

An LGDAE Method to Solve Nonlinear Cauchy Problem Without Initial Temperature

Chein-Shan Liu¹

Abstract: We recover an unknown initial temperature for a nonlinear heat conduction equation $u_t(x, t) = u_{xx}(x, t) + H(x, t, u, u_x)$, under the Cauchy boundary conditions specified on the left-boundary. The method in the present paper transforms the Cauchy problem into an inverse heat source problem to find $F(x)$ in $T_t(x, t) = T_{xx}(x, t) + H + F(x)$. By using the $GL(N, \mathbb{R})$ Lie-group differential algebraic equations (LGDAE) algorithm to integrate the numerical method of lines discretized equations from sideways heat equation, we can fast recover the initial temperature and two boundary conditions on the right-boundary. The accuracy and efficiency are confirmed by comparing the exact solutions with the recovered results, where a large noisy disturbance is imposed on the Cauchy data.

Keywords: Cauchy problem, Missing initial condition, Lie-group differential algebraic equations (LGDAE) method, Sideways heat equation.

1 Introduction

In this paper we consider an inverse heat conduction problem (IHCP) by recovering an unknown initial condition for a nonlinear heat conduction equation under the Cauchy type boundary conditions:

$$u_t(x, t) = u_{xx}(x, t) + H(x, t, u, u_x), \quad 0 < x < \ell, \quad t > 0, \quad (1)$$

$$u(0, t) = u_0(t), \quad (2)$$

$$u_x(0, t) = q_0(t), \quad (3)$$

where H may be a nonlinear function of u and u_x , and the initial condition is missing to be an unknown function of x :

$$u(x, 0) = f(x) = ? \quad (4)$$

¹ Department of Civil Engineering, National Taiwan University, Taipei 106-17, Taiwan. Email: liucs@ntu.edu.tw

For the compatibility of data we require that $f(0) = u_0(0)$ and $f_x(0) = q_0(0)$.

The sideways heat equation means the Cauchy problem for the heat conduction equation, in which the temperature and heat flux are specified as functions of time at one end of the two boundaries [Dorroh and Ru (1999); Chang, Liu and Chang (2005)]. Besides the above two Cauchy boundary conditions in Eqs. (2) and (3) for a typical sideways heat equation, the present method does not need other data to recover $f(x)$, $u(\ell, t)$ and $u_x(\ell, t)$.

In the theory of partial differential equations (PDEs) there are two kinds Cauchy problems. One is for the elliptic type PDE, and another is for the parabolic type PDE; they are subjected to incomplete boundary conditions with some portion un-specifying but some portion over-specifying. Both are of the non-characteristic type initial value problems. It is well known that the Cauchy problems are highly ill-posed with a little error of input data producing a large error of numerical solution [Eldén (1987); Eldén, Berntsson and Reginska (2000); Qian and Fu (2007); Hao, Reinhardt and Schneider (2001); Liu (2008a, 2008b); Chi, Yeih and Liu (2009); Marin (2009); Liu and Kuo (2011); Liu, Kuo and Liu (2011); Liu and Zhang (2013); Yeih, Chan, Fan, Chang and Liu (2014)].

In many industrial applications we may want to determine the temperature and heat flux on the surface of a body, but the surface itself is inaccessible for a measurement of temperature or heat flux. It may also be the case that locating a measurement device on the surface would disturb the measurements so that an incorrect temperature or heat flux is recorded. In such a situation one is restricted to internal measurements which being carried out on an accessible boundary. Berntsson (2003) has presented an example of industrial application where the sideways heat equation can be used. Sometimes we may encounter the problem that when we tackle the sideways heat equation the measurement of initial temperature is impossible, because the heat conducting device is already in service.

Wang, Cheng, Nakagawa and Yamamoto (2010) have treated the IHCP without needing of initial condition, and proved the uniqueness in determining both a boundary value and an initial value for linear sideways heat equation. Liu (2011) has studied the problem by recovering the initial condition under boundary conditions given at two boundaries. Without needing of initial condition, Liu (2014a) has proposed an iterative method to recover the heat conductivity of a nonlinear heat conduction equation, and Liu (2014b) has proposed an iterative algorithm to recover heat source under the Cauchy type boundary conditions and a final time condition. Recently, Liu and Wang (2014) have proposed a quasi-reversibility regularization method to obtain a regularized solution and convergence estimates without needing of initial temperature for the linear Cauchy problem.

As we know, there are very few methods to deal with the nonlinear sideways heat equation without initial conduction, which is very difficult to be solved numerically. In this paper we will provide a simple and yet stable numerical method to solve this highly ill-posed nonlinear Cauchy problem with multiple unknowns of initial condition and right-boundary conditions.

The outline of this paper is given as follows. In Section 2 we propose a variable transformation, such that the nonlinear Cauchy problem without initial value becomes a nonlinear inverse heat source problem with zero initial value. Sections 3-5 devote to the development of a Lie-group differential algebraic equations (LGDAE) method and a numerical algorithm based on the Lie-group $GL(N, \mathbb{R})$ for the general DAEs system. In Section 6, we view the nonlinear inverse heat source problem in Section 2 as a special type nonlinear DAEs, and derive a numerical algorithm to solve the resultant DAEs. Numerical examples are given in Section 7 to validate the efficiency and accuracy of LGDAE. Some conclusions are drawn in the final Section 8.

2 A variable transformation and the numerical method of lines

Let

$$T(x, t) = u(x, t) - f(x), \tag{5}$$

where $f(x)$ is an unknown function of initial temperature to be determined. From Eqs. (1)-(4) it follows that

$$T_t(x, t) = T_{xx}(x, t) + H(x, t, T + f, T_x + f') + F(x), \tag{6}$$

$$T(0, t) = u_0(t) - u_0(0), \tag{7}$$

$$T_x(0, t) = q_0(t) - q_0(0), \tag{8}$$

$$T(x, 0) = 0, \tag{9}$$

where $F(x) = f''(x)$ is viewed as an unknown spatially-dependent function of heat source.

Here, a novel method of Lie-group differential algebraic equations (LGDAE) method will be developed to estimate the unknown heat source $F(x)$ and unknown initial condition $f(x)$, which merely requires the boundary conditions and initial condition given by Eqs. (7)-(9) for estimating $F(x)$, and the boundary conditions given by Eqs. (2) and (3) for estimating $f(x)$.

The numerical method of lines is simple that for a given system of PDEs we discretize all but one of the independent variables. The semi-discrete procedure yields

a coupled system of ordinary differential equations (ODEs), which are then being numerically integrated to obtain solution. For Eq. (6) we adopt the numerical method of lines to discretize the time coordinate t by t_i , by letting $S^i(x) = T_x(x, t_i)$ and $T^i(x) = T(x, t_i)$, and keep x a continuous variable, where $t_i = (i - 1)\Delta t = (i - 1)t_f/(m - 1)$ and Δt is a uniform time-stepsize. Such that we can derive

$$\frac{\partial T^i(x)}{\partial x} = S^i(x), \quad i = 1, \dots, m + 1, \tag{10}$$

$$\frac{\partial S^1(x)}{\partial x} = -\frac{3T^1 - 4T^2 + T^3}{2\Delta t} - H_1 - F(x), \quad i = 1, \tag{11}$$

$$\frac{\partial S^i(x)}{\partial x} = \frac{T^{i+1} - T^{i-1}}{2\Delta t} - H_i - F(x), \quad i = 2, \dots, m - 1, \tag{12}$$

$$\frac{\partial S^m(x)}{\partial x} = \frac{3T^m - 4T^{m-1} + T^{m-2}}{2\Delta t} - H_m - F(x), \quad i = m, \tag{13}$$

$$\frac{\partial S^{m+1}(x)}{\partial x} = F(x), \tag{14}$$

where $H_i = H(x, t_i, T^i + f, S^i + f') = H(x, t_i, T^i + T^{m+1}, S^i + S^{m+1})$ by defining $T^{m+1}(x) = f(x)$ and $S^{m+1}(x) = f'(x)$. If we can know $F(x)$, we can integrate Eqs. (10)-(14) by using the group preserving scheme (GPS) developed by Liu (2001), where we denote the spatial stepsize by $\Delta x = \ell/(n - 1)$. Besides the above ODEs we have a constraint equation:

$$T^1(x) = 0, \tag{15}$$

which is obtained from Eq. (9). Here, we must emphasize that $T^1(x)$ plays two roles: satisfying Eqs. (10) and (15). Recently, Liu (2014c) has argued that the Cauchy problem of heat equation is solvable, because the field equation is extendable to the initial time by using the concept of analytic continuation. The above technique to treat the temperature $T^1(x)$ on the line at initial time is indeed an application of the analytic continuation method.

3 Lie-group differential algebraic equations method

Eqs. (10)-(15) constitute a set of differential algebraic equations (DAEs) with $F(x)$ to be an unknown function. Here we generalize the above DAEs to

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{y}, t), \quad \mathbf{x}(0) = \mathbf{x}_0, \quad t \in \mathbb{R}, \quad \mathbf{x} \in \mathbb{R}^N, \quad \mathbf{y} \in \mathbb{R}^q, \tag{16}$$

$$\mathbf{F}(\mathbf{x}, \mathbf{y}, t) = \mathbf{0}, \quad \mathbf{F} \in \mathbb{R}^q, \tag{17}$$

and propose a novel method to solve the above DAEs, which govern the evolution of $N + q$ variables $x_i, i = 1, \dots, N$ and $y_j, j = 1, \dots, q$ with N ordinary differential equations (ODEs) and q nonlinear algebraic equations (NAEs). The vector \mathbf{y} in Eq. (16) is viewed as unknown parameter. We have replaced x in Eqs. (10)-(15) by t for a purpose of the demonstration for the general DAEs written in a time domain.

There are many numerical methods used to solve ODEs. But only a few is used to solve DAEs. The present technique used to solve the inverse heat source problem is a DAE method, whose pre-requirement is however a powerful and stable method to solve the DAEs as to be described below. The DAEs are more difficult to be solved numerically than ODEs and NAEs.

Liu (2013a) was the first to find the essential form for n -dimensional nonlinear ordinary differential equations (ODEs) in terms of the Lie-algebra $gl(n, \mathbb{R})$ of $GL(n, \mathbb{R})$, and developed a very effective Lie-group $GL(n, \mathbb{R})$ preserving scheme to solve ODEs. Then, Liu (2013b) developed a Lie-group $GL(n, \mathbb{R})$ preserving scheme to solve ODEs by assuming that the coefficient matrix is constant in a small time incremental step. Moreover, Liu (2013c) has developed a powerful numerical method to solve the nonlinear DAEs based on the above Lie-group $GL(n, \mathbb{R})$ preserving scheme, which is named the Lie-group DAE (LGDAE) method. It is also interesting that the LGDAE can be used to solve the sliding control problem by Liu (2014d). Liu and Atluri (2013) have employed the LGDAE to solve the problem of numerical differential of a noisy signal.

4 The $GL(N, \mathbb{R})$ structure of differential equations system

The Lie-group is a differentiable manifold, which is endowed with a group structure that is compatible with the underlying topology of the manifold. The Lie-group solver can provide a better algorithm that retains the orbit generated from numerical solution on the manifold which is associated with the Lie-group.

The general linear group is a Lie group, whose manifold is an open subset $GL(N, \mathbb{R}) := \{\mathbf{G} \in \mathbb{R}^{N \times N} | \det \mathbf{G} \neq 0\}$ of the linear space of all $N \times N$ non-singular matrices. Thus, $GL(N, \mathbb{R})$ is also an $N \times N$ -dimensional manifold.

The general linear group $GL(N, \mathbb{R})$ gives uniquely a real Lie-algebra $gl(N, \mathbb{R})$. Consider a one-parameter subgroup $\mathbf{G}(t), t \in \mathbb{R}$, of the general linear group $GL(N, \mathbb{R})$, which is a curve passing through the group identity at $t = 0$,

$$\mathbf{G}(0) = \mathbf{I}_N, \tag{18}$$

and which operates from the left on the N -dimensional Euclidean space \mathbb{R}^N , resulting in a Lie-group equation:

$$\dot{\mathbf{G}}(t) = \mathbf{A}(t)\mathbf{G}(t), \tag{19}$$

where $\mathbf{A} \in gl(N, \mathbb{R})$ is the corresponding Lie-algebra.

Here we give a new form of the dynamics in Eq. (16) from the $GL(N, \mathbb{R})$ Lie-group structure. In order to fit the form in Eq. (19), the vector field \mathbf{f} on the right-hand side of Eq. (16) can be written as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}, \tag{20}$$

where

$$\mathbf{A} = \frac{\mathbf{f}}{\|\mathbf{x}\|} \otimes \frac{\mathbf{x}}{\|\mathbf{x}\|} \tag{21}$$

is the coefficient matrix. Here $\mathbf{u} \otimes \mathbf{y}$ denotes the dyadic operation of \mathbf{u} and \mathbf{y} , i.e., $(\mathbf{u} \otimes \mathbf{y})\mathbf{z} = \mathbf{y} \cdot \mathbf{z}\mathbf{u}$.

Because the coefficient matrix \mathbf{A} is well-defined for $\|\mathbf{x}\| > 0$, the Lie-group element \mathbf{G} generated from the above dynamical system (20) with $\dot{\mathbf{G}} = \mathbf{A}\mathbf{G}$ satisfies $\det \mathbf{G}(t) \neq 0$, such that $\mathbf{G} \in GL(N, \mathbb{R})$.

5 An implicit $GL(N, \mathbb{R})$ Lie-group scheme

Eq. (20) is a new starting point for the development of the Lie-group $GL(N, \mathbb{R})$ scheme. In order to develop a numerical scheme from Eqs. (20) and (21), we suppose that the coefficient matrix \mathbf{A} is constant with

$$\mathbf{a} = \frac{\bar{\mathbf{f}}}{\|\bar{\mathbf{x}}\|}, \quad \mathbf{b} = \frac{\bar{\mathbf{x}}}{\|\bar{\mathbf{x}}\|} \tag{22}$$

being two constant vectors, which can be obtained by taking the values of \mathbf{f} and \mathbf{x} at a suitable mid-point of $\bar{t} \in [t_0 = 0, t]$, where $t \leq t_0 + h$ and h is a small time stepsize. Thus from Eqs. (20) and (21) it follows that

$$\dot{\mathbf{x}} = \mathbf{b} \cdot \mathbf{x}\mathbf{a}. \tag{23}$$

Let

$$w = \mathbf{b} \cdot \mathbf{x}, \tag{24}$$

and Eq. (23) becomes

$$\dot{\mathbf{x}} = w\mathbf{a}. \tag{25}$$

At the same time, from the above two equations we can derive the following ODE for w :

$$\dot{w} = cw, \tag{26}$$

where

$$c = \mathbf{a} \cdot \mathbf{b} \tag{27}$$

is supposed to be a constant value in a small time interval of $t \in [t_0, t_0 + h]$. Thus, we have

$$w(t) = w_0 \exp(ct), \quad t \in [t_0, t_0 + h], \tag{28}$$

where $w_0 = \mathbf{b} \cdot \mathbf{x}_0$.

Inserting Eq. (28) for $w(t)$ into Eq. (25) and integrating the resultant equation we can obtain

$$\mathbf{x}(t) = [\mathbf{I}_N + \eta(t)\mathbf{a}\mathbf{b}^T]\mathbf{x}_0, \tag{29}$$

where \mathbf{x}_0 is the initial value of \mathbf{x} at an initial time $t = t_0 = 0$, and

$$\eta(t) = \frac{e^{ct} - 1}{c}. \tag{30}$$

Let \mathbf{G} be the coefficient matrix before \mathbf{x}_0 in Eq. (29), i.e.,

$$\mathbf{G} = \mathbf{I}_N + \eta\mathbf{a}\mathbf{b}^T. \tag{31}$$

Then we can prove

$$\det \mathbf{G} = e^{ct} > 0, \tag{32}$$

which means that \mathbf{G} is a Lie-group element of $GL(N, \mathbb{R})$.

Accordingly, we can develop the following scheme based on the Lie-group $GL(N, \mathbb{R})$ for solving the ODEs in Eq. (16):

- (i) Give $0 \leq \theta \leq 1$.
- (ii) Give an initial value of \mathbf{x}_0 at an initial time $t = t_0$ and a time stepsize h .
- (iii) For $k = 0, 1, \dots$, we repeat the following computations to a specified terminal time $t = t_f$:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + h\mathbf{f}_k. \tag{33}$$

With the above \mathbf{x}_{k+1} generated from an Euler step as an initial guess we iteratively

solve the new \mathbf{x}_{k+1} by

$$\begin{aligned} \bar{\mathbf{x}}_k &= (1 - \theta)\mathbf{x}_k + \theta\mathbf{x}_{k+1}, \\ \mathbf{a}_k &= \frac{\bar{\mathbf{f}}_k}{\|\bar{\mathbf{x}}_k\|} = \frac{\mathbf{f}(\bar{\mathbf{x}}_k, \mathbf{y}_k, t_k + \theta h)}{\|\bar{\mathbf{x}}_k\|}, \\ \mathbf{b}_k &= \frac{\bar{\mathbf{x}}_k}{\|\bar{\mathbf{x}}_k\|}, \\ c_k &= \mathbf{a}_k \cdot \mathbf{b}_k, \\ \eta_k &= \frac{\exp(c_k h) - 1}{c_k}, \\ \mathbf{z}_{k+1} &= \mathbf{x}_k + \eta_k \mathbf{b}_k \cdot \mathbf{x}_k \mathbf{a}_k. \end{aligned} \tag{34}$$

If \mathbf{z}_{k+1} converges according to a given stopping criterion:

$$\|\mathbf{z}_{k+1} - \mathbf{x}_{k+1}\| < \varepsilon_2, \tag{35}$$

then go to (iii) to the next time step; otherwise, let $\mathbf{x}_{k+1} = \mathbf{z}_{k+1}$ and go to Eq. (34). In the above \mathbf{y}_k is viewed as a constant vector within a small time step. In order to determine \mathbf{y}_k the present Lie-group scheme is coupled with the following Newton iterative scheme.

Within a small time step we can suppose that the variables $y_j, j = 1, \dots, q$ are constant in the interval of $t_k < t < t_{k+1}$. We give an initial guess of $y_j, j = 1, \dots, q$, and insert them into Eq. (16). Then we apply the above implicit $GL(N, \mathbb{R})$ Lie-group scheme to find the next \mathbf{x}_{k+1} , by supposing that \mathbf{x}_k and \mathbf{y}_k are already obtained in the previous time step. When \mathbf{x}_{k+1} are available we insert them into Eq. (17), and then apply the Newton iterative scheme to solve \mathbf{y}_{k+1} by

$$\mathbf{y}_{k+1}^{\ell+1} = \mathbf{y}_{k+1}^\ell - \mathbf{B}^{-1} \mathbf{F}(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}^\ell, t_{k+1}), \tag{36}$$

until the following convergence criterion is satisfied:

$$\|\mathbf{y}_{k+1}^{\ell+1} - \mathbf{y}_{k+1}^\ell\| < \varepsilon_1; \tag{37}$$

otherwise, go to Eq. (34) and insert the new $\mathbf{y}_{k+1}^{\ell+1}$. In above, the component B_{ij} of the Jacobian matrix \mathbf{B} is given by $\partial F_i / \partial y_j$.

The numerical scheme is a combination of the Lie-group method based on $GL(N, \mathbb{R})$ and the Newton iterative method to solve the DAEs in Eqs. (16) and (17), which is called the Lie-group differential algebraic equations (LGDAE) method.

6 Numerical algorithm

Now we apply the above LGDAE with $N = 2m + 2$ and $q = 1$ to solve $\mathbf{T} = (T^1, \dots, T^m, T^{m+1})^T$, $\mathbf{S} = (S^1, \dots, S^m, S^{m+1})^T$ and hence $f(x) = T^{m+1}(x)$ through Eqs. (10)-(14), and simultaneously solve the unknown function $F(x)$ through Eq. (15) by the Newton iterative method. The numerical processes are given below:

- (i) Give an initial guess of F_0 , for example, $F_0 = 0$.
- (ii) Give initial conditions of \mathbf{T}_0 and \mathbf{S}_0 at an initial point $x = 0$ and a spatial stepsize Δx .
- (iii) For $k = 0, 1, \dots$, we repeat the following computations to a terminal point $x = \ell$:

$$\begin{aligned} \mathbf{T}_{k+1} &= \mathbf{T}_k + \Delta x \mathbf{S}_k, \\ \mathbf{S}_{k+1} &= \mathbf{S}_k + \Delta x \mathbf{f}_k, \end{aligned}$$

where \mathbf{f}_k denotes the k -th step value of the right-hand side in Eqs. (11)-(14). With the above \mathbf{T}_{k+1} and \mathbf{S}_{k+1} generated from an Euler step as an initial guess we then iteratively solve the new \mathbf{T}_{k+1} and \mathbf{S}_{k+1} by

$$\begin{aligned} \bar{\mathbf{T}}_k &= (1 - \theta)\mathbf{T}_k + \theta\mathbf{T}_{k+1}, \\ \bar{\mathbf{S}}_k &= (1 - \theta)\mathbf{S}_k + \theta\mathbf{S}_{k+1}, \\ \mathbf{a}_k^t &= \frac{\bar{\mathbf{S}}_k}{\sqrt{\|\bar{\mathbf{T}}_k\|^2 + \|\bar{\mathbf{S}}_k\|^2}}, \\ \mathbf{a}_k^s &= \frac{\bar{\mathbf{f}}_k}{\sqrt{\|\bar{\mathbf{T}}_k\|^2 + \|\bar{\mathbf{S}}_k\|^2}}, \\ \mathbf{b}_k^t &= \frac{\bar{\mathbf{T}}_k}{\sqrt{\|\bar{\mathbf{T}}_k\|^2 + \|\bar{\mathbf{S}}_k\|^2}}, \\ \mathbf{b}_k^s &= \frac{\bar{\mathbf{S}}_k}{\sqrt{\|\bar{\mathbf{T}}_k\|^2 + \|\bar{\mathbf{S}}_k\|^2}}, \\ c_k &= \mathbf{a}_k^t \cdot \mathbf{b}_k^t + \mathbf{a}_k^s \cdot \mathbf{b}_k^s, \\ d_k &= \mathbf{T}_k \cdot \mathbf{b}_k^t + \mathbf{S}_k \cdot \mathbf{b}_k^s, \\ \eta_k &= \frac{\exp(c_k \Delta x) - 1}{c_k}, \\ \mathbf{z}_{k+1}^t &= \mathbf{T}_k + \eta_k d_k \mathbf{a}_k^t, \\ \mathbf{z}_{k+1}^s &= \mathbf{S}_k + \eta_k d_k \mathbf{a}_k^s. \end{aligned} \tag{38}$$

If \mathbf{z}_{k+1}^t and \mathbf{z}_{k+1}^s converge according to a given stopping criterion:

$$\sqrt{\|\mathbf{z}_{k+1}^t - \mathbf{T}_{k+1}\|^2 + \|\mathbf{z}_{k+1}^s - \mathbf{S}_{k+1}\|^2} < \varepsilon_2, \tag{39}$$

then go to (iv); otherwise, let $\mathbf{T}_{k+1} = \mathbf{z}_{k+1}^t$ and $\mathbf{S}_{k+1} = \mathbf{z}_{k+1}^s$ and go to Eq. (38).

(iv) For $j = 0, 1, \dots$, we repeat the following computations:

$$F_{k+1}^{j+1} = F_{k+1}^j - \frac{E_j}{E_j'}, \tag{40}$$

where the prime denotes the differential with respect to F , and

$$\begin{aligned} (\mathbf{a}_k^s)' &= \frac{\bar{\mathbf{f}}_k'}{\sqrt{\|\bar{\mathbf{T}}_k\|^2 + \|\bar{\mathbf{S}}_k\|^2}}, \\ c_k' &= \mathbf{b}_k^s \cdot (\mathbf{a}_k^s)', \\ \eta_k' &= \frac{c_k'[(hc_k - 1)\exp(c_k h) + 1]}{c_k^2}, \\ \mathbf{T}_{k+1}' &= \eta_k' d_k \mathbf{a}_k^t, \\ E_j &= T_{k+1}^1, \\ E_j' &= (T_{k+1}^1)'. \end{aligned} \tag{41}$$

Obviously, from Eqs. (11)-(14) one has $\bar{\mathbf{f}}_k' = (-\mathbf{1}_m, 1)^T = (-1, \dots, -1, 1)^T$, where $\mathbf{1}_m = (1, \dots, 1)^T$. If F_{k+1}^j converges according to

$$|F_{k+1}^{j+1} - F_{k+1}^j| < \varepsilon_1, \tag{42}$$

then go to (iii) for the next step; otherwise, let $F_{k+1}^j = F_{k+1}^{j+1}$ and go to Eq. (38). The above iteration with ε_1 as a convergence criterion is called the outer iteration or outer loop, while that with ε_2 as a convergence criterion is called the inner iteration or inner loop. In the computations given below the maximum numbers of iterations for inner and outer iterations are fixed to be 100 and 200, respectively. In all the computations given below we will fix the weighting factor θ to be $\theta = 1/2$.

7 Numerical tests

In this section we test the proposed LGDAE in the solution of the Cauchy problem without initial condition. All the required left boundary conditions can be derived from exact solutions. Here we consider the noise being imposed on the left-boundary conditions by

$$\hat{u}_0(t_i) = u_0(t_i)[1 + \sigma R(i)], \quad \hat{q}_0(t_i) = q_0(t_i)[1 + \sigma R(i)], \tag{43}$$

where $R(i)$ are random numbers in $[-1, 1]$, and σ is the intensity of noise.

We define the root-mean-square-error (RMSE) in the recoveries of initial condition and right boundary conditions by

$$\text{RMSE1} := \sqrt{\frac{1}{n} \sum_{j=1}^n [u^*(x_j, 0) - u(x_j, 0)]^2}, \tag{44}$$

$$\text{RMSE2} := \sqrt{\frac{1}{m} \sum_{k=1}^m [u^*(\ell, t_k) - u(\ell, t_k)]^2}, \tag{45}$$

$$\text{RMSE3} := \sqrt{\frac{1}{m} \sum_{k=1}^m [u_x^*(\ell, t_k) - u_x(\ell, t_k)]^2}, \tag{46}$$

where m is the number of the discretized times and n is the number of steps used in the integration of the governing equations along the x -direction. When $u^*(x_j, 0)$, $u^*(\ell, t_k)$ and $u_x^*(\ell, t_k)$ denote the numerical solutions, $u(x_j, 0)$, $u(\ell, t_k)$ and $u_x(\ell, t_k)$ denote the exact solutions.

7.1 Example 1

In this example the exact solution of u is

$$u(x, t) = 1 - \frac{1}{2} \left[e^{x/\sqrt{2}} \cos(x/\sqrt{2} + t) + e^{-x/\sqrt{2}} \cos(x/\sqrt{2} - t) \right], \quad 0 < x < 1, \quad 0 < t < 2\pi. \tag{47}$$

Under the convergence criteria $\varepsilon_2 = 10^{-12}$ for inner iterations and $\varepsilon_1 = 10^{-4}$ for outer iterations, we apply the LGDAE to solve the above problem, where the following parameters: $m = 21$, $n = 251$ and a noise with $\sigma = 0.01$ are considered. As shown in Fig. 1(a), the numbers of iterations are few with one, four or six for inner iterations and one for outer iterations. It shows that the LGDAE is a highly efficient method, which is convergent very fast. In Fig. 1(b) we compare the recovered temperature with the exact one, and in Fig. 1(c) we compare the recovered heat flux with the exact one, which are almost coincident. Thus we plot the numerical error in Fig. 2, of which the maximum error and RMSE1 of initial temperatures are, respectively, 1.7×10^{-2} and 4×10^{-3} , the maximum error and RMSE2 of right-end temperatures are, respectively, 2.1×10^{-2} and 9.9×10^{-3} , and the maximum error and RMSE3 of right-end heat fluxes are 3.4×10^{-2} and 1.6×10^{-2} , respectively.

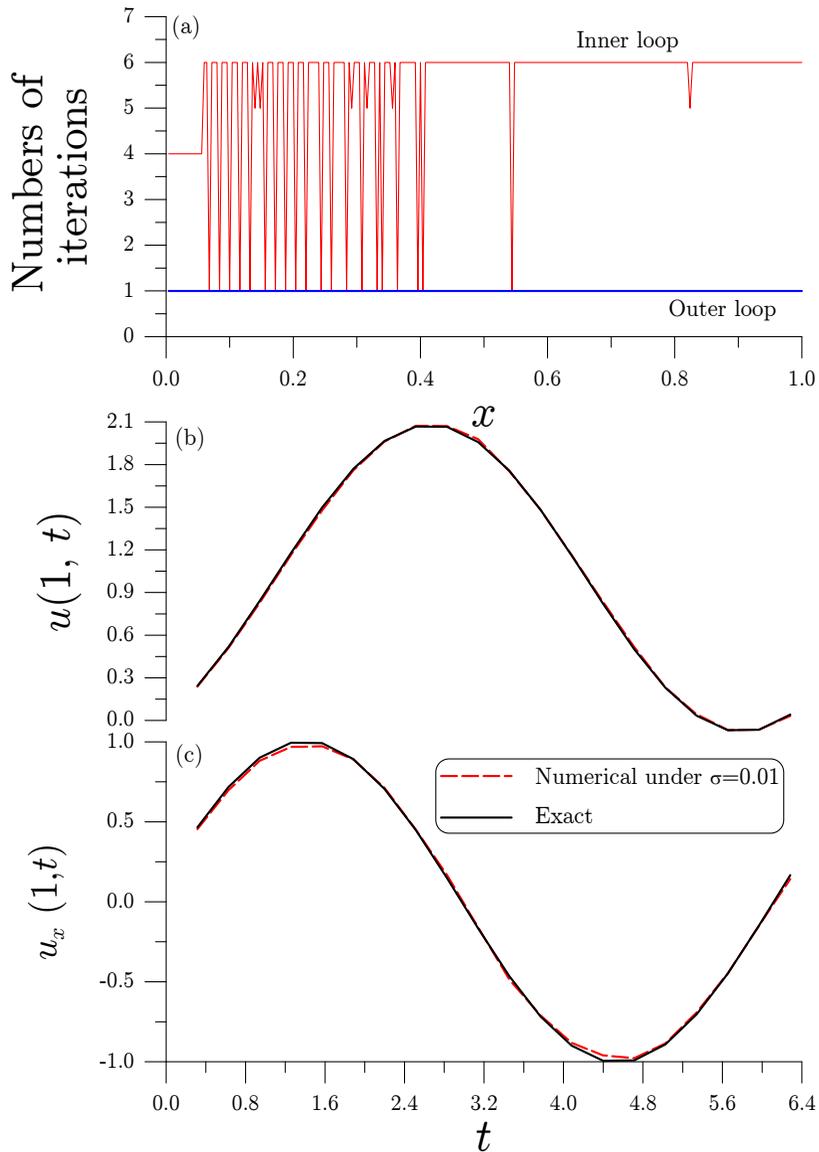


Figure 1: For example 1: (a) the numbers of inner and outer iterations, (b) comparing right-end temperatures, and (c) comparing right-end heat fluxes.

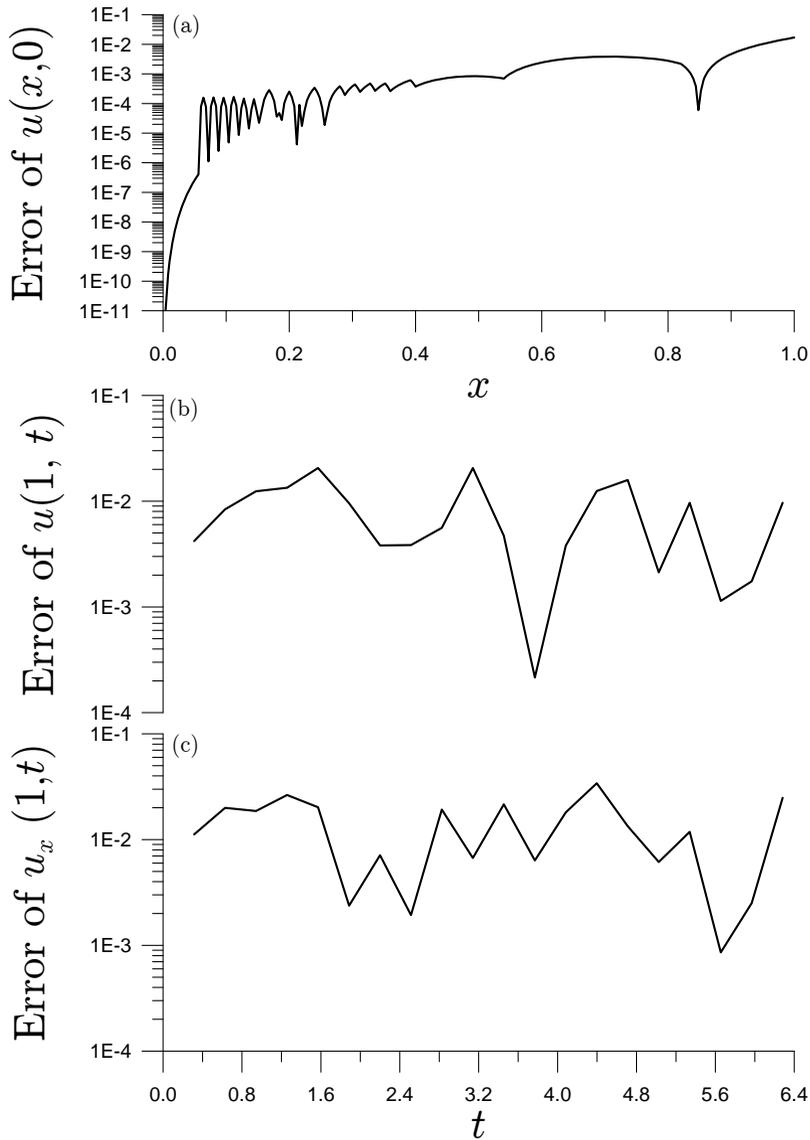


Figure 2: For example 1 showing the numerical errors of (a) initial temperature, (b) right-end temperature, and (c) right-end heat flux.

7.2 Example 2

Next, we assume that the exact solution of u is

$$u(x,t) = \sin(\pi x)e^{-\pi^2 t}, \quad H(x) = 0, \quad 0 < x < 0.5, \quad 0 < t < 1. \quad (48)$$

Under the following parameters: $m = 31$, $n = 251$ and under a noise with $\sigma = 0.01$, we use the LGDAE to solve the above inverse problem to recover the initial temperature and two right-end boundary conditions as shown in Fig. 3, of which the maximum error and RMSE1 of initial temperatures are, respectively, 1.3×10^{-2} and 7.9×10^{-3} , the maximum error and RMSE2 of right-end temperatures are, respectively, 5.6×10^{-3} and 1.3×10^{-3} , and the maximum error and RMSE3 of right-end heat fluxes are 3.4×10^{-2} and 9.1×10^{-3} , respectively.

7.3 Example 3

Then we consider a nonlinear heat conduction equation:

$$u_t(x,t) = u_{xx}(x,t) - u^2(x,t) + (e^{-t} \cos x)^2, \quad 0 < x < 0.5, \quad 0 < t < 1, \quad (49)$$

where the exact solution is supposed to be

$$u(x,t) = e^{-t} \cos x. \quad (50)$$

Inserting Eq. (5) into Eq. (49) we can obtain

$$T_t(x,t) = T_{xx}(x,t) - T^2(x,t) - 2f(x)T(x,t) - f^2(x) + e^{-2t} \cos^2 x + F(x). \quad (51)$$

Now we can apply the LGDAE to solve the above equation to find $f(x)$ and $F(x)$ under the following parameters: $m = 21$, $n = 251$ and under a noise with $\sigma = 0.005$. The numerical errors are shown in Fig. 4, of which the maximum error and RMSE1 of initial temperatures are, respectively, 5.2×10^{-2} and 2.4×10^{-2} , the maximum error and RMSE2 of right-end temperatures are, respectively, 6.5×10^{-3} and 3×10^{-3} , and the maximum error and RMSE3 of right-end heat fluxes are 6×10^{-2} and 1.7×10^{-2} , respectively.

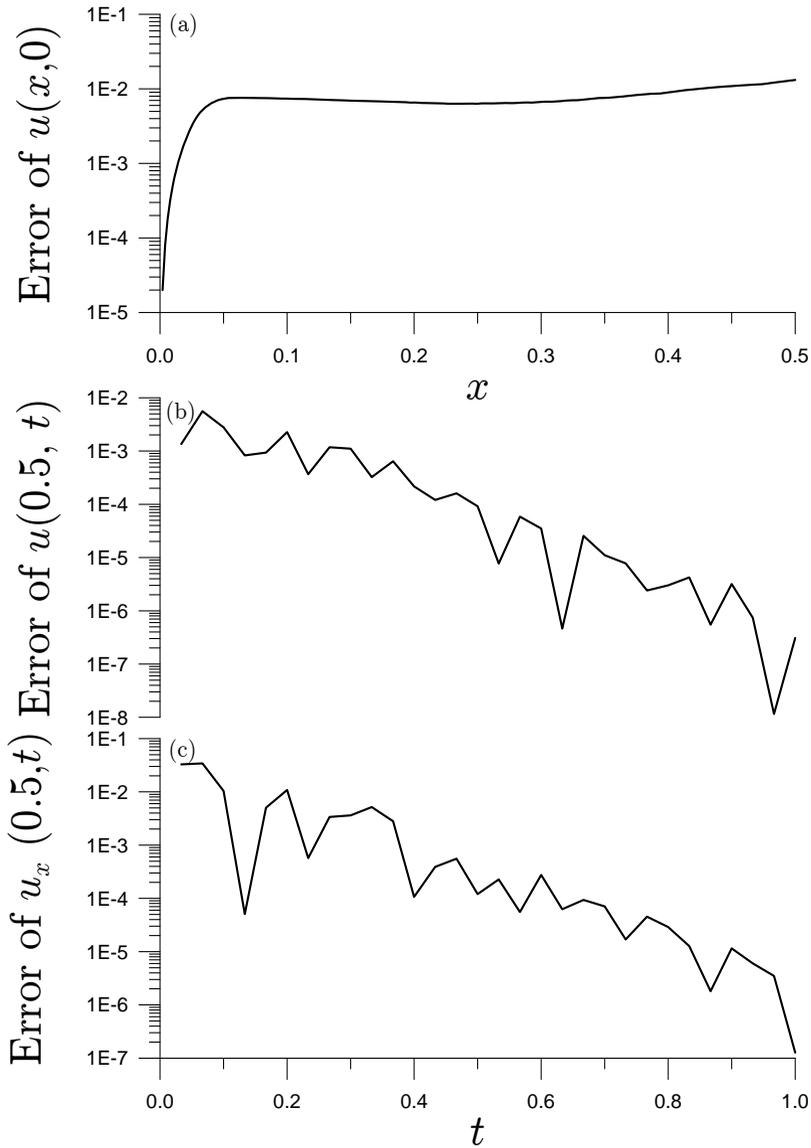


Figure 3: For example 2 showing the numerical errors of (a) initial temperature, (b) right-end temperature, and (c) right-end heat flux.

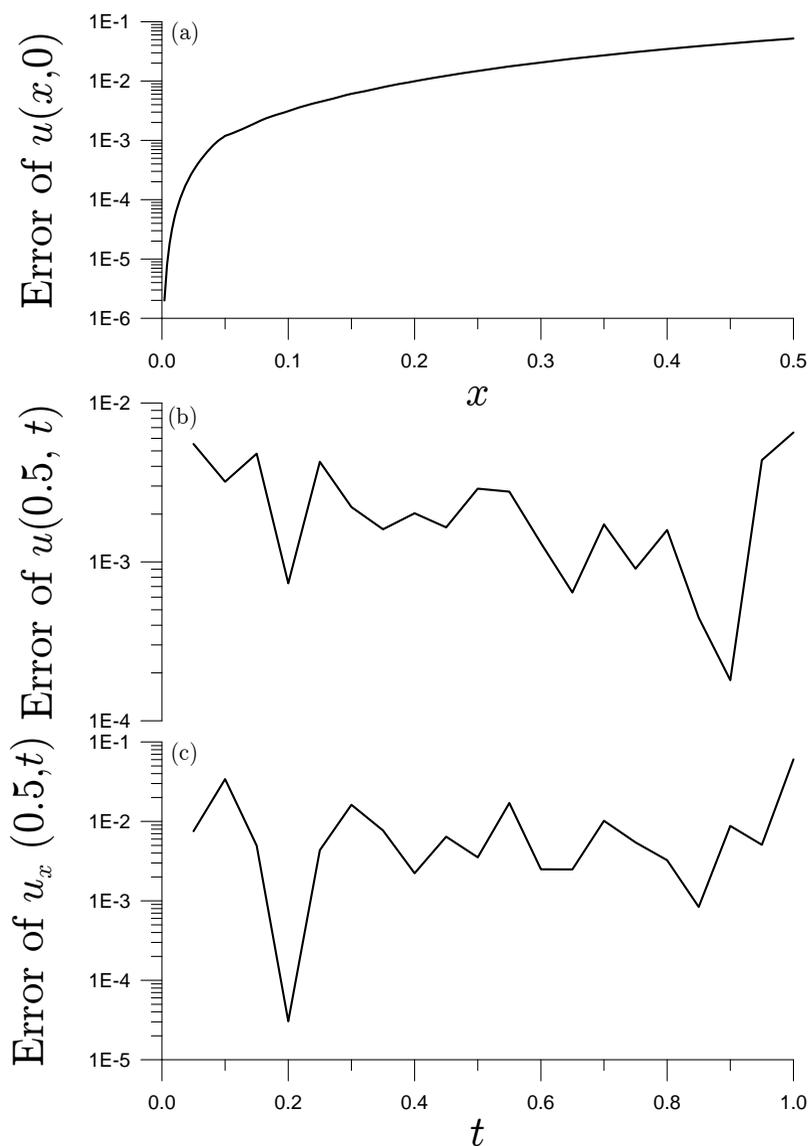


Figure 4: For nonlinear example 3 showing the numerical errors of (a) initial temperature, (b) right-end temperature, and (c) right-end heat flux.

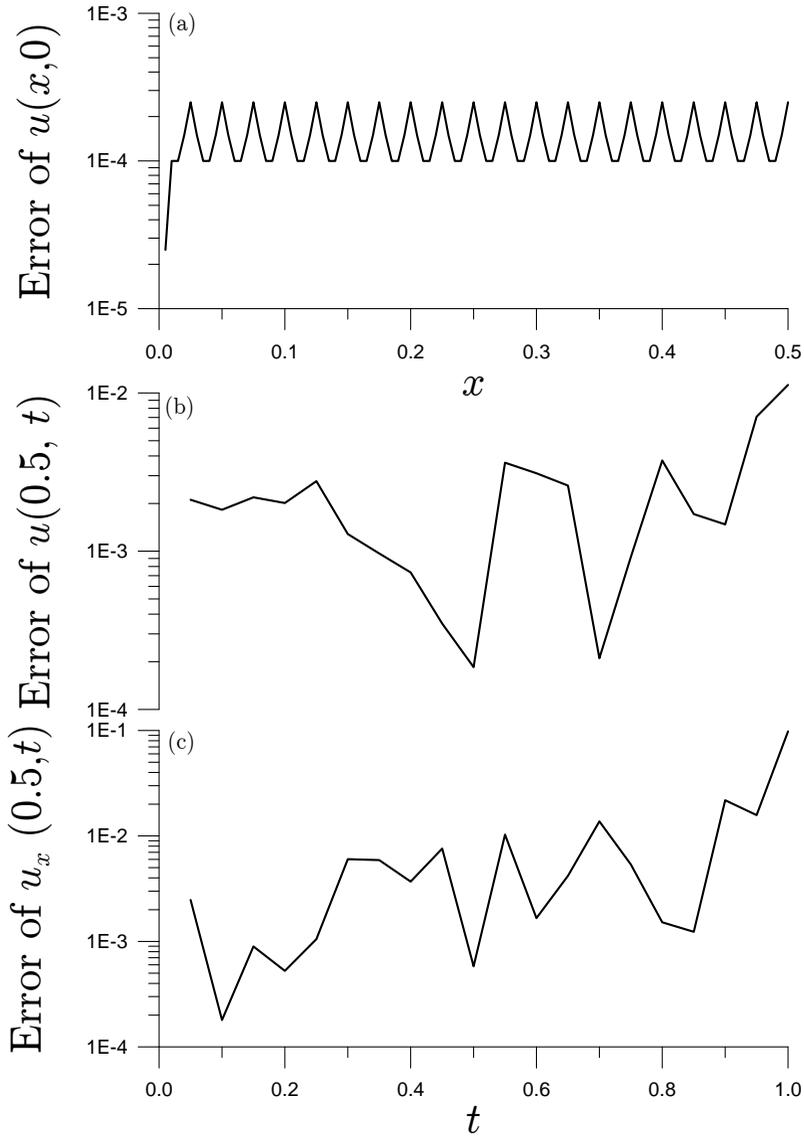


Figure 5: For example 4 of Burgers equation showing the numerical errors of (a) initial temperature, (b) right-end temperature, and (c) right-end heat flux.

7.4 Example 4

Finally, we consider a nonlinear Burgers equation:

$$u_t(x,t) = u_{xx}(x,t) - u(x,t)u_x(x,t) + 2x^3 + 4xt, \quad 0 < x < 0.5, \quad 0 < t < 1, \quad (52)$$

where the exact solution is supposed to be

$$u(x,t) = x^2 + 2t. \quad (53)$$

Inserting Eq. (5) into Eq. (52) we can obtain

$$T_t(x,t) = T_{xx}(x,t) - [T(x,t) + f(x)]T_x(x,t) - f'(x)[T(x,t) + f(x)] + 2x^3 + 4xt + F(x). \quad (54)$$

When we apply the LGDAE to solve the above inverse problem under the following parameters: $m = 21$, $n = 101$ and under a noise with $\sigma = 0.002$, the numerical errors are shown in Fig. 5, of which the maximum error and RMSE1 of initial temperatures are, respectively, 2.5×10^{-4} and 1.6×10^{-4} , the maximum error and RMSE2 of right-end temperatures are, respectively, 1.1×10^{-2} and 3.5×10^{-3} , and the maximum error and RMSE3 of right-end heat fluxes are, respectively, 9.8×10^{-2} and 2.3×10^{-2} .

8 Conclusions

There are very few methods that can solve the nonlinear sideways heat conduction problem without initial condition, which is a highly ill-posed inverse heat conduction problem. The existing methods in the literature are most restricted to solve linear Cauchy problems. We have transformed the nonlinear Cauchy problem without initial condition into a nonlinear heat source identification problem with a zero initial condition. Then, we have explained that the discretized version after using the numerical method of lines is a set of nonlinear differential algebraic equations, for which we have used the newly developed LGDAE to solve the unknown initial temperature and recover boundary temperature and heat flux on the right-end. Although under a large noisy disturbance on the Cauchy data, the accuracy and efficiency of numerical solutions were confirmed by comparing the recovered results with exact solutions.

Acknowledgement: Taiwan's National Science Council project NSC-102-2221-E-002-125-MY3 and the 2011 Outstanding Research Award, granted to the author, are highly appreciated.

References

- Berntsson, F.** (2003): Sequential solution of the sideways heat equation by windowing of the data. *Inv. Prob. Sci. Eng.*, vol. 11, pp. 91-103.
- Chang, C. W.; Liu, C.-S.; Chang, J. R.** (2005): A group preserving scheme for inverse heat conduction problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 10, pp. 13-38.
- Chi, C. C.; Yeih, W.; Liu, C.-S.** (2009): A novel method for solving the Cauchy problem of Laplace equation using the fictitious time integration method. *CMES: Computer Modeling in Engineering & Sciences*, vol. 47, pp. 167-190.
- Dorroh, J. R.; Ru, X.** (1999): The application of the method of quasi-reversibility to the sideways heat equation. *J. Math. Anal. Appl.*, vol. 236, pp. 503-519.
- Eldén, L.** (1987): Approximations for a Cauchy problem for the heat equation. *Inverse Problem*, vol. 3, pp. 263-273.
- Eldén, L.; Berntsson, F.; Reginska, T.** (2000): Wavelet and Fourier methods for solving the sideways heat equation. *SIAM J. Sci. Comput.*, vol. 21, pp. 2187-2205.
- Hao, D. N.; Reinhardt, H. J.; Schneider, A.** (2001): Numerical solution to a sideways parabolic equation. *Int. J. Numer. Meth. Engng.*, vol. 50, pp. 1253-1267.
- 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.** (2008a): A highly accurate MCTM for inverse Cauchy problems of Laplace equation in arbitrary plane domains. *CMES: Computer Modeling in Engineering & Sciences*, vol. 35, pp. 91-111.
- Liu, C.-S.** (2008b): A highly accurate MCTM for direct and inverse problems of biharmonic equation in arbitrary plane domains. *CMES: Computer Modeling in Engineering & Sciences*, vol. 30, pp. 65-75.
- Liu, C.-S.** (2011): A self-adaptive LGSM to recover initial condition or heat source of one-dimensional heat conduction equation by using only minimal boundary thermal data. *Int. J. Heat Mass Transfer*, vol. 54, pp. 1305-1312.
- Liu, C.-S.** (2013a): A method of Lie-symmetry $GL(n, \mathbb{R})$ for solving non-linear dynamical systems. *Int. J. Non-Linear Mech.*, vol. 52, pp. 85-95.
- Liu, C.-S.** (2013b): A state feedback controller used to solve an ill-posed linear system by a $GL(n, \mathbb{R})$ iterative algorithm. *Commu. Numer. Anal.*, vol. 2013, Article ID cna-00181, 22 pages.
- Liu, C.-S.** (2013c): Solving nonlinear differential algebraic equations by an implicit $GL(n, \mathbb{R})$ Lie-group method. *J. Appl. Math.*, vol. 2013, ID 987905, 8 pages.
- Liu, C.-S.** (2014a): An iterative method to recover heat conductivity function of

a nonlinear heat conduction equation. *Num. Heat Transfer, B: Fundamentals*, vol. 65, pp. 80-101.

Liu, C.-S. (2014b): An iterative algorithm for identifying heat source by using a DQ and a Lie-group method. *Inv. Prob. Sci. Eng.*, dx.doi.org/10.1080/17415977.2014.880907.

Liu, C.-S. (2014c): Lie-group differential algebraic equations method to recover heat source in a Cauchy problem with analytic continuation data. *Int. J. Heat Mass Transfer*, vol. 78, pp. 538-547.

Liu, C.-S. (2014d): A new sliding control strategy for nonlinear system solved by the Lie-group differential algebraic equation method. *Commun. Nonlinear Sci. Numer. Simulat.*, vol. 19, pp. 2012-2038.

Liu, C.-S.; Atluri, S. N. (2013): A $GL(n, \mathbb{R})$ differential algebraic equation method for numerical differentiation of noisy signal. *CMES: Computer Modeling in Engineering & Sciences*, vol. 92, pp. 213-239.

Liu, C.-S.; Kuo, C. L. (2011): A spring-damping regularization and a novel Lie-group integration method for nonlinear inverse Cauchy problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 77, pp. 57-80.

Liu, C.-S.; Kuo, C. L.; Liu D. (2011): The spring-damping regularization method and the Lie-group shooting method for inverse Cauchy problems. *CMC: Computers, Materials & Continua*, vol. 24, pp. 105-123.

Liu, J. C.; Wang, J. G. (2014): Cauchy problem for the heat equation in a bounded domain without initial value. *CMES: Computer Modeling in Engineering & Sciences*, vol. 97, pp. 437-462.

Liu, J. C.; Zhang, Q. G. (2013): Cauchy problem for the Laplace equation in 2D and 3D doubly connected domains. *CMES: Computer Modeling in Engineering & Sciences*, vol. 93, pp. 203-220.

Marin, L. (2009): An alternating iterative MFS algorithm for the Cauchy problem in two-dimensional anisotropic heat conduction. *CMC: Computers, Materials & Continua*, vol. 12, pp. 71-99.

Qian, Z.; Fu, C. L. (2007): Semi-discrete central difference method for determining surface heat flux of IHCP. *J. Korean Math. Soc.*, vol. 44, pp. 1397-1415.

Wang, Y.; Cheng, J.; Nakagawa, J.; Yamamoto, M. (2010): A numerical method for solving the inverse heat conduction problem without initial value. *Inv. Prob. Sci. Eng.*, vol. 18, pp. 655-671.

Yeih, W.; Liu, C.-S.; Kuo, C. L., Atluri, S. N. (2010): On solving the direct/inverse Cauchy problems of Laplace equation in a multiply connected domain, using the generalized multiple-source-point boundary-collocation Trefftz method

& characteristic lengths. *CMC: Computers, Materials & Continua*, vol. 17, pp. 275-302.

Yeih, W.; Chan, I. Y.; Fan, C. M.; Chang, J. J.; Liu, C.-S. (2014): Solving the Cauchy problem of the nonlinear steady-state heat equation using double iteration process. *CMES: Computer Modeling in Engineering & Sciences*, vol. 99, pp. 169-194.

