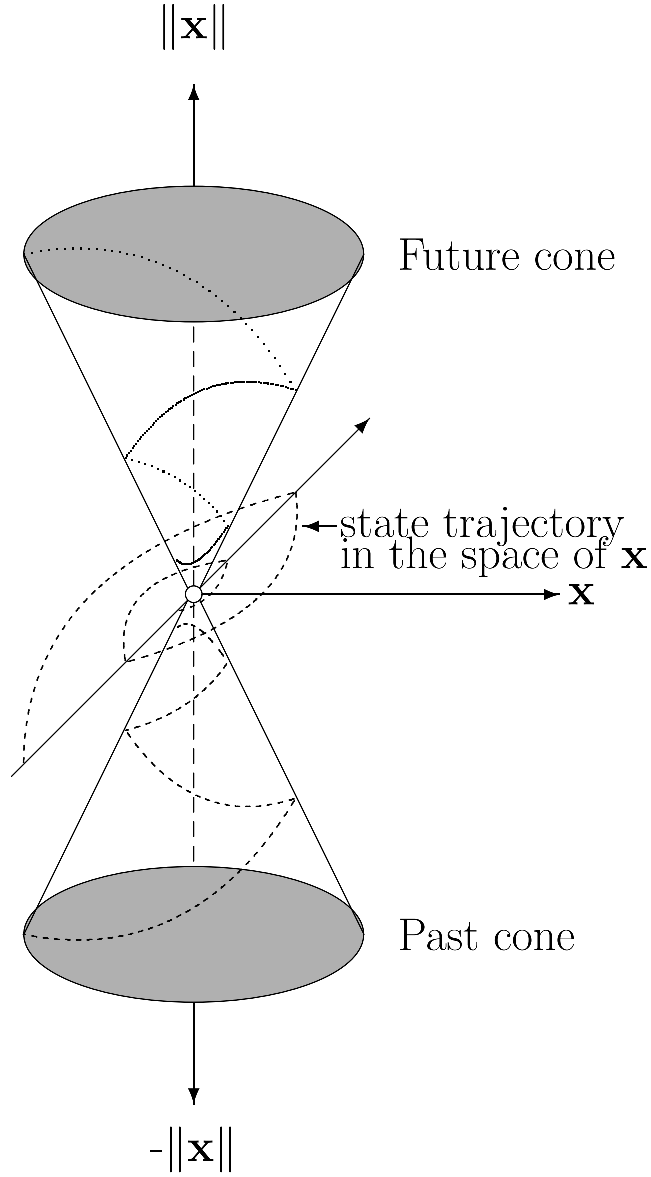
$$\mathbf{B} := \begin{bmatrix} \mathbf{0}_{n\times n} & -\frac{\mathbf{f}(\mathbf{x},t)}{\|\mathbf{x}\|} \\ -\frac{\mathbf{f}^{\mathrm{T}}(\mathbf{x},t)}{\|\mathbf{x}\|} & 0 \end{bmatrix}, \tag{15}$$

satisfying

$$\mathbf{B}^{\mathrm{T}}\mathbf{g} + \mathbf{g}\mathbf{B} = \mathbf{0}, \tag{16}$$

is a Lie algebra $so(n, 1)$ of the proper orthochronous Lorentz group $SO_o(n, 1)$.

See Fig. 2: the past cone dynamics.


$\|\mathbf{x}\|$
Future cone
state trajectory
in the space of $\mathbf{x}$
$\mathbf{x}$
Past cone
-$\|\mathbf{x}\|$

What are **Inverse Problems** ?

The inverse problem arises when one or more conditions in the direct problem are absent. It can be classified into five types depending on what part in the direct problem is absent.

(1) The backward problem: initial condition is unknown.

(2) The sideway problem: boundary condition is unknown.

(3) The identification of source: the external source exerted on the system is unknown.

(4) The parameter identification problem: the system parameter or function is unknown.

(5) The geometric shape identification problem: the domain of system is unknown.

For partial differential equations (PDEs) there are further classified into three categories: **Elliptic type**, **Parabolic type** and **Hyperbolic type**.

For the **direct problem** the numericalist pursuits the **accuracy** because the direct problems are most **well-defined**.

Conversely, for the **inverse problem** the numericalist pursuits the **stability** because the inverse problems are all **ill-posed**.

How to apply the GPS or BGPS on the inverse problems.

The **semi-discretization** is simple in concept that for a given system of partial differential equations discretize all but one of the independent variables. The semi-discrete procedure yields a coupled system of ordinary differential equations which are then numerically integrated.

For example, for the one-dimensional backward in time Burgers equation:

$$u_t + uu_x = \frac{1}{R}u_{xx}, \quad a < x < b, \quad 0 < t < T, \tag{17}$$

$$u(a,t) = u_a(t), \quad u(b,t) = u_b(t), \quad 0 \le t \le T, \tag{18}$$

$$u(x,T) = f(x), \quad a \le x \le b, \tag{19}$$

we adopt the numerical method of line to discretize the spatial coordinate x by

$$\left.\frac{\partial u(x,t)}{\partial x}\right|_{x=a+i\Delta x} = \frac{u_{i+1}(t) - u_{i-1}(t)}{2\Delta x}, \tag{20}$$

$$\left.\frac{\partial^2 u(x,t)}{\partial x^2}\right|_{x=a+i\Delta x} = \frac{u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)}{(\Delta x)^2}, \tag{21}$$

where Δx is a uniform discretization spacing length, and $u_i(t) = u(a + i\Delta x, t)$, such that Eq. (17) can be approximated by

$$\frac{\partial u_i(t)}{\partial t} = \frac{1}{R(\Delta x)^2}[u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)] - u_i(t)\frac{u_{i+1}(t) - u_{i-1}(t)}{2\Delta x}. \tag{22}$$

The next step is to advance the solution from the final condition to the desired time $t = 0$. Really, in Eq. (22) there are totally n coupled nonlinear differential equations for the n variables $u_i(t), i = 1, 2, \ldots, n$, which can be numerically integrated to obtain the numerical solutions.

For the inverse problems there are many challenges. Since 2005 I have made some contributions in this field including five published papers and over ten submitted papers. Some results about the above mentioned papers are shown here.

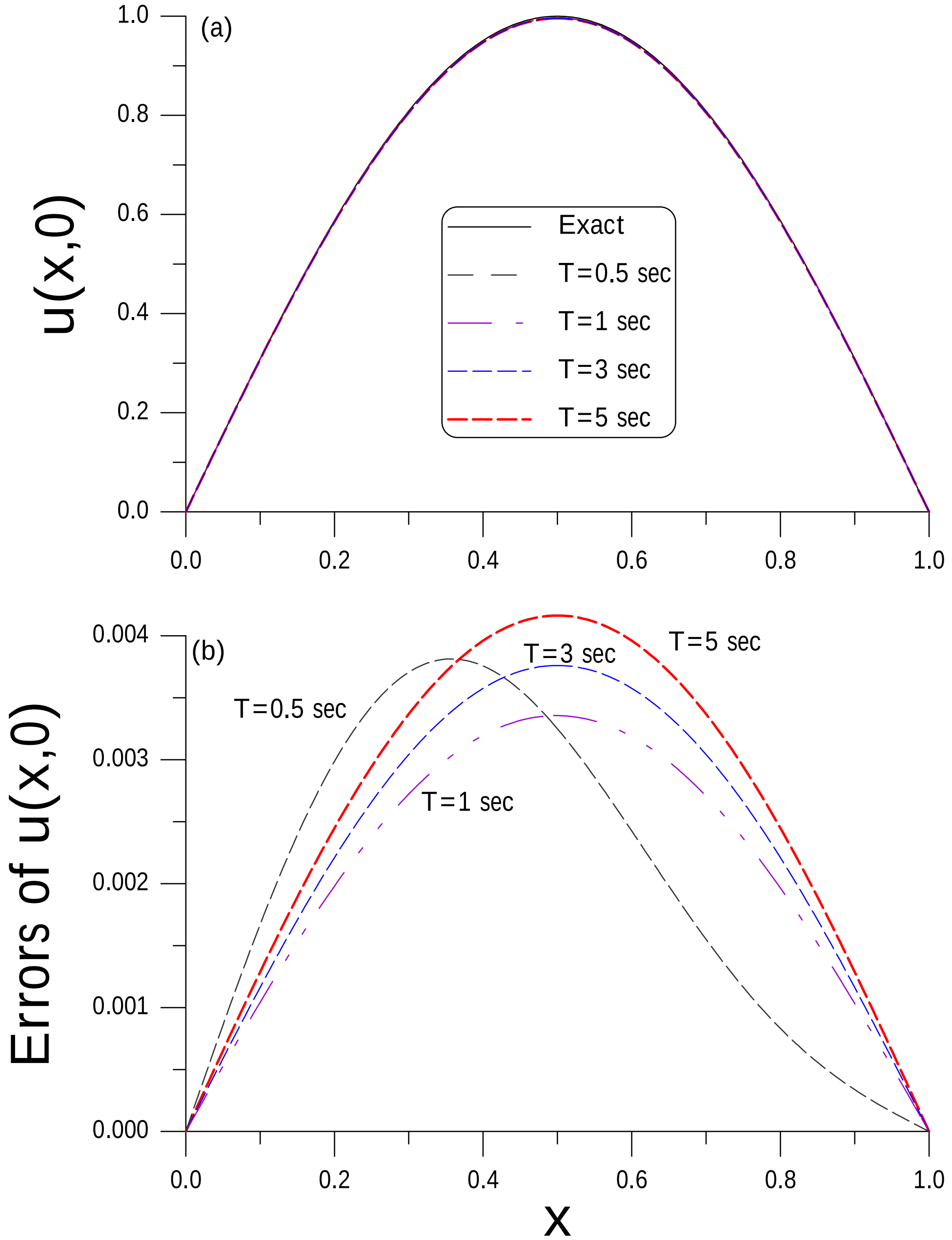


Figure 2: The comparison of exact solutions and numerical solutions for Example 1 of backward Burgers' equation with different final times: T=0.5,1,3,5 sec.

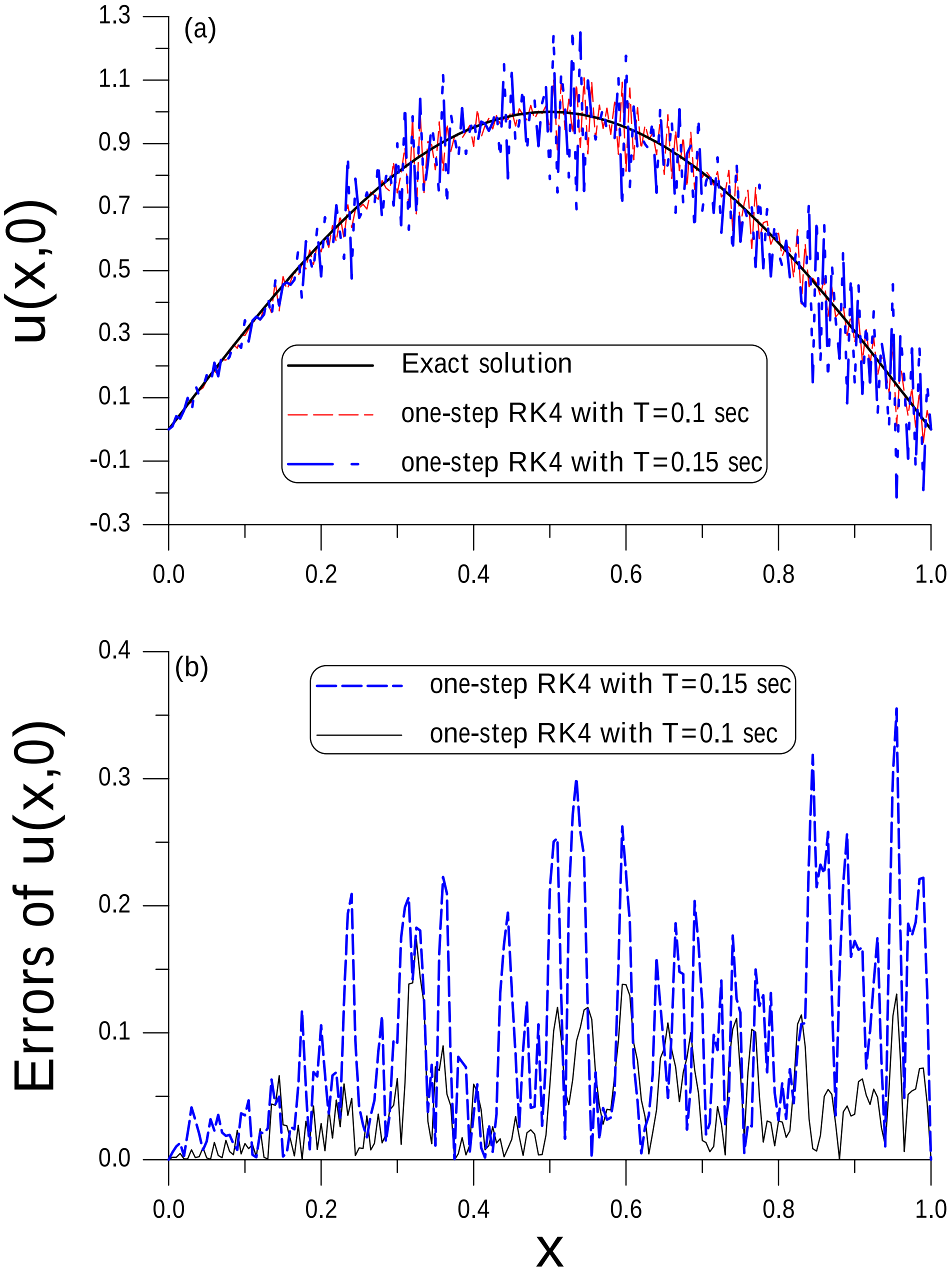


Figure 3: The comparison of exact solutions and numerical solutions for Example 1 by one-step RK4 with different final times: T=0.1,0.15 sec.

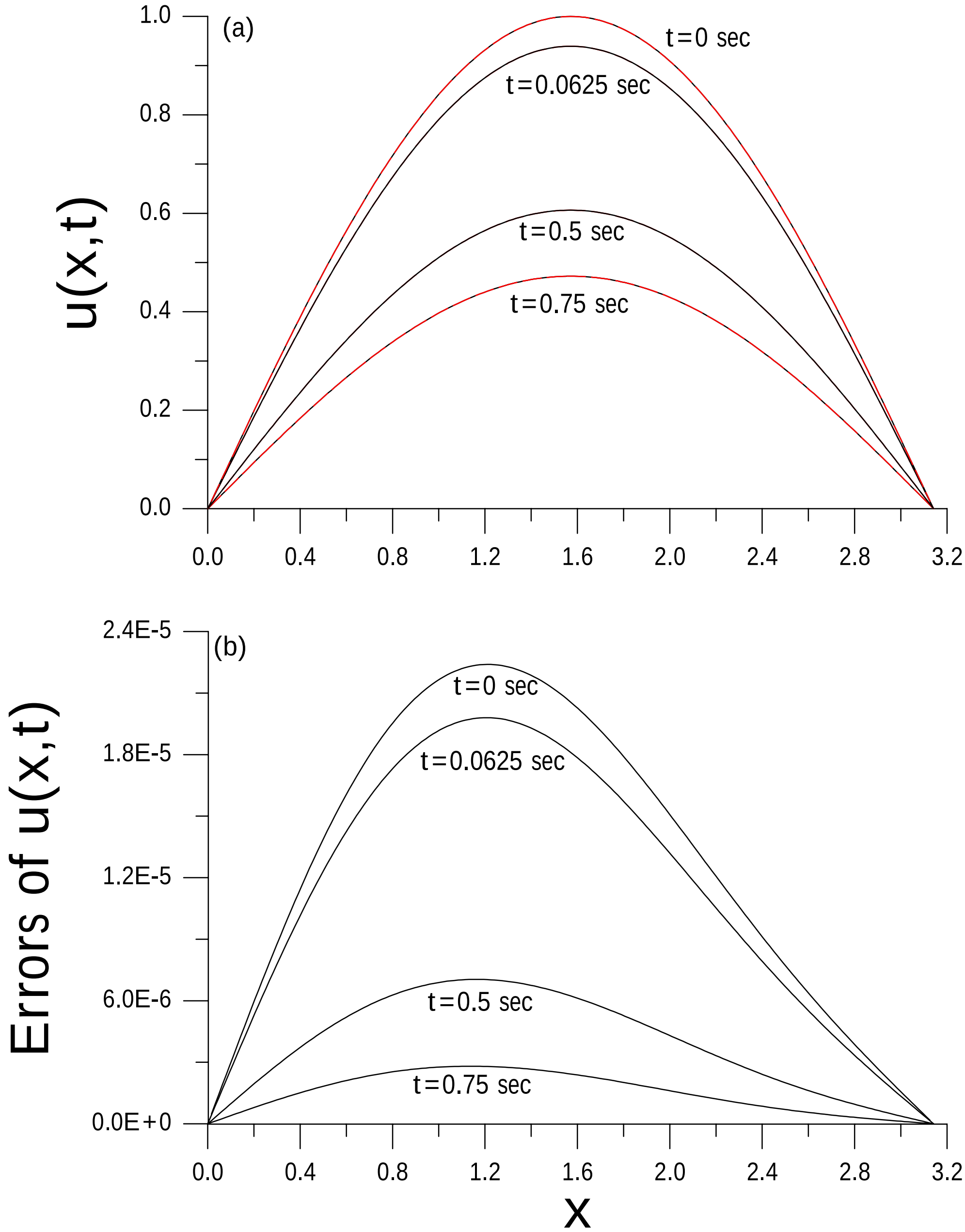


Figure 6: The comparison of exact solutions and numerical solutions for Example 2 with data at different times been retrieved: t=0.75,0.5,0.0625,0 sec.

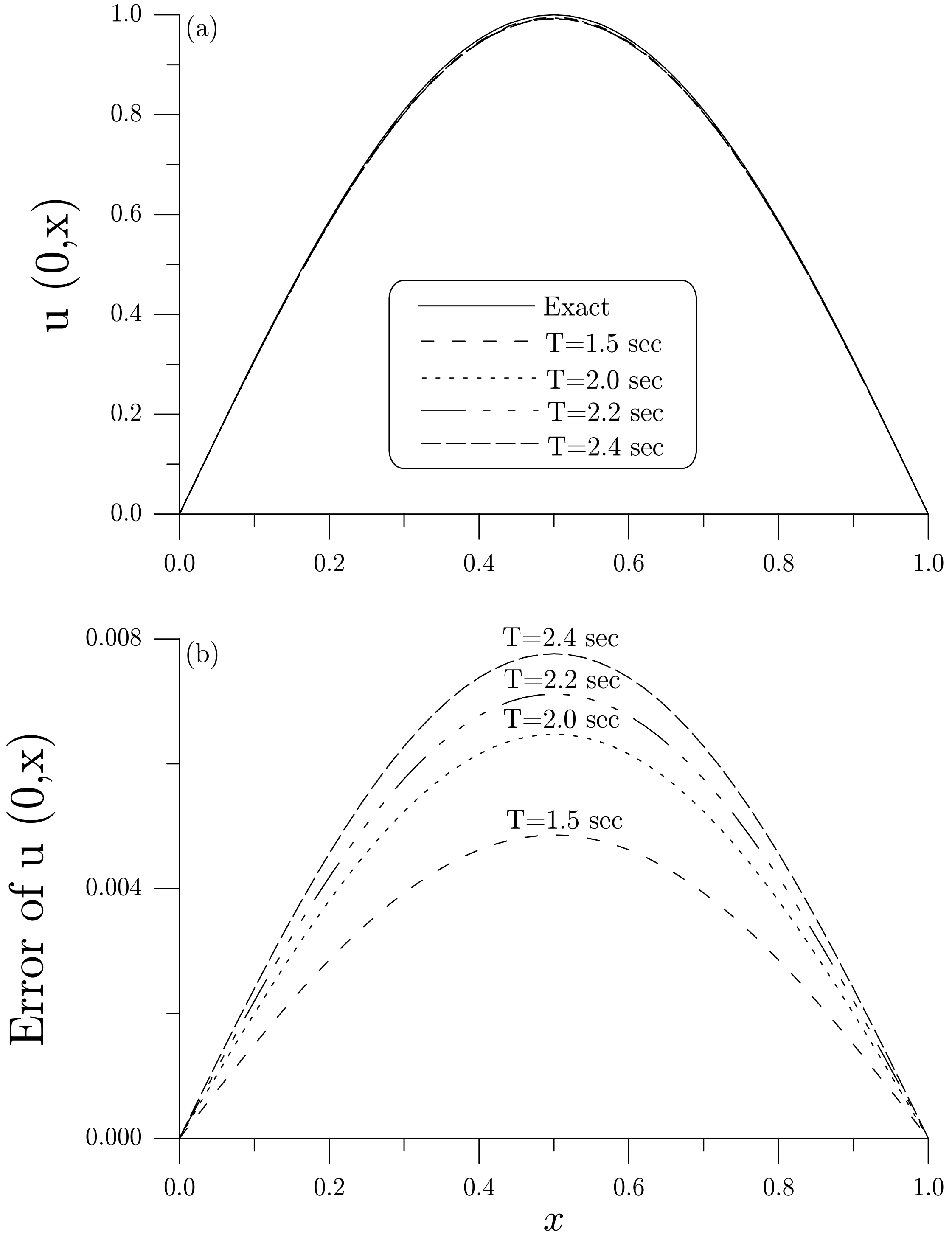


Figure 7: The comparison of exact solutions and numerical solutions for Example 3 were made in (a) with different final times of $T = 1.5,\ 2,\ 2.2,\ 2.4$ sec, and (b) the errors of numerical solutions.

Next, I come to the topic of boundary value problems. The problem is how to use the initial value integrator GPS to the BVPs. The answer is the Lie group shooting method:

Chein-Shan Liu, 2006, The Lie-group shooting method for nonlinear two-point boundary value problems exhibiting multiple solutions, CMES: Computer Modeling in Engineering & Sciences, in press.

3. Lie Group Shooting Method for BVPs (ODEs and PDEs)

The backward heat conduction problem (BHCP) we consider is

$$u_t = u_{xx}, \quad a < x < b, \quad 0 < t < T, \tag{23}$$

$$u(a,t) = u_a(t), \quad u(b,t) = u_b(t), \quad 0 \le t \le T \tag{24}$$

$$u(x,T) = f(x), \quad a \le x \le b, \tag{25}$$

where u is a scalar temperature field of heat distribution.

Here we are going to calculate the BHCP by a semi-discretization method, which replaces Eq. (23) by a set of ODEs:

$$\dot{u}_i(t) = \frac{1}{(\Delta x)^2}[u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)] \tag{26}$$

Regularization:

$$\alpha u(x, 0) + u(x, T) = f(x). \tag{27}$$

3.1 One-step GPS

Let us write

$$\mathbf{u} := \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}, \quad \mathbf{f} := \frac{1}{(\Delta x)^2} \begin{bmatrix} u_2 - 2u_1 + u_0 \\ u_3 - 2u_2 + u_1 \\ \vdots \\ u_{n+1} - 2u_n + u_{n-1} \end{bmatrix}. \tag{28}$$

Then, Eq. (26) for $i = 1, \ldots, n$ can be expressed as a vector form:

$$\dot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}), \tag{29}$$

in which Eq. (27) as being a constraint is written to be

$$\alpha \mathbf{u}(0) + \mathbf{u}(T) = \mathbf{h}, \tag{30}$$

where

$$\mathbf{h} := \begin{bmatrix} h(x_1) \\ h(x_2) \\ \vdots \\ h(x_n) \end{bmatrix}. \tag{31}$$

3.2 Generalized mid-point rule

According to Liu (2001) we have a group-preserving scheme (GPS) to guarantee that each $\mathbf{X}_k$ is located on the cone:

$$\mathbf{X}_{k+1} = \mathbf{G}(k)\mathbf{X}_k, \tag{32}$$

where $\mathbf{X}_k$ denotes the numerical value of $\mathbf{X}$ at the discrete t_k.

Applying scheme (32) to

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

with a specified initial condition $\mathbf{X}^0 = \mathbf{X}(0)$ we can compute the solution $\mathbf{X}(t)$ by GPS. Assuming that the stepsize used in GPS is $\Delta t = T/K$, and starting from an initial augmented condition $\mathbf{X}^0 = ((\mathbf{u}^0)^{\mathrm{T}}, \|\mathbf{u}^0\|)^{\mathrm{T}}$ we will calculate the value $\mathbf{X}^f = ((\mathbf{u}(T))^{\mathrm{T}}, \|\mathbf{u}(T)\|)^{\mathrm{T}}$ at $t = T$.

By applying Eq. (32) step-by-step we can obtain

$$\mathbf{X}^f = \mathbf{G}_K(h)\cdots\mathbf{G}_1(h)\mathbf{X}^0, \tag{33}$$

However, 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 Lie group, $\mathbf{G}_K(h)\cdots\mathbf{G}_1(h)$ is also a Lie group denoted by

$\mathbf{G}$. Hence, we have

$$\mathbf{X}^f = \mathbf{G}\mathbf{X}^0. \tag{34}$$

This is a one-step transformation from $\mathbf{X}^0$ to $\mathbf{X}^f$.

We can calculate $\mathbf{G}$ by a generalized mid-point rule, which is obtained from an exponential mapping of $\mathbf{A}$ by taking the values of the argument variables of $\mathbf{A}$ at a generalized mid-point:

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_n + \frac{(a-1)}{\|\hat{\mathbf{f}}\|^2}\hat{\mathbf{f}}\hat{\mathbf{f}}^{\mathrm{T}} & \frac{b\hat{\mathbf{f}}}{\|\hat{\mathbf{f}}\|} \\ \frac{b\hat{\mathbf{f}}^{\mathrm{T}}}{\|\hat{\mathbf{f}}\|} & a \end{bmatrix}, \tag{35}$$

where

$$\hat{\mathbf{u}} = r\mathbf{u}^0 + (1-r)\mathbf{u}^f, \tag{36}$$

$$\hat{\mathbf{f}} = \mathbf{f}(\hat{t}, \hat{\mathbf{u}}), \tag{37}$$

$$a = \cosh\left(T\frac{\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{u}}\|}\right), \tag{38}$$

$$b = \sinh\left(T\frac{\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{u}}\|}\right). \tag{39}$$

Here, we use the initial $\mathbf{u}^0 = (u_1(0), \ldots, u_n(0))$ and the final $\mathbf{u}^f = (u_1(T), \ldots, u_n(T))$ through a suitable weighting factor r to calculate $\mathbf{G}$, where $r \in (0,1)$ is a parameter and $\hat{t} = (1-r)T$.

Recall that the **mean value theorem** is

$$\int_a^b f(x)dx = (b-a)f(c), \quad c \in [a,b].$$

The approach of Eq. (35) can be realized alternatively by using

$$\dot{\mathbf{G}} = \mathbf{A}(t,\mathbf{u})\mathbf{G}. \tag{40}$$

Integrating the above equation and using the mean-value theorem we obtain

$$\mathbf{G} = \exp\left[\int_0^T \mathbf{A}(t,\mathbf{u})dt\right] = \exp[T\mathbf{A}(\hat{t},\hat{\mathbf{u}})]. \tag{41}$$

The above methods applied a generalized mid-point rule or the mean value theorem on the calculations of $\mathbf{G}$, and the resultant is a single-parameter Lie group element denoted by $\mathbf{G}(r)$.

3.3 A Lie group mapping between two points on the cone

Let us define a new vector

$$\mathbf{F} := \frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{u}}\|}, \tag{42}$$

such that Eqs. (35), (38) and (39) can also be expressed as

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_n + \frac{a-1}{\|\mathbf{F}\|^2}\mathbf{F}\mathbf{F}^{\mathrm{T}} & \frac{b\mathbf{F}}{\|\mathbf{F}\|} \\ \frac{b\mathbf{F}^{\mathrm{T}}}{\|\mathbf{F}\|} & a \end{bmatrix}, \tag{43}$$

$$a = \cosh[T\|\mathbf{F}\|], \tag{44}$$

$$b = \sinh[T\|\mathbf{F}\|]. \tag{45}$$

From Eqs. (34) and (43) it follows that

$$\mathbf{u}^f = \mathbf{u}^0 + \eta \mathbf{F}, \tag{46}$$

$$\|\mathbf{u}^f\| = a\|\mathbf{u}^0\| + b\frac{\mathbf{F}\cdot\mathbf{u}^0}{\|\mathbf{F}\|}, \tag{47}$$

where (through some derivations)

$$\eta = \frac{T\|\mathbf{u}^f - \mathbf{u}^0\|}{\ln Z}, \tag{48}$$

$$Z = \frac{\frac{\|\mathbf{u}^f\|}{\|\mathbf{u}^0\|} + \sqrt{\left(\frac{\|\mathbf{u}^f\|}{\|\mathbf{u}^0\|}\right)^2 - 1 + \cos^2\theta}}{1+\cos\theta}, \tag{49}$$

$$\cos\theta := \frac{[\mathbf{u}^f - \mathbf{u}^0]\cdot\mathbf{u}^0}{\|\mathbf{u}^f - \mathbf{u}^0\|\|\mathbf{u}^0\|}. \tag{50}$$

Therefore, between any two points $(\mathbf{u}^0, \|\mathbf{u}^0\|)$ and $(\mathbf{u}^f, \|\mathbf{u}^f\|)$ on the cone, there exists a Lie group element $\mathbf{G} \in SO_o(n,1)$ mapping $(\mathbf{u}^0, \|\mathbf{u}^0\|)$ onto $(\mathbf{u}^f, \|\mathbf{u}^f\|)$, which is given by

$$\begin{bmatrix} \mathbf{u}^f \\ \\ \|\mathbf{u}^f\| \end{bmatrix} = \mathbf{G} \begin{bmatrix} \mathbf{u}^0 \\ \\ \|\mathbf{u}^0\| \end{bmatrix}, \tag{51}$$

where $\mathbf{G}$ is uniquely determined by $\mathbf{u}^0$ and $\mathbf{u}^f$.

3.4 Shooting method

From Eqs. (42) and (46) it follows that

$$\mathbf{u}^f = \mathbf{u}^0 + \eta \frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{u}}\|}. \tag{52}$$

By Eq. (30) we obtain

$$\alpha \mathbf{u}^0 + \mathbf{u}^f = \mathbf{h}. \tag{53}$$

Eqs. (52) and (53) can be utilized to solve $\mathbf{u}^0$:

$$\mathbf{u}^0 = \frac{1}{1+\alpha}\left[\mathbf{h} - \eta \frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{u}}\|}\right]. \tag{54}$$

The above derivation of the governing equations (52)-(54) is stemed from by letting the two Lie group elements $\mathbf{G}(\mathbf{u}^0, \mathbf{u}^f)$ and $\mathbf{G}(r)$ equal.

For a specified r, Eq. (54) can be used to generate the new $\mathbf{u}^0$, until $\mathbf{u}^0$ converges according to a given stopping criterion:

$$\|\mathbf{u}_{i+1}^0 - \mathbf{u}_i^0\| \leq \varepsilon. \tag{55}$$

If $\mathbf{u}^0$ is available, we can return to Eq. (29) and integrate it to obtain $\mathbf{u}(T)$. The above process can be done for all r in the interval of $r \in (0,1)$. Among these solutions we may pick up the r, which leads to the smallest error of Eq. (30). That is,

$$\min_{r\in(0,1)} \|\alpha\mathbf{u}^0 + \mathbf{u}^f - \mathbf{h}\|. \tag{56}$$

3.5 Numerical Examples

Example 1: Post buckling of elastica

Let us consider the Euler problem of a slender rod with simple support subjecting to a compressive load:

$$\mu + p^2 u = 0, \quad u(0) = u(\pi) = 0, \tag{57}$$

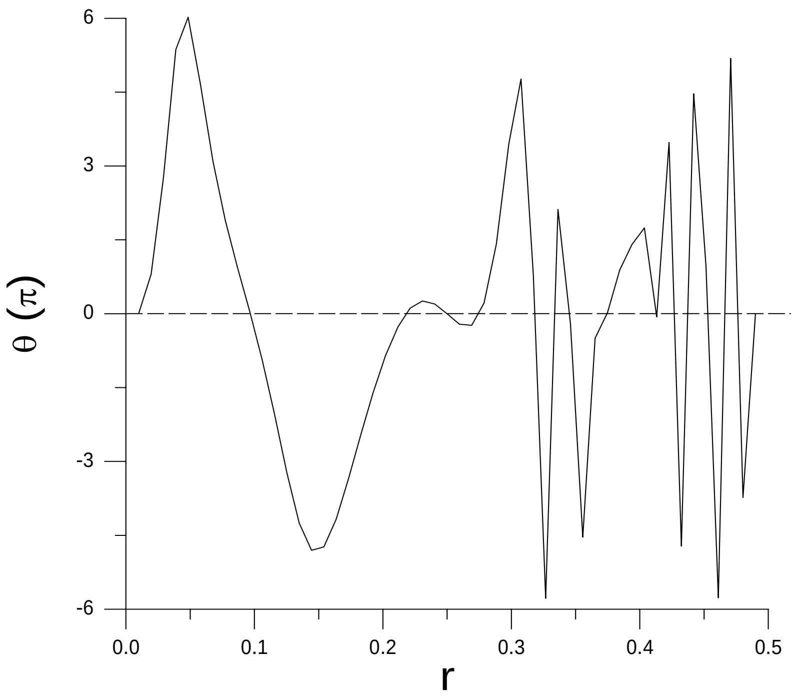
where μ is the curvature and u is the lateral displacement. By using the definition of $\mu = \theta'(s)$, where θ is the tangential angle of the deflection curve with the vertical axis and s is the arc length, and taking Eq. (57) differential with respect to s we obtain a nonlinear Neumann boundary value problem:

$$\theta''(s) + p^2 \sin\theta(s) = 0, \quad \theta'(0) = 0, \quad \theta'(\pi) = 0. \tag{58}$$

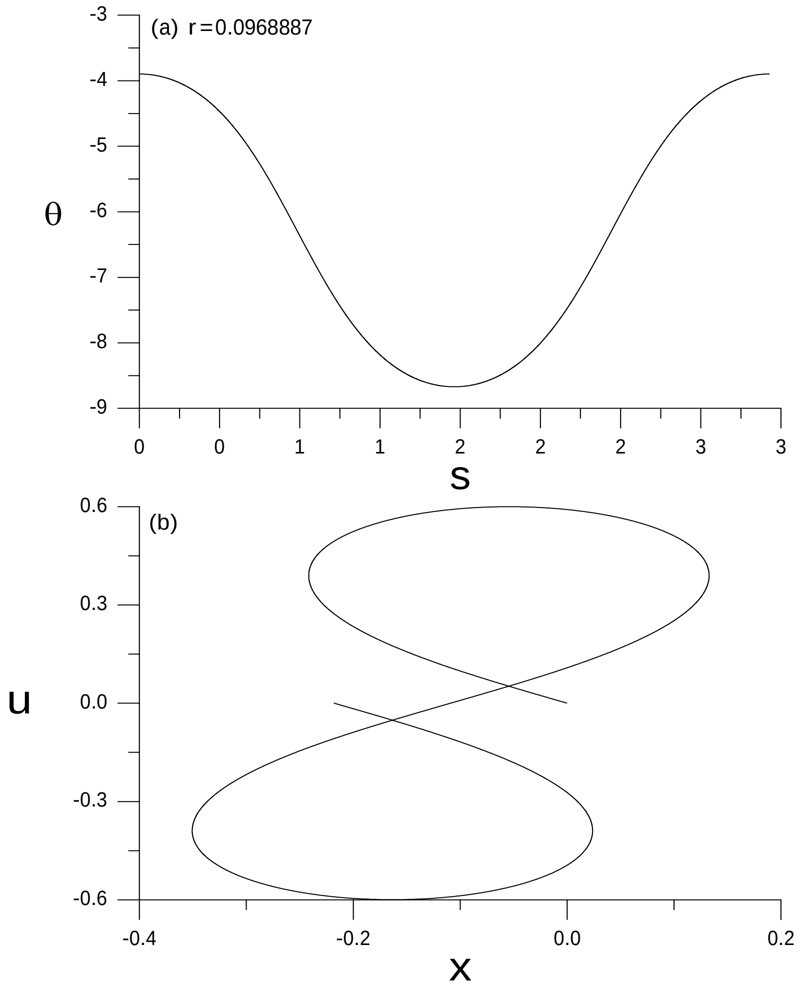
Our aim is to find the missing initial condition

$$\theta(0) = C$$

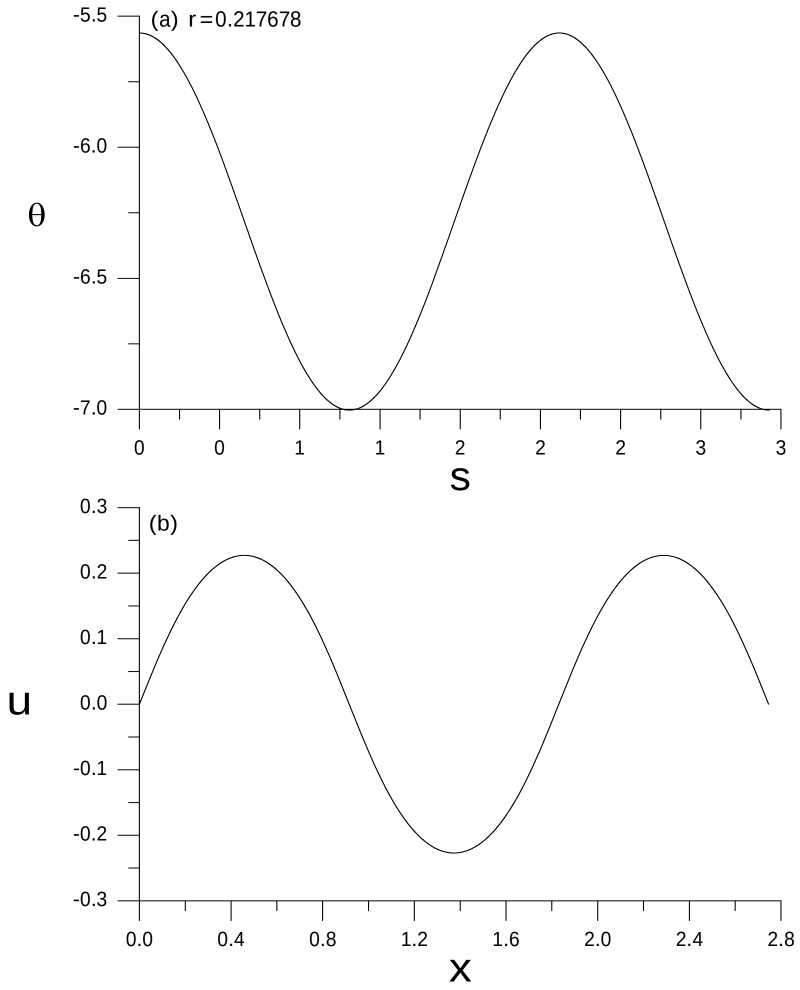
without any iteration.



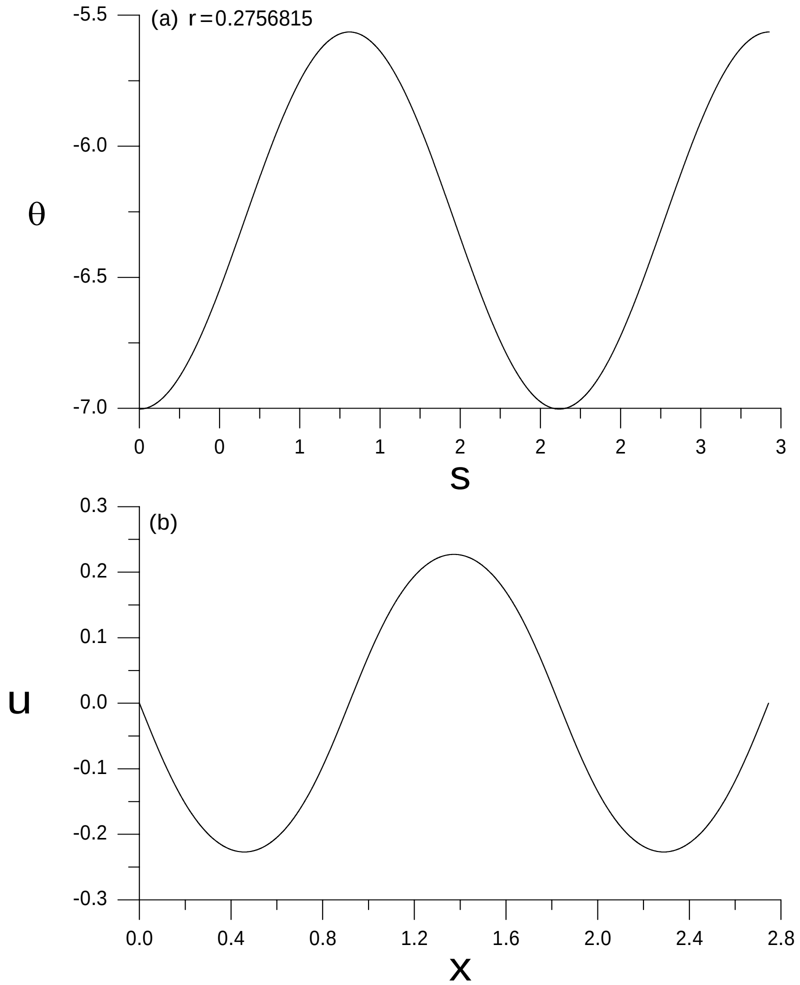
6
3
0
-3
-6
θ (π)
0.0
0.1
0.2
0.3
0.4
0.5
r

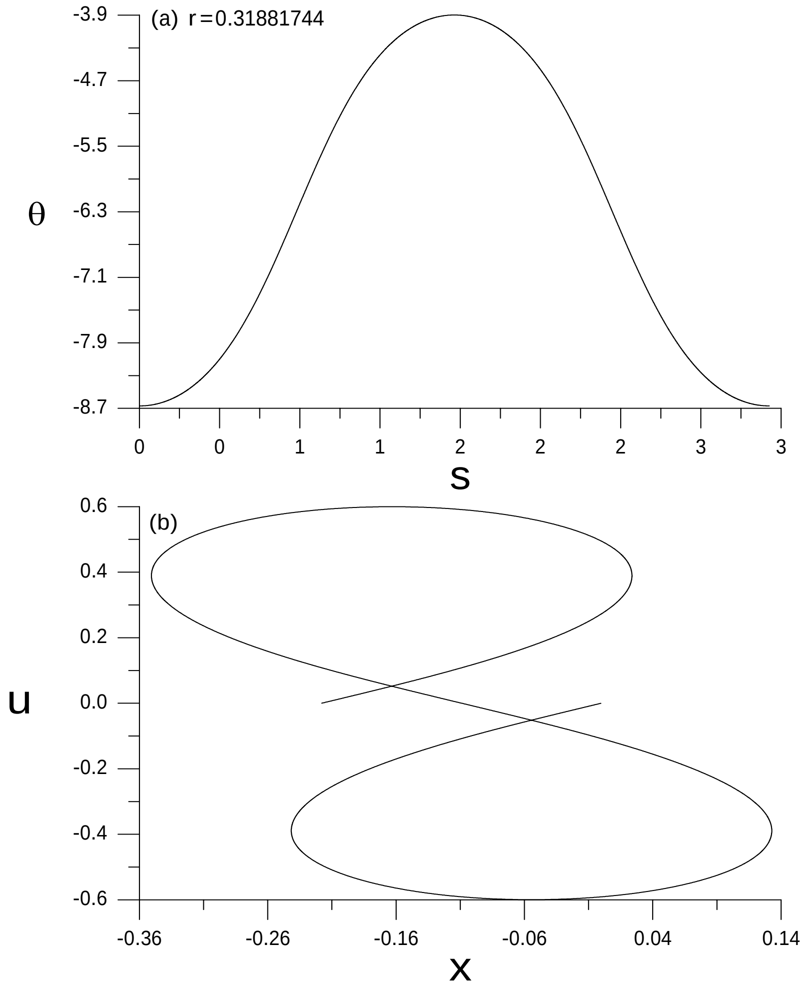


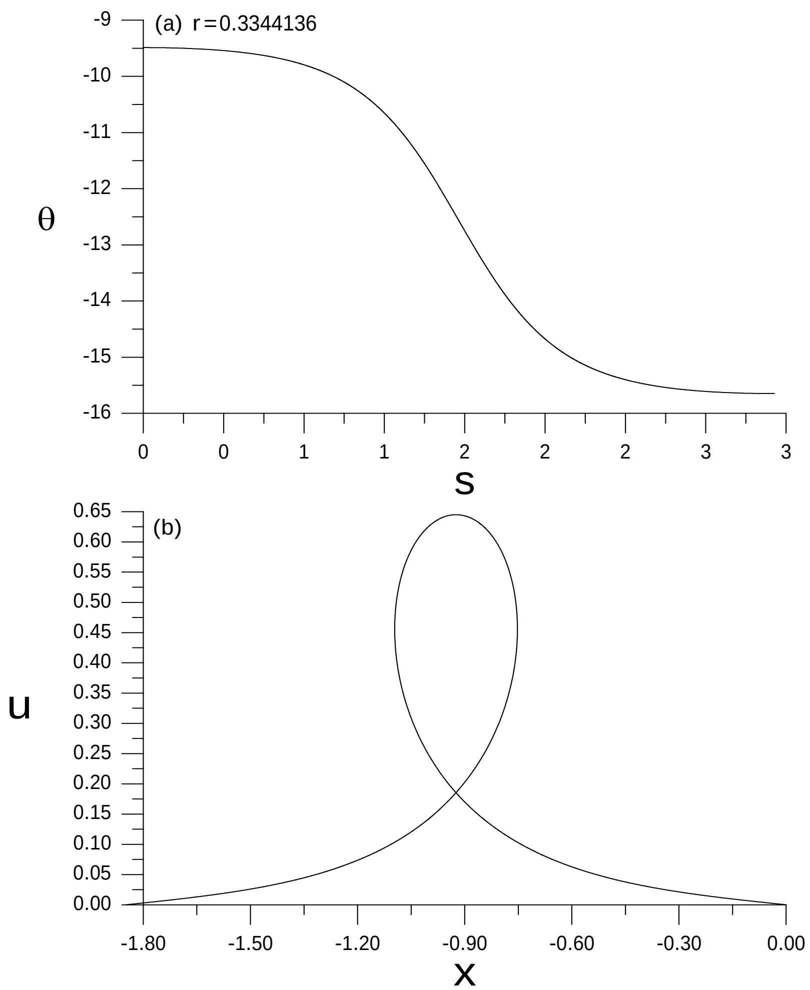
(a) r = 0.0968887
θ
s
-3
-4
-5
-6
-7
-8
-9
0
0
1
1
2
2
2
3
3
(b)
u
x
0.6
0.3
0.0
-0.3
-0.6
-0.4
-0.2
0.0
0.2



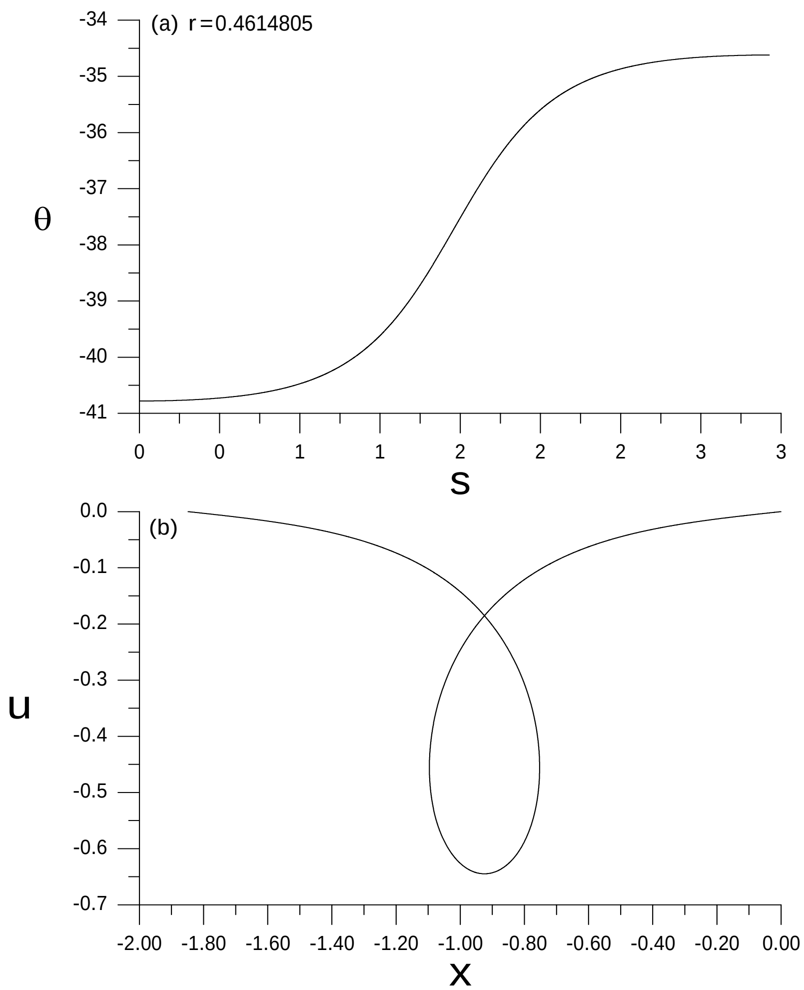
(a) r = 0.217678
θ
-5.5
-6.0
-6.5
-7.0
0
0
1
1
2
2
2
3
3
s
(b)
u
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
0.0
0.4
0.8
1.2
1.6
2.0
2.4
2.8
x


(a) r=0.2756815
-5.5
-6.0
-6.5
-7.0
θ
0
0
1
1
2
2
2
3
3
s
(b)
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
u
0.0
0.4
0.8
1.2
1.6
2.0
2.4
2.8
x


(a) r=0.31881744
θ
-3.9
-4.7
-5.5
-6.3
-7.1
-7.9
-8.7
0
0
1
1
2
2
2
3
3
s
(b)
u
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
-0.36
-0.26
-0.16
-0.06
0.04
0.14
x



(a) r=0.3344136
θ
-9
-10
-11
-12
-13
-14
-15
-16
0
0
1
1
2
2
2
3
3
s
(b)
u
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
-1.80
-1.50
-1.20
-0.90
-0.60
-0.30
0.00
x



(a) r = 0.4614805
θ
-34
-35
-36
-37
-38
-39
-40
-41
0
0
1
1
2
2
2
3
3
s
(b)
u
0.0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-0.7
-2.00
-1.80
-1.60
-1.40
-1.20
-1.00
-0.80
-0.60
-0.40
-0.20
0.00
x

Example 2: Backward heat conduction problem

Let us first consider a one-dimensional benchmark BHCP:

$$u_t = u_{xx}, \quad 0 < x < 1, \quad 0 < t < T, \tag{59}$$

with the boundary conditions

$$u(0, t) = u(1, t) = 0, \tag{60}$$

and the final time condition

$$u(x, T) = \sin(\pi x) \exp(-\pi^2 T). \tag{61}$$

The data to be retrieved is given by

$$u(x, t) = \sin(\pi x) \exp(-\pi^2 t), \quad T > t \geq 0. \tag{62}$$

The mission here is to find the unknown initial condition:

$$u(x, 0) = ?$$

We apply the Lie-group shooting method (LGSM) for this backward problem of n differential equations with the final data given by Eq. (61).

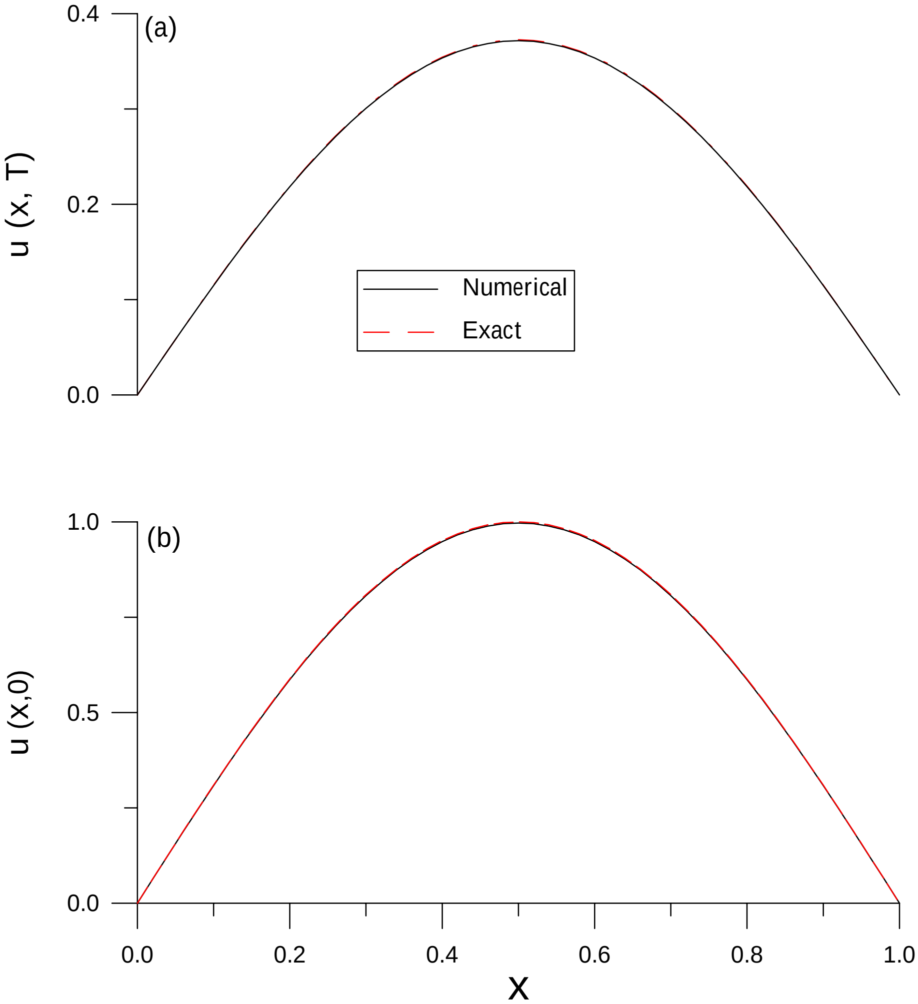
We first test the new method for a small T with $T = 0.1$ sec. The computation is under $n = 50$, i.e. $\Delta x = 1/50$, and $h = 0.0001$ sec. In order to solve Eq. (30) by the LGSM we start from an initial guess $\mathbf{u}^0 = 0.1 \sin \pi x_i$, and pick up the minimum of Eq. (56) in the range $r \in (0, 10^{-8})$. For each r, about through 14 iterations we can satisfy the stopping criterion in Eq. (55) with $\varepsilon = 10^{-5}$.

We compare the computed $u(x, T)$ and $u(x, 0)$ with the exact ones in Fig. 1 by fixing $\alpha = 0.001$. From Fig. 1 it can be seen that the computed data at the grid points are almost located on the sine curves obtained from Eq. (62) with $t = 0.1$ and $t = 0$.

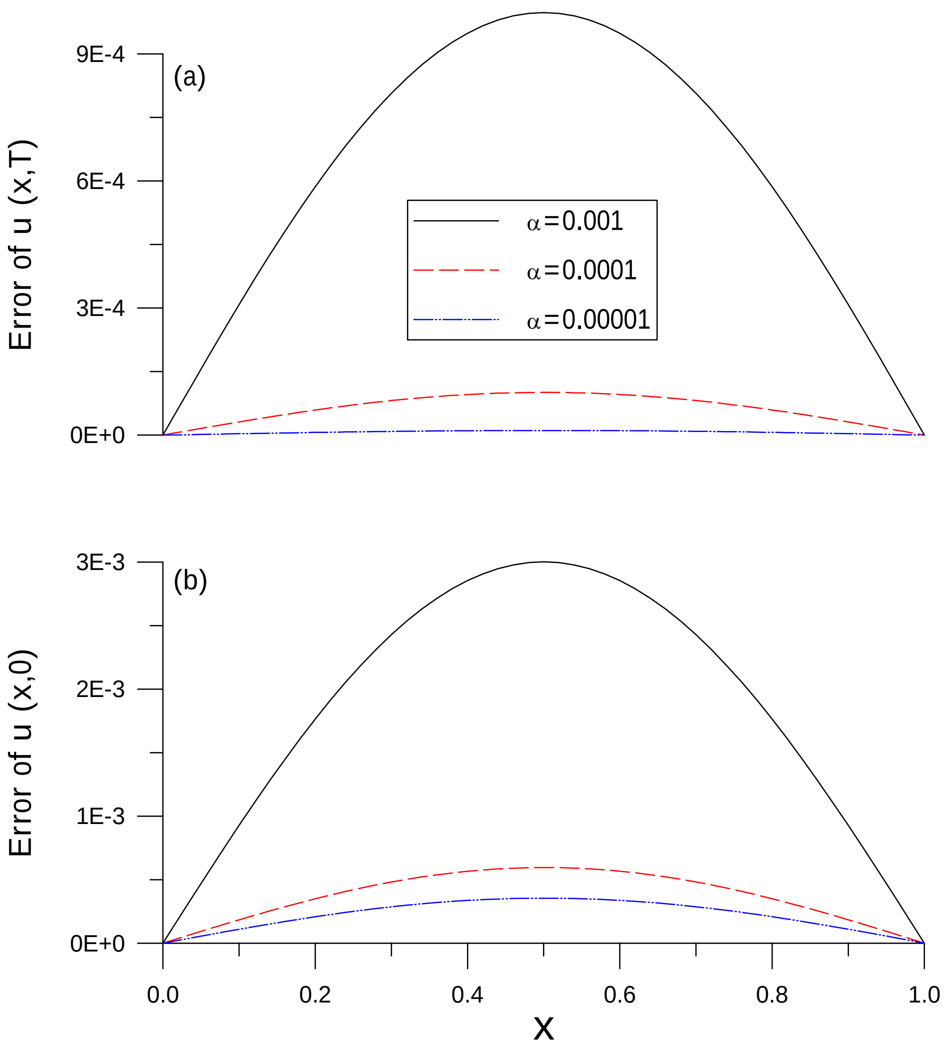
It is hardly to see the difference of these two numerical solutions with the exact solutions. Therefore, we plot the numerical errors defined by taking the absolute of the difference of numerical results with exact data in Fig. 2. We also plotted the numerical errors for the other two smaller $\alpha = 10^{-4}, 10^{-5}$.

Let us further investigate some very severly ill-posed cases of this benchmark BHCP, where $T = 1.5, 2.5, 3$ sec were employed, such that when the final data are in the order of $O(10^{-7})$–$O(10^{-13})$ we want to use LGSM to retrieve the desired initial data $\sin \alpha x$, which is in the order of $O(1)$.

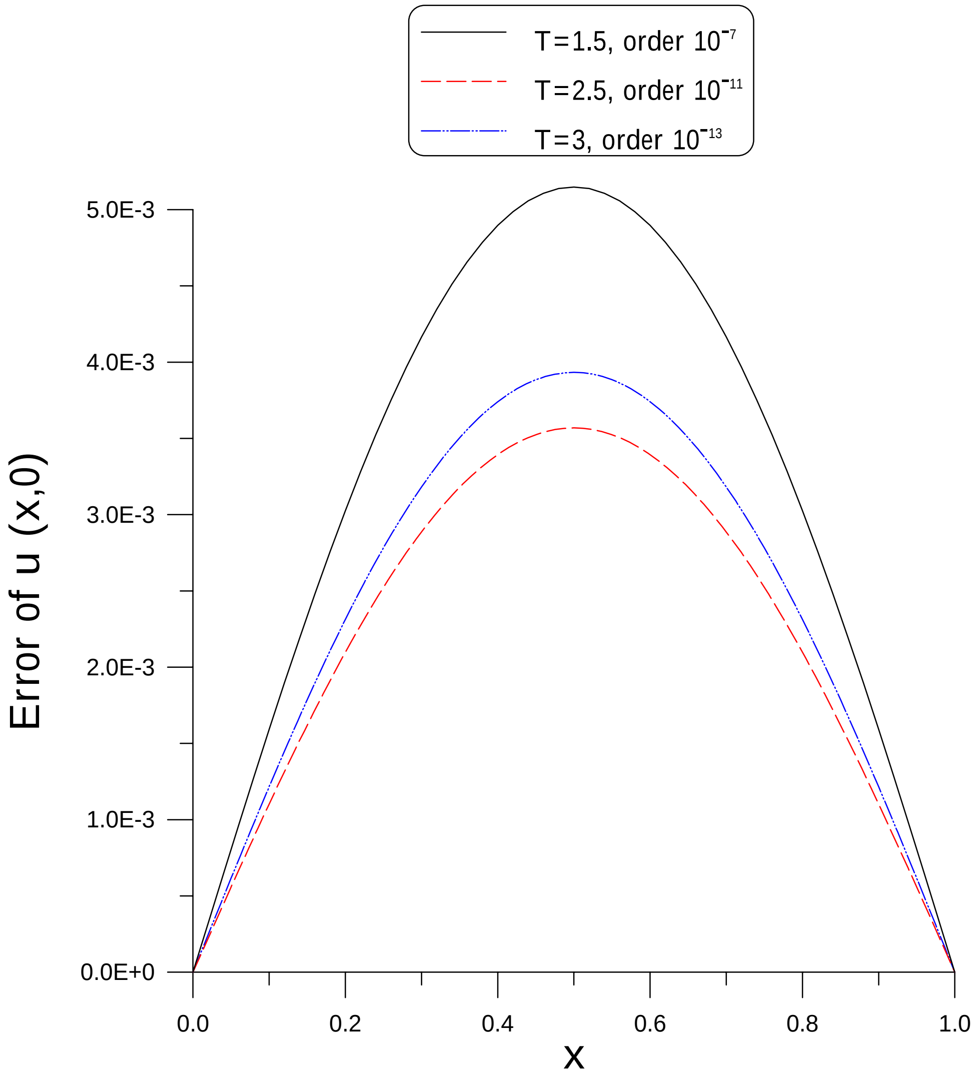
In Fig. 3 we show the numerical errors for these three cases. The maximum error for the case of $T = 3$ sec is about 3.5×10^{-3}. Even for the severe case up to $T = 3$ sec, the computation leads to the maximum error occurring at $x = 0.5$ about 0.0035.



(a)
u (x, T)
0.4
0.2
0.0
Numerical
Exact
(b)
u (x,0)
1.0
0.5
0.0
0.0
0.2
0.4
0.6
0.8
1.0
X



(a)
Error of u (x,T)
9E-4
6E-4
3E-4
0E+0
α=0.001
α=0.0001
α=0.00001
(b)
Error of u (x,0)
3E-3
2E-3
1E-3
0E+0
0.0
0.2
0.4
0.6
0.8
1.0
X


T=1.5, order 10^-7
T=2.5, order 10^-11
T=3, order 10^-13
5.0E-3
4.0E-3
3.0E-3
2.0E-3
1.0E-3
0.0E+0
Error of u (x,0)
0.0
0.2
0.4
0.6
0.8
1.0
x

To my best knowledge, there has no report that the numerical methods for this severly ill-posed BHCP can provide more accurate results than us.

The other topics that could be applied by the LGSM are elastic buckling, doubly-connected torsion problem, boundary layer equations, multiple solutions of nonlinear BVPs.

4. One-Step Estimation Method (OSEM) for Inverse Problems

The one-step estimation method (OSEM) means that we can use one-step Lie group method to estimate the unknown parameter or function in PDEs.

The detail is given in

Chein-Shan Liu, 2006, One-step GPS for the estimation of temperature-dependent thermal conductivity, Int. J. Heat Mass Transf., on line available.

The problem is to estimate $k(u)$ in

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left[k(u)\frac{\partial u}{\partial x}\right], \tag{63}$$

or written as

$$u_t = k'(u)u_x^2 + k(u)u_{xx}. \tag{64}$$

A nonlinear PDE.

The semi-discretization is

$$\dot{u}_i(t) = \frac{1}{(\Delta x)^2}\{k_{i+1}[u_{i+1}(t) - u_i(t)] - k_i[u_i(t) - u_{i-1}(t)]\} \quad (65)$$

with coefficients $k_i = k(u_i),\ i = 1, \ldots, n$ unknown.

One step GPS is

$$\mathbf{X}_T = \mathbf{G}(K\Delta t)\mathbf{X}_0 = \mathbf{G}(T)\mathbf{X}_0. \quad (66)$$

This is a one-step transformation from $\mathbf{X}_0$ to $\mathbf{X}_T$.

One feasible method to calculate $\mathbf{G}(T)$ is given by

$$\mathbf{G}(T) = \exp[T\mathbf{A}(0)] = \begin{bmatrix} \mathbf{I}_n + \frac{(a-1)}{\|\mathbf{f}_0\|^2}\mathbf{f}_0\mathbf{f}_0^{\mathrm{t}} & \frac{b\mathbf{f}_0}{\|\mathbf{f}_0\|} \\ \frac{b\mathbf{f}_0^{\mathrm{t}}}{\|\mathbf{f}_0\|} & a \end{bmatrix}, \quad (67)$$

where

$$a := \cosh\left(\frac{T\|\mathbf{f}_0\|}{\|\mathbf{u}_0\|}\right), \quad b := \sinh\left(\frac{T\|\mathbf{f}_0\|}{\|\mathbf{u}_0\|}\right). \quad (68)$$

Test the accuracy for direct problem by using one-step GPS with $k = 1$ and

$$k(u) = 1 + 4.5 \exp\left(\frac{u}{80}\right) + 2.5 \sin\left(\frac{u}{5}\right).$$

Of course $T = \Delta t$ can not be too large.

For $k = 1$ we have closed-form solution. We compare the test for $T = 0.4$ sec. One-step GPS give result with error in the order 10^{-6}. In order to get the same accurate result RK4 requires 40000 steps, i.e., its time step is $\Delta t = 0.00001$ sec. If we use one-step RK4 and Euler method, the results are very bad.

For

$$k(u) = 1 + 4.5 \exp\left(\frac{u}{80}\right) + 2.5 \sin\left(\frac{u}{5}\right),$$

we have no closed-form solution. We produce the "exact solution" by RK4 with very small time step size $\Delta t = 0.00001$ sec.

Then the results of one-step GPS for $T = 0.004$ and $T = 0.01$ are compared with one-step Euler method.

Why should we use the one-step method to estimate the unknown $k(u)$. Numerical integration is essentially an iterative process to find the solution at any desired time T. If many steps are used to estimate the unknowns, it is impossible to solve these coupling equations. However if it is only one-step, we can solve

$$u_i^T = u_i^0 + \frac{\eta}{(\Delta x)^2}\{k_{i+1}[u_{i+1}^0 - u_i^0] - k_i[u_i^0 - u_{i-1}^0]\}, \qquad (69)$$

and η is fully determined by the data u_i^T and u_i^0.

This method we call one-step estimation method (OSEM). The advantages are that **it does not require any prior information on the functional forms of thermal conductivity; no initial guesses are required; no itera-**

tions are required; and the inverse problem can be solved in a linear domain with closed-form solution.

The estimation is very good as shown below. It is rather promising used the OSEM to estimate the parameters or functions in the evolution type PDEs.

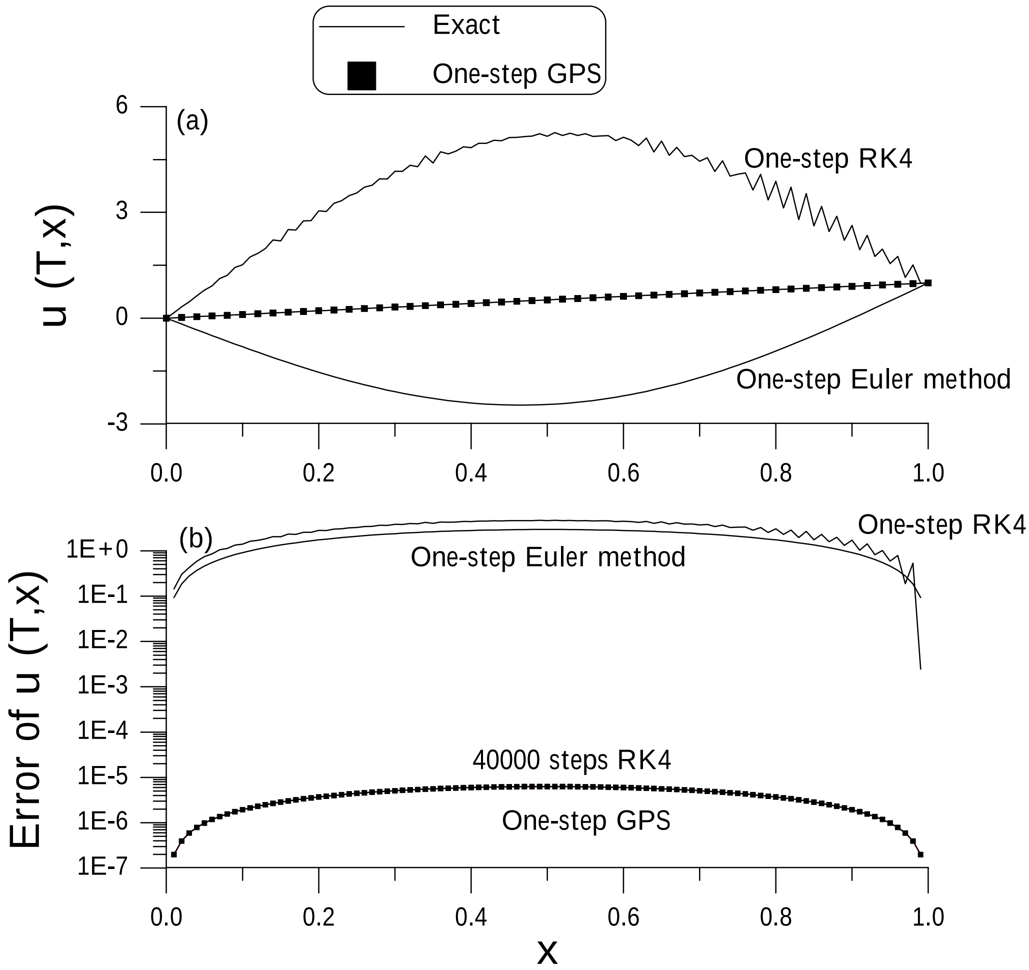


Fig. 1. Comparing numerical solutions of one-step GPS, RK4 and Euler methods for Example 1 in (a), and the numerical errors were compared in (b).

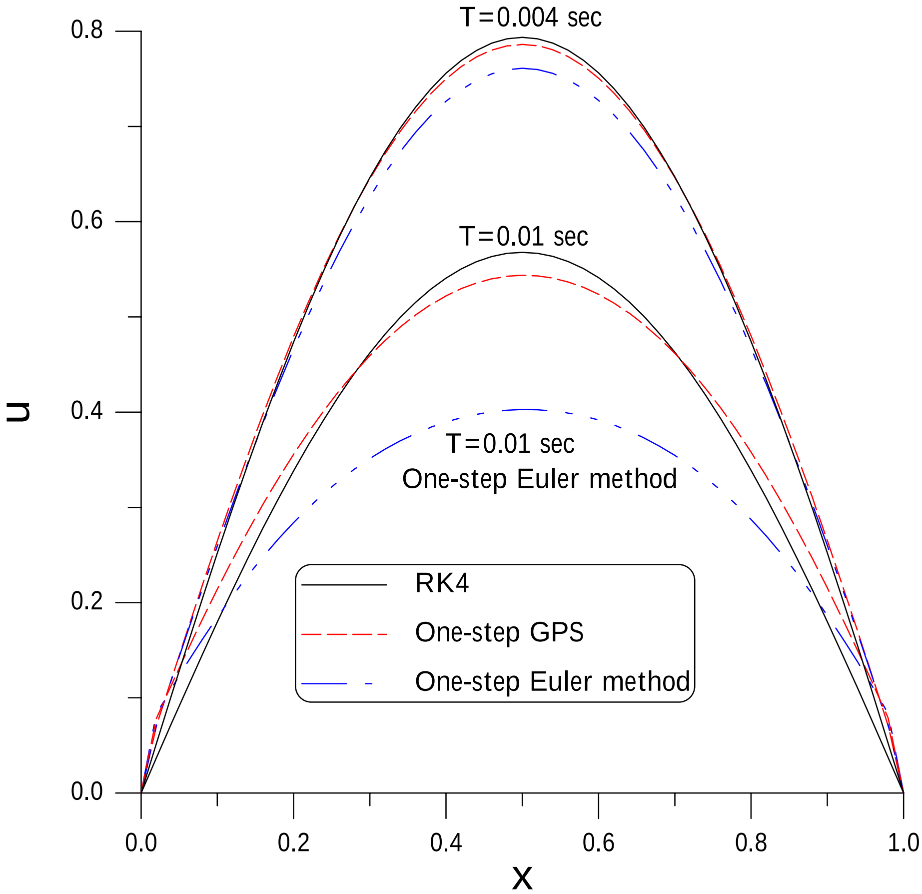


Fig. 2. Comparing numerical solutions of one-step GPS and Euler method for Example 2 with ‘‘exact’’ solutions calculated by RK4.

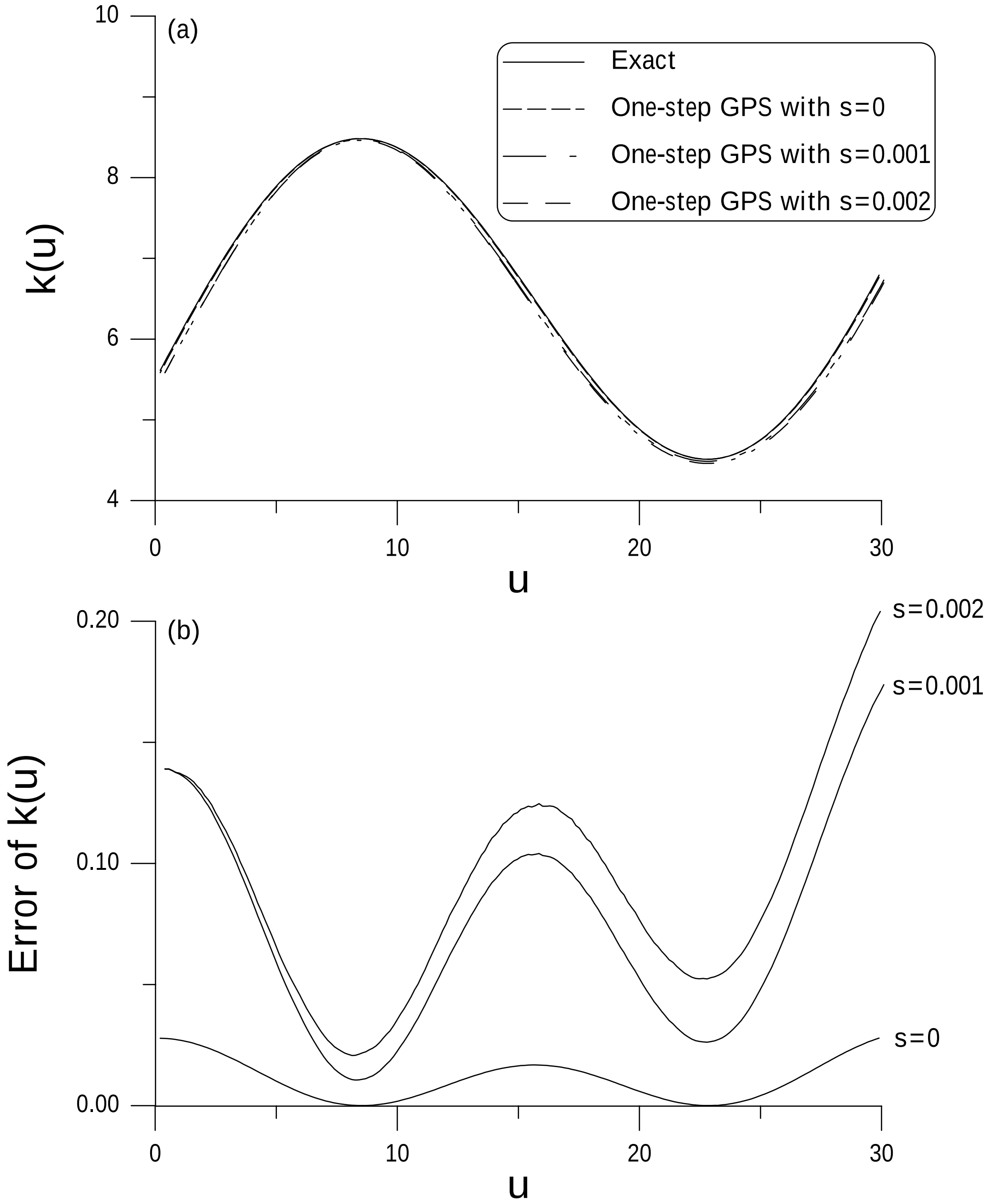


Fig. 3. Estimating k(u) for Example 2 with different noise levels: (a) comparing exact solution and numerical solutions calculated by one-step GPS, (b) numerical errors.

5. Conclusions

1. The Lie group method (GPS) together with the cone structure for ODEs have developed since 2001. Its application on the integration of nonlinear engineering problems is still not very popular. However, the GPS including the quantities $\|\mathbf{x}\|$, $\|\mathbf{f}\|$ and $\mathbf{x}\cdot\mathbf{f}$ in its formulation

$$\mathbf{x}(\ell+1)=\mathbf{x}(\ell)+\frac{2\tau^2\mathbf{f}(\ell)\cdot\mathbf{x}(\ell)+2\tau\|\mathbf{x}(\ell)\|^2}{\|\mathbf{x}(\ell)\|^2-\tau^2\|\mathbf{f}(\ell)\|^2}\mathbf{f}(\ell)$$

may deserve more study on its geometric properties and structures, which as I know is very different from the conventional numerical integration methods. Recently, a study to appear in the Journal of Sound and Vibration, indicates that the GPS and its extension can reflect the chaotical phenomenon for chaotic systems.

2. A complete cone picture: future cone and past cone, is just brought out by myself in the recent two papers in 2006. In addition to the application to the backward problems, I believe that the topic about the past cone dynamics would be very interest. That is, if we reverse the time direction of all human-made dynamical equations, what picture can be seen. Does there have the so-called past time dynamics in the universe ?

3. In the last year the Lie group shooting method is developed, which broadly extend our calculations and studies from the initial value problems to the boundary value problems including ODEs and PDEs. Up to now for the second order ODEs the unknown initial conditions can be solved exactly without iterations as shown in the paper in CMES.

However, I have a plain to extend it to the higher-order equations without iterations to find all missing initial conditions.

4. One-step estimation method (OSEM) for the inverse problem of parameter identification has been developed. Due to its many advantages over other estimation methods, I suggest to use it on the hyperbolic equations in the near future, and even for the very difficult elliptical type inverse problems.

5. Thanks for your sincere attention in this reviewing talk about my recent works on the Lie group methods in engineering problems, which is the first time that I contributed such field in the public area. Please also take care my most papers on the **plasticity theory**.