CMC, vol.18, no.1, pp.1-21, 2010

# A Lie-Group Adaptive Method for Imaging a Space-Dependent Rigidity Coefficient in an Inverse Scattering Problem of Wave Propagation

**Chein-Shan Liu**[1]

**Abstract:** We are concerned with the reconstruction of an unknown space-dependent rigidity coefficient in a wave equation. This problem is known as one of the inverse scattering problems. Based on a two-point Lie-group equation we develop a Lie-group adaptive method (LGAM) to solve this inverse scattering problem through iterations, which possesses a special character that by using only *two boundary conditions and two initial conditions*, as those used in the direct problem, we can effectively reconstruct the unknown rigidity function by a *self-adaption between the local in time differential governing equation and the global in time algebraic Lie-group equation*. The accuracy and efficiency of the present LGAM are assessed by comparing the imaged results with some postulated exact solutions. By means of LGAM, it is quite versatile to handle the wave inverse scattering problem for the image of the rigidity coefficient without needing any extra information from the wave motion.

**Keywords:** Inverse problem, Wave inverse scattering problem, Lie-group adaptive method (LGAM), Iterative method

## 1 Introduction

The parameter characterizing material property distributed in a solid medium may be quite complicated, which depends on direction and position. Inverse scattering is how we can obtain a large part of our information about the constitutents. We know about the interior structure of the earth by solving the inverse problem of determining the sound speed by measuring the travel times of seismic waves. Inverse scattering is also often-used in the non-destructive evaluation of structures to find cracks and corrosions. The task of inverse scattering theory is to determine material properties of the target, given sufficiently many input and output pairs. It

[1] Department of Civil Engineering, National Taiwan University, Taipei, Taiwan. E-mail: liucs@ntu.edu.tw

seeks to determine the profile of material property or an obstacle from its scattering amplitude [Majda (1976, 1977)], which may be a quite difficult research issue.

For the material property identification problem of wave equations there are two kinds of approach: time-independent approach and time-dependent approach. In fact, most often the inverse scattering problems are stated in a time-independent formulation that results after taking a Fourier transformation in the time variable, or inserting a time-harmonic plane wave into the wave equations.

To motivate the present study, we consider the longitudinal wave motion of a one-dimensional rod with variable Young's modulus $E(x)$ and cross-sectional area $A(x)$:

$$\frac{1}{A(x)}\frac{\partial}{\partial x}\left(E(x)A(x)\frac{\partial u(x,t)}{\partial x}\right)=\rho(x)\frac{\partial^2 u(x,t)}{\partial t^2}, \tag{1}$$

where $\rho(x)$ is the variable mass-density, and $u(x,t)$ is the longitudinal displacement.

Let $u(x,t)=e^{i\omega t}y(x)$, and Eq. (1) can be simplified as

$$-\frac{d}{dx}\left(E(x)A(x)\frac{dy(x)}{dx}\right)=\omega^2\rho(x)A(x)y(x), \tag{2}$$

where $\omega$ is the vibrational frequency. In the inverse scattering problem, it is technically important to identify the material properties $E(x)$, $\rho(x)$ and the geometric variable $A(x)$ for some measured frequencies $\omega$ of the vibrating rod. This problem is known as an inverse scattering problem of a vibrating rod for specified frequencies. Liu and Atluri (2008) have solved the above problem by using the Fictitious Time Integration Method (FTIM). Under the given eigenvalues they can recover the rigidity function $E(x)A(x)$.

Instead of identifying all the functions of $E(x)$, $A(x)$, and $\rho(x)$, we restrict ourselves to only identify $\alpha(x)=E(x)$ by supposing $\rho(x)=A(x)=1$, because $E(x)$ is the most important factor in the inverse wave scattering problem. The presently proposed method can solve this sort inverse problem by directly treating the following wave equation:

$$\frac{\partial^2 u(x,t)}{\partial t^2}=\frac{\partial}{\partial x}\left[\alpha(x)\frac{\partial u(x,t)}{\partial x}\right]+h(x,t),\ 0<x<\ell,\ 0<t<t_f, \tag{3}$$

$$u(x,0)=f(x), \tag{4}$$

$$u_t(x,0)=g(x), \tag{5}$$

$$u(0,t)=u_0(t), \tag{6}$$

$$u(\ell,t)=u_\ell(t), \tag{7}$$

where $h(x,t)$, $f(x)$, $g(x)$, $u_0(t)$ and $u_\ell(t)$ are given functions, and $\alpha(x) > 0$ is to be determined.

In the time-domain approach of the inverse scattering problems, there are some literature related to the present issue, to name a few, Baev (1986, 1987, 1988), Tadi (1997, 1998, 1999), and Na and Kallivokas (2009). For the above inverse problem, $\alpha(x)$ can be estimated, if one can provide an extra measurement of data as that given by an over-specified boundary condition at $x = \ell$:

$$\frac{\partial u(\ell,t)}{\partial x} = F_m(t), \tag{8}$$

or sometimes by an internal measurement of $u$ at a point $x_m$:

$$u(x_m,t) = u_m(t). \tag{9}$$

For the problem governed by Eqs. (3)-(8) or (9) there are some studies as can be seen from the papers by Tadi (1998), Na and Kallivokas (2009), and references therein.

Liu (2006a, 2006b, 2006c) has extended the group-preserving scheme (GPS) developed previously by Liu (2001) for ODEs to solve the boundary value problems (BVPs). In the construction of the Lie-group method for the calculations of BVPs, Liu (2006a) has introduced the idea of one-step GPS by utilizing the closure property of the Lie group, and hence, the resulting shooting method has been named the Lie-group shooting method (LGSM). After that, Liu (2006d) has used this concept to establish a one-step estimation method to estimate the temperature-dependent heat conductivity, and then extended the Lie-group method to estimate the thermophysical properties of heat conductivity and heat capacity [Liu (2006e, 2007)]. The Lie-group method possesses a great advantage than other numerical methods due to its group structure, and it is a powerful technique to solve the inverse problem of parameter identification [Liu (2008a, 2008b, 2009)]. Recently, Liu and Atluri (2010) have made a breakthrough by solving the Calderón inverse problem with a more general version of the Lie-group shooting method.

In this paper we introduce a new concept of the self-adaption by a two-point Lie-group equation, such that we can iteratively solve the inverse scattering problem in Eq. (3) by imaging $\alpha(x)$, which only using the data given in Eqs. (4)-(7). This paper is organized as follows. In Section 2 we give a semi-discretization of the governing equation by the numerical method of line. Section 3 is devoted to develop a Lie-group formulation of the inverse scattering problem, including a group-preserving scheme, a one-step Lie-group transformation, and a two-point Lie-group equation. The Lie-group method is described in Section 4. In Section 5, we adjust the Lie-group shooting method as being a Lie-group adaptive method suitable for the estimation of parameter without having a real target, and the numerical procedures are described. The numerical tests are carried out in Section 6 with five numerical examples. Finally, some significiant observations are drawn in Section 7.

## 2 The numerical method of line

First, let $v(x,t) = \partial u(x,t)/\partial t$, and then Eq. (3) in a state-space description can be expressed as

$$\frac{\partial u(x,t)}{\partial t} = v(x,t), \tag{10}$$

$$\frac{\partial v(x,t)}{\partial t} = \frac{\partial}{\partial x}\left[\alpha(x)\frac{\partial u(x,t)}{\partial x}\right] + h(x,t). \tag{11}$$

Second, we use a semi-discretization method to discretize the quantities of $u(x,t)$ and $v(x,t)$ in the space domain, and then we can obtain a system of ODEs for $u$ and $v$ with $t$ as an independent variable:

$$\dot{u}_i(t) = v_i(t), \;\; i = 1,\ldots,n, \tag{12}$$

$$\dot{v}_i(t) = \frac{1}{(\Delta x)^2}\left(\alpha_{i+1}[u_{i+1}(t) - u_i(t)] - \alpha_i[u_i(t) - u_{i-1}(t)]\right) + h_i(t), \;\; i = 1,\ldots,n, \tag{13}$$

where $\Delta x = \ell/(n+1)$ is a uniform spatial increment with $n$ the number of interior grid points, and $x_i = i\Delta x$ are the discretized coordinates of $x$, at which the displacement and velocity are, respectively, discretized as $u_i(t) = u(x_i,t)$ and $v_i(t) = v(x_i,t)$. Besides, $h_i(t) = h(x_i,t)$ and $\alpha_i = \alpha(x_i)$ are, respectively, the discretized quantities of $h(x,t)$ and $\alpha(x)$ at the spatial point $x_i$.

When $i = 1$ in Eq. (13), the term $u_0(t)$ appeared there is determined by the boundary condition in Eq. (6). Similarly, when $i = n$, the term $u_{n+1}(t) = u_\ell(t)$ is determined by the boundary condition in Eq. (7). On the other hand, the term $\alpha_{n+1}$ is supposed to be measurable at the right-boundary.

The two initial conditions are given by

$$u_i(0) = f(x_i), \;\; i = 1,\ldots,n, \tag{14}$$

$$v_i(0) = g(x_i), \;\; i = 1,\ldots,n, \tag{15}$$

which are obtained from Eqs. (4) and (5) by discretizations.

## 3 A Lie-group formulation

In order to explore our new method in a self-contained fashion, let us first briefly sketch the group-preserving scheme (GPS) for ODEs and one-step GPS in this section.

### 3.1 The group-preserving scheme

We write Eqs. (12) and (13) in a vector form:

$$\dot{\mathbf{y}} = \mathbf{f}(t, \mathbf{y}), \tag{16}$$

where the dot denotes the differential with respect to $t$, and

$$\mathbf{y} := \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{bmatrix} := \begin{bmatrix} \mathbf{v} \\ \mathbf{f}_2(t, \mathbf{u}) \end{bmatrix}, \tag{17}$$

in which $\mathbf{u} = (u_1, \ldots, u_n)^{\mathrm{t}}$ and $\mathbf{v} = (v_1, \ldots, v_n)^{\mathrm{t}}$ with the superscript $^{\mathrm{t}}$ for the transpose. The components of $\mathbf{f}_2$ represent the right-hand side of Eq. (13).

When both the vector $\mathbf{y}$ and its magnitude $\|\mathbf{y}\| := \sqrt{\mathbf{y}^{\mathrm{t}}\mathbf{y}} = \sqrt{\mathbf{y} \cdot \mathbf{y}}$ are combined into a single augmented vector

$$\mathbf{X} = \begin{bmatrix} \mathbf{y} \\ \|\mathbf{y}\| \end{bmatrix}, \tag{18}$$

Liu (2001) has transformed Eq. (16) into an augmented system:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} := \begin{bmatrix} \mathbf{0}_{2n \times 2n} & \frac{\mathbf{f}(t,\mathbf{y})}{\|\mathbf{y}\|} \\ \frac{\mathbf{f}^{\mathrm{t}}(t,\mathbf{y})}{\|\mathbf{y}\|} & 0 \end{bmatrix} \mathbf{X}, \tag{19}$$

where $\mathbf{A}$ is an element of the Lie-algebra $so(2n, 1)$ satisfying

$$\mathbf{A}^{\mathrm{t}}\mathbf{g} + \mathbf{g}\mathbf{A} = \mathbf{0}, \tag{20}$$

and

$$\mathbf{g} = \begin{bmatrix} \mathbf{I}_{2n} & \mathbf{0}_{2n \times 1} \\ \mathbf{0}_{1 \times 2n} & -1 \end{bmatrix} \tag{21}$$

is a Minkowski metric. Here, $\mathbf{I}_{2n}$ is the $2n$-order identity matrix.

The augmented variable $\mathbf{X}$ can be viewed as a point in the Minkowski space $\mathbb{M}^{2n+1}$, satisfying the cone condition:

$$\mathbf{X}^{\mathrm{t}}\mathbf{g}\mathbf{X} = \mathbf{y} \cdot \mathbf{y} - \|\mathbf{y}\|^2 = 0. \tag{22}$$

Accordingly, Liu (2001) has developed a group-preserving scheme (GPS) to guarantee that each $\mathbf{X}_k$ automatically locates on the cone:

$$\mathbf{X}_{k+1} = \mathbf{G}(k)\mathbf{X}_k, \tag{23}$$

where $\mathbf{X}_k$ denotes the numerical value of $\mathbf{X}$ at a discrete time $t_k$, and $\mathbf{G}(k) \in SO_o(2n,1)$ satisfies

$$\mathbf{G}^{\mathrm{t}}\mathbf{g}\mathbf{G} = \mathbf{g}, \tag{24}$$

$$\det \mathbf{G} = 1, \tag{25}$$

$$G_0^0 > 0, \tag{26}$$

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

### 3.2 One-step Lie-group transformation

Throughout this paper we use the superscripted symbol $\mathbf{y}^0$ to denote the value of $\mathbf{y}$ at $t = 0$, and $\mathbf{y}^f$ the value of $\mathbf{y}$ at $t = t_f$.

Applying the scheme in Eq. (23) to Eq. (19) with an initial condition $\mathbf{X}(0) = \mathbf{X}^0$ we can compute $\mathbf{X}(t)$ by the GPS. Assuming that the time stepsize used in the GPS is $\Delta t = t_f/K$, we can calculate the value of $\mathbf{X}^f = ((\mathbf{y}^f)^{\mathrm{t}}, \|\mathbf{y}^f\|)^{\mathrm{t}}$ at a final time $t = t_f$, by applying the scheme in Eq. (23) to Eq. (19) step-by-step:

$$\mathbf{X}^f = \mathbf{G}_K \cdots \mathbf{G}_1 \mathbf{X}^0. \tag{27}$$

Because each $\mathbf{G}_i$, $i = 1,\ldots,K$, is an element of the Lie-group $SO_o(2n,1)$, and by the closure property of the Lie group, $\mathbf{G}_K \cdots \mathbf{G}_1$ is also a Lie-group element denoted by $\mathbf{G}$. Hence, from Eq. (27) it follows that

$$\mathbf{X}^f = \mathbf{G}\mathbf{X}^0, \tag{28}$$

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

Now the problem is how to calculate $\mathbf{G}$. While an exact solution of $\mathbf{G}$ is not available, we can calculate $\mathbf{G}$ through a numerical method 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. The Lie-group element generated from such an $\mathbf{A} \in so(2n,1)$ by an exponential mapping is

$$\mathbf{G}(r) = \begin{bmatrix} \mathbf{I}_{2n} + \frac{a-1}{\|\hat{\mathbf{f}}\|^2}\hat{\mathbf{f}}\hat{\mathbf{f}}^{\mathrm{t}} & \frac{b\hat{\mathbf{f}}}{\|\hat{\mathbf{f}}\|} \\ \frac{b\hat{\mathbf{f}}^{\mathrm{t}}}{\|\hat{\mathbf{f}}\|} & a \end{bmatrix}, \tag{29}$$

where

$$\hat{\mathbf{y}} = r\mathbf{y}^0 + (1-r)\mathbf{y}^f, \tag{30}$$

$$\hat{\mathbf{f}} = \mathbf{f}(\hat{t}, \hat{\mathbf{y}}), \tag{31}$$

$$a = \cosh\left(\frac{t_f\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{y}}\|}\right), \quad b = \sinh\left(\frac{t_f\|\hat{\mathbf{f}}\|}{\|\hat{\mathbf{y}}\|}\right). \tag{32}$$

Here, we have derived a single-parameter Lie-group element $\mathbf{G}(r)$ in terms of $r \in [0,1]$, and $\hat{t} = (1-r)t_f$.

### 3.3 A two-point Lie-group equation

Upon defining

$$\mathbf{F} := \frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{y}}\|}, \tag{33}$$

Eqs. (29) and (32) can be expressed as

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_{2n} + \frac{a-1}{\|\mathbf{F}\|^2}\mathbf{F}\mathbf{F}^{\mathrm{t}} & \frac{b\mathbf{F}}{\|\mathbf{F}\|} \\ \frac{b\mathbf{F}^{\mathrm{t}}}{\|\mathbf{F}\|} & a \end{bmatrix}, \tag{34}$$

$$a = \cosh(t_f\|\mathbf{F}\|), \;\; b = \sinh(t_f\|\mathbf{F}\|). \tag{35}$$

From Eqs. (18), (28) and (34) it follows that

$$\mathbf{y}^f = \mathbf{y}^0 + \eta\mathbf{F}, \tag{36}$$

$$\|\mathbf{y}^f\| = a\|\mathbf{y}^0\| + b\frac{\mathbf{F}\cdot\mathbf{y}^0}{\|\mathbf{F}\|}, \tag{37}$$

where

$$\eta := \frac{(a-1)\mathbf{F}\cdot\mathbf{y}^0 + b\|\mathbf{y}^0\|\,\|\mathbf{F}\|}{\|\mathbf{F}\|^2}. \tag{38}$$

Eq. (36) is written as

$$\mathbf{F} = \frac{1}{\eta}(\mathbf{y}^f - \mathbf{y}^0). \tag{39}$$

Substituting $\mathbf{F}$ into Eq. (37) and dividing both the sides by $\|\mathbf{y}^0\|$, we can obtain

$$\frac{\|\mathbf{y}^f\|}{\|\mathbf{y}^0\|} = a + b\frac{(\mathbf{y}^f - \mathbf{y}^0)\cdot\mathbf{y}^0}{\|\mathbf{y}^f - \mathbf{y}^0\|\|\mathbf{y}^0\|}, \tag{40}$$

where, after inserting Eq. (39) for $\mathbf{F}$ into Eq. (35), $a$ and $b$ are now written as

$$a = \cosh\left(\frac{t_f\|\mathbf{y}^f - \mathbf{y}^0\|}{\eta}\right), \;\; b = \sinh\left(\frac{t_f\|\mathbf{y}^f - \mathbf{y}^0\|}{\eta}\right). \tag{41}$$

Let

$$\cos\theta := \frac{(\mathbf{y}^f - \mathbf{y}^0)\cdot \mathbf{y}^0}{\|\mathbf{y}^f - \mathbf{y}^0\|\|\mathbf{y}^0\|}, \tag{42}$$

$$S := t_f\|\mathbf{y}^f - \mathbf{y}^0\|, \tag{43}$$

and thus from Eqs. (40) and (41) it follows that

$$\frac{\|\mathbf{y}^f\|}{\|\mathbf{y}^0\|} = \cosh\left(\frac{S}{\eta}\right) + \cos\theta\sinh\left(\frac{S}{\eta}\right). \tag{44}$$

Upon defining

$$Z := \exp\left(\frac{S}{\eta}\right), \tag{45}$$

we can derive [Liu (2008b, 2010)]

$$Z = \frac{(\cos\theta - 1)\|\mathbf{y}^0\|}{\cos\theta\|\mathbf{y}^0\| + \|\mathbf{y}^f - \mathbf{y}^0\| - \|\mathbf{y}^f\|}, \tag{46}$$

and from Eqs. (45) and (43) it follows that

$$\eta = \frac{t_f\|\mathbf{y}^f - \mathbf{y}^0\|}{\ln Z}. \tag{47}$$

Therefore, we arrive to an important result that between any two points $(\mathbf{y}^0, \|\mathbf{y}^0\|)$ and $(\mathbf{y}^f, \|\mathbf{y}^f\|)$ on the cone, there exists a Lie-group element $\mathbf{G} \in SO_o(2n,1)$ mapping $(\mathbf{y}^0, \|\mathbf{y}^0\|)$ onto $(\mathbf{y}^f, \|\mathbf{y}^f\|)$, which is given by

$$\begin{bmatrix} \mathbf{y}^f \\ \|\mathbf{y}^f\| \end{bmatrix} = \mathbf{G}(t_f)\begin{bmatrix} \mathbf{y}^0 \\ \|\mathbf{y}^0\| \end{bmatrix}, \tag{48}$$

$$\mathbf{G}(t_f) = \begin{bmatrix} \mathbf{I}_{2n} + \frac{a-1}{\|\mathbf{F}\|^2}\mathbf{F}\mathbf{F}^{\mathrm{t}} & \frac{b\mathbf{F}}{\|\mathbf{F}\|} \\ \frac{b\mathbf{F}^{\mathrm{t}}}{\|\mathbf{F}\|} & a \end{bmatrix}, \tag{49}$$

$$a = \cosh(t_f\|\mathbf{F}\|), \;\; b = \sinh(t_f\|\mathbf{F}\|), \tag{50}$$

$$\mathbf{F} = \frac{1}{\eta}(\mathbf{y}^f - \mathbf{y}^0) = \frac{\ln Z}{t_f}\frac{\mathbf{y}^f - \mathbf{y}^0}{\|\mathbf{y}^f - \mathbf{y}^0\|}. \tag{51}$$

It should be emphasized that the above $\mathbf{G}(t_f)$ is different from the $\mathbf{G}(r)$ in Eq. (29). In order to extrude it being a Lie-group mapping between the quantities spanned a whole time interval of $[0,t_f]$ we write it to be $\mathbf{G}(t_f)$, which is independent on $\mathbf{f}$ and

$r$. In contrast, $\mathbf{G}(r)$ is a function of $r$ and $\mathbf{f}$. However, these two Lie-group elements $\mathbf{G}(r)$ and $\mathbf{G}(t_f)$ are both indispensable in our development of the Lie-group method, which is coined as the following *Lie-group equation*:

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

$$\frac{\mathbf{y}^f - \mathbf{y}^0}{\|\mathbf{y}^f - \mathbf{y}^0\|} = \frac{t_f}{\ln Z}\frac{\hat{\mathbf{f}}}{\|\hat{\mathbf{y}}\|}, \tag{53}$$

by equating the two $\mathbf{F}$'s in Eqs. (51) and (33), i.e., $\mathbf{G}(t_f) = \mathbf{G}(r)$. When $t_f$ is a physical time, $\ln Z$ can be viewed as a geometrical time of the ODEs system.

Corresponding to the *local in time differential equation* (16), the above is a *global in time algebraic equation*, defined at two points $t = 0$ and $t = t_f$, and it is a two-point Lie-group equation. Previously, Liu (2006a) has derived this equation, but did not write it explicitly, and originally this equation was used to solve the two-point boundary value problem; later, Liu (2008a, 2008b, 2009, 2010) has called this equation a *Lie-group shooting equation* and employed it to solve many inverse problems. At there the situation is that this equation was used for the inverse problems which have a real target to be shot. Presently, we release this constraint with a new concept that the above equation is just a two-point Lie-group equation describing a nonlinear relation between these two quantities of $\mathbf{y}^0$ and $\mathbf{y}^f$ defined at two different times $t = 0$ and $t = t_f$. This equation is indeed inherent in all ODEs, no matter there is a target or does not have a target in the ODEs. In the next section for the inverse problem of recovering an unknown coefficient without the help from a real target, i.e., extra data measurement, like as that in Eq. (8) or Eq. (9), we will employ the above Lie-group equation to derive algebriac equations system to solve $\alpha(x)$.

## 4 The Lie-group method

From Eqs. (12)-(15) it follows that

$$\dot{\mathbf{u}} = \mathbf{v}, \tag{54}$$

$$\dot{\mathbf{v}} = \mathbf{f}_2(t, \mathbf{u}), \tag{55}$$

$$\mathbf{u}(0) = \mathbf{u}^0, \ \ \mathbf{u}(t_f) = \mathbf{u}^f, \tag{56}$$

$$\mathbf{v}(0) = \mathbf{v}^0, \ \ \mathbf{v}(t_f) = \mathbf{v}^f, \tag{57}$$

where $\mathbf{u}^0$ and $\mathbf{v}^0$ are known from Eqs. (14) and (15).

By using Eq. (17) for $\mathbf{y}$ we have

$$\mathbf{y}^0 = \begin{bmatrix} \mathbf{u}^0 \\ \mathbf{v}^0 \end{bmatrix}, \ \mathbf{y}^f = \begin{bmatrix} \mathbf{u}^f \\ \mathbf{v}^f \end{bmatrix}, \tag{58}$$

and by Eq. (52) we can obtain

$$\mathbf{u}^f = \mathbf{u}^0 + \frac{\eta}{\|\hat{\mathbf{y}}\|}\hat{\mathbf{v}}, \tag{59}$$

$$\mathbf{v}^f = \mathbf{v}^0 + \frac{\eta}{\|\hat{\mathbf{y}}\|}\hat{\mathbf{f}}_2, \tag{60}$$

where

$$\begin{aligned}\|\hat{\mathbf{y}}\| &= \sqrt{\|\hat{\mathbf{u}}\|^2 + \|\hat{\mathbf{v}}\|^2} \\ &= \sqrt{\|r\mathbf{u}^0 + (1-r)\mathbf{u}^f\|^2 + \|r\mathbf{v}^0 + (1-r)\mathbf{v}^f\|^2},\end{aligned} \tag{61}$$

$$\hat{\mathbf{f}}_2 = \begin{bmatrix} \frac{1}{(\Delta x)^2}[\alpha_2(\hat{u}_2 - \hat{u}_1) - \alpha_1(\hat{u}_1 - \hat{u}_0)] + \hat{h}_1 \\ \frac{1}{(\Delta x)^2}[\alpha_3(\hat{u}_3 - \hat{u}_2) - \alpha_2(\hat{u}_2 - \hat{u}_1)] + \hat{h}_2 \\ \vdots \\ \frac{1}{(\Delta x)^2}[\alpha_n(\hat{u}_n - \hat{u}_{n-1}) - \alpha_{n-1}(\hat{u}_{n-1} - \hat{u}_{n-2})] + \hat{h}_{n-1} \\ \frac{1}{(\Delta x)^2}[\alpha_{n+1}(\hat{u}_{n+1} - \hat{u}_n) - \alpha_n(\hat{u}_n - \hat{u}_{n-1})] + \hat{h}_n \end{bmatrix}, \tag{62}$$

where $\hat{u}_i = ru_i^0 + (1-r)u_i^f = rf(x_i) + (1-r)u_i^f$, $\hat{h}_i = h_i(\hat{t})$, and $\hat{u}_0 = u_0(\hat{t})$ and $\hat{u}_{n+1} = u_\ell(\hat{t})$.

From Eqs. (60) and (62) we can obtain a closed-form formula to calculate $\alpha_i$:

$$\alpha_i = \frac{(\Delta x)^2}{\hat{u}_i - \hat{u}_{i-1}}\left[\frac{\hat{u}_{i+1} - \hat{u}_i}{(\Delta x)^2}\alpha_{i+1} + \hat{h}_i - \frac{\|\hat{\mathbf{y}}\|}{\eta}(v_i^f - v_i^0)\right]. \tag{63}$$

Here, $\alpha_{n+1}$ is the right-boundary value of $\alpha$, which is supposed to be measurable. Because $\eta$ is a nonlinear function of $u_i^f$ and $v_i^f$, Eq. (63) provides us a mathematical tool to calculate $\alpha_i$ through iterations.

## 5 A Lie-group adaptive method to compute $\alpha(x)$

Now, the numerical procedures for estimating $\alpha_i$ are described as follows. We assume an initial value of $\alpha_i$, for example, $\alpha_i = 1$. Substituting it into Eqs. (12) and (13) we can apply the GPS to integrate them from $t = 0$ to $t = t_f$. Here $t_f$ is a parameter chosen by the user. Then, we can obtain $u_i^f$ and $v_i^f$, and inserting them into Eq. (63) by fixing $r = 1$ we can calculate a new $\alpha_i$, which is then compared with the old $\alpha_i$. If the difference of these two sets of $\alpha_i$ is smaller than a given criterion, then we stop the iteration and the final $\alpha_i$ is obtained. The numerical processes are summarized as follows:

**Step 1:** Give an initial $\alpha_i = 1$.

**Step 2:** For $j = 1, 2 \ldots$ we repeat the following calculations. Calculate $u_i^f$ and $v_i^f$ by using the GPS to integrate Eqs. (12) and (13) from $t = 0$ to $t = t_f$.

**Step 3:** Insert the above calculated $u_i^f$ and $v_i^f$, denoted respectively by $u_i^f(j)$ and $v_i^f(j)$, together with $u_i^0$ and $v_i^0$ given by Eqs. (14) and (15) into

$$\alpha_i(j) = \frac{(\Delta x)^2}{\hat{u}_i(j) - \hat{u}_{i-1}(j)} \left[ \frac{\hat{u}_{i+1}(j) - \hat{u}_i(j)}{(\Delta x)^2} \alpha_{i+1}(j) + \hat{h}_i - \frac{\|\hat{\mathbf{y}}(j)\|}{\eta(j)} \{v_i^f(j) - v_i^0\} \right], \tag{64}$$

where $\eta(j)$ and $\|\hat{\mathbf{y}}(j)\|$ are calculated from Eqs. (47) and (61) by inserting $u_i^f(j)$, $v_i^f(j)$, $u_i^0$ and $v_i^0$. If $\alpha_i(j)$ converges by satisfying a given convergence criterion:

$$C_j =: \sqrt{\frac{1}{n} \sum_{i=1}^{n} [\alpha_i(j+1) - \alpha_i(j)]^2} < \varepsilon, \tag{65}$$

then stop; otherwise, go to **Step 2**. $C_j$ measures the convergence speed.

Basically, the present method is used the two-point Lie-group equation (52) to derive Eq. (64), where the values of $u_i^f(j)$ and $v_i^f(j)$ at a time $t_f$ are obtained by repeatedly using the time-direction integrator GPS for Eqs. (12) and (13), and then we can adjust $\alpha_i$ by Eq. (64). The final time data $u_i^f(j)$ and $v_i^f(j)$ are not obtained through measurements. Because of the iteration processes as being a combination of the GPS and the Lie-group equation, the present algorithm is quite different from other algorithms. It can be seen that the present algorithm is simple, straightforward, and easy to numerical implementation. In order to distinct the present method from the earlier ones, we may call it a *Lie-group adaptive method* (LGAM), where the adaptions are performed by the governing equations themselves. The rationale of this algorithm is that the *local in time differential equation* (13) and the *global in time algebraic equation* (63) must self-adapt to a situation that they are compatible, such that $\alpha_i$ can be computed from them through a self-adaption in the iteration process.

## 6 Numerical investigations

### 6.1 Example 1

Let us first use the following example to demonstrate the above process. This example is given by

$$\alpha(x) = (x-3)^2, \;\; x \in (0,1), \tag{66}$$

$$h(x,t) = -5(x-3)^2 e^{-t}, \tag{67}$$

with the boundary conditions

$$u(0,t) = 9e^{-t}, \;\; u(1,t) = 4e^{-t}, \tag{68}$$

and the initial conditions

$$u(x,0) = (x-3)^2, \;\; v(x,0) = -(x-3)^2 \tag{69}$$

as the only inputs in the new algorithm of LGAM.

The exact solution of $u$ is given by

$$u(x,t) = (x-3)^2 e^{-t}. \tag{70}$$

Besides the conditions given in Eqs. (68) and (69), we do not need the data of $u$ inside the domain of $\Omega := \{(x,t) \mid 0 < x < 1, \; 0 < t \le t_f\}$. However, we write it explicitly for that we can conveniently derive all the required boundary conditions and initial conditions from it.

Now we apply the LGAM to this problem of the identification of $\alpha(x)$, where we have fixed $\Delta x = 1/40$, $\Delta t = 0.2/500$, and $t_f = 0.2$. Under the stopping criterion with $\varepsilon = 10^{-5}$, the numerical process is convergent within 30 iterations. In Fig. 1(a) we show the rate of convergence, which is an exponential convergence. In Fig. 1(b), we plot the tentative $\alpha_i$ for the first iteration, the fifth iteration, and the ninth iteration, the last of which is already very close to the exact solution. As shown in Fig. 1(c) the numerically obtained $\alpha_i$ is almost coincident with the exact one, with the Root-Mean-Squared-Error (RMSE) about 0.0248, and the maximal relative error about $4.6 \times 10^{-3}$.

### 6.2 Example 2

In this example we let $h(x,t) = 0$ by giving

$$\alpha(x) = 1 + e^{-x}, \;\; x \in (0,1), \tag{71}$$

$$u(x,t) = \exp(x+t). \tag{72}$$

The required conditions can be derived from Eq. (72) readily.

We apply the LGAM to this problem by using $\Delta x = 1/30$, $\Delta t = 0.2/100$, and $t_f = 0.2$. Under the stopping criterion with $\varepsilon = 10^{-4}$, the numerical process is convergent within 13 iterations. In Fig. 2(a) we show the rate of convergence, which is an exponential convergence. In Fig. 2(b), we compare the numerical solution with the exact solution, which are almost coincident. The numerical solution of $\alpha_i$ is very close to the exact one with the RMSE about $7.12 \times 10^{-4}$, and the maximal relative error about $8.02 \times 10^{-4}$ as shown in Fig. 2(c).

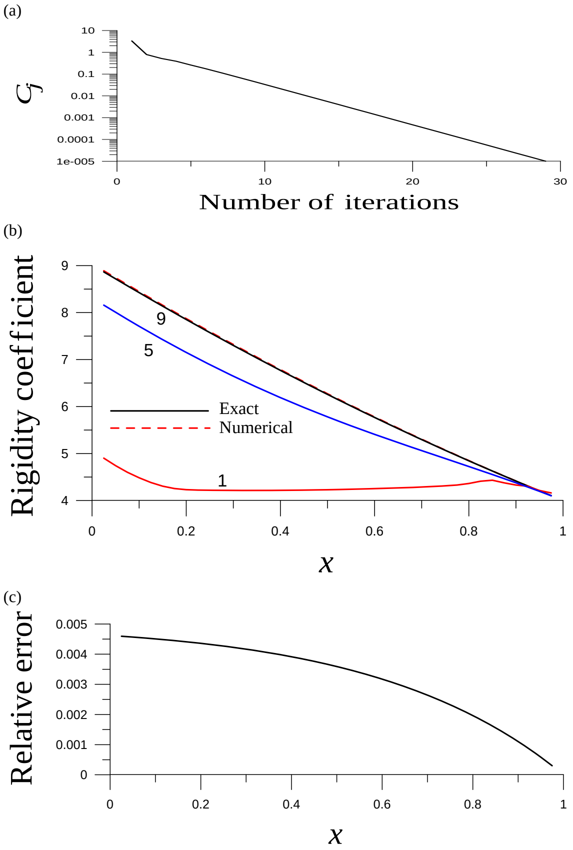

Figure 1: For example 1: (a) the convergence rate, (b) comparing numerical and exact solutions of rigidity coefficient, and (c) showing the relative error.

### *6.3 Example 3*

The identified function $\alpha(x)$ is oscillatory, given as follows:

$$\alpha(x) = 1+\sin(3\pi x), \quad x \in (0,1), \tag{73}$$

and the function $h(x,t)$ is calculated from

$$h(x,t) = \frac{\partial^2 u(x,t)}{\partial t^2} - \frac{\partial}{\partial x}\left[\alpha(x)\frac{\partial u(x,t)}{\partial x}\right], \tag{74}$$

by inserting Eq. (72) for $u(x,t)$.

In this identification of $\alpha(x)$ we use $\Delta x = 1/80$, $\Delta t = 0.01/50$, and $t_f = 0.01$. Fig. 3 shows that the numerical solution marked by the dashed line is close to the solid

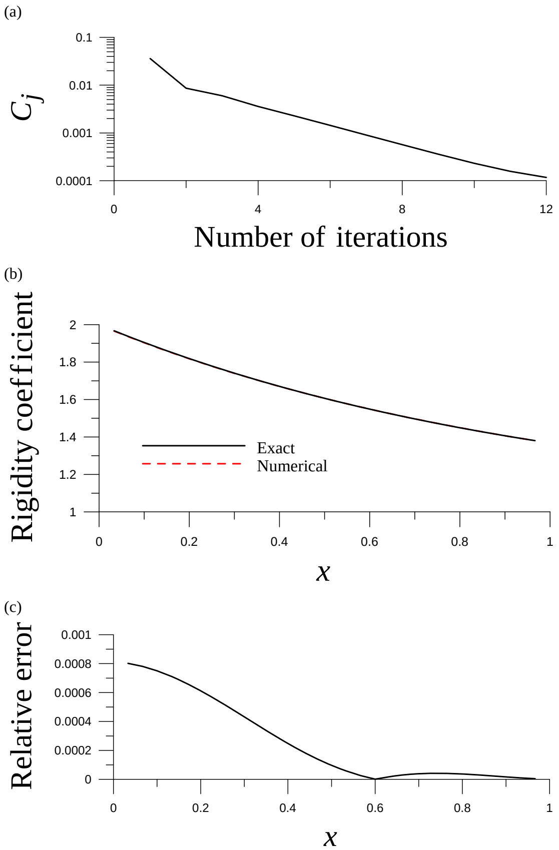

Figure 2: For example 2: (a) the convergence rate, (b) comparing numerical and exact solutions of rigidity coefficient, and (c) showing the relative error.

line of the exact solution, with the RMSE about $3.48 \times 10^{-2}$, and the maximal absolute error about $5.02 \times 10^{-2}$.

### 6.4 Example 4

The identified function $\alpha(x)$ is a one-hump function:

$$\alpha(x) = 2 + \exp\left(-\frac{(x-0.5)^2}{0.05}\right), \quad x \in (0,1), \tag{75}$$

and the function $h(x,t)$ is calculated from Eq. (74) by inserting Eq. (70) for $u(x,t)$. In this identification of $\alpha(x)$ we have applied the LGAM with $\Delta x = 1/100$, $\Delta t = 0.05/50$, and $t_f = 0.05$. Fig. 4(a) shows the convergence rate under a stopping cri-

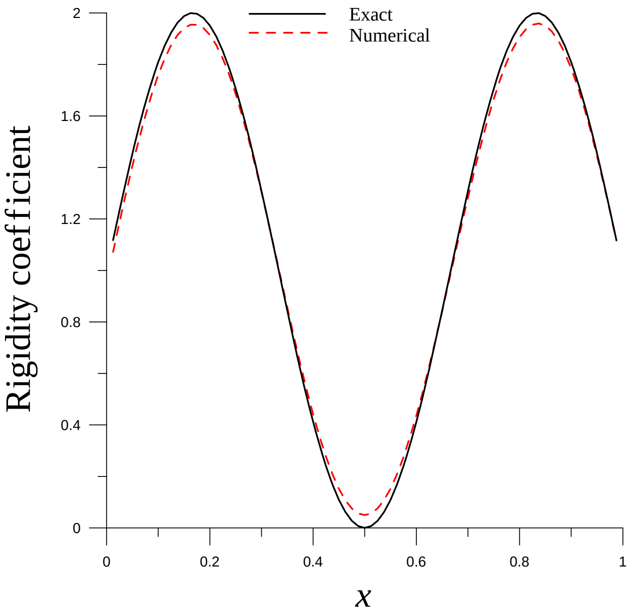

Figure 3: For example 3 comparing numerical and exact solutions of rigidity coefficient.

terion $\varepsilon = 10^{-4}$, which is convergent within 179 iterations. The numerical solution marked by the dashed line is close to the solid line of the exact solution as shown in Fig. 4(b), with the RMSE about $3.57 \times 10^{-2}$, and the maximal absolute error about $5.89 \times 10^{-2}$.

Now, we impose other boundary conditions and initial conditions obtained from Eq. (72) for $u(x,t)$, and the function $h(x,t)$ is calculated from Eq. (74) by inserting the same $u(x,t)$ in Eq. (72). Fig. 5 shows that the numerical solution marked by the dashed line is rather close to the solid line of the exact solution. The RMSE is about $1.06 \times 10^{-2}$, and the maximal absolute error is about $1.95 \times 10^{-2}$.

This example demonstrates that the new algorithm is not affected by the boundary conditions and initial conditions. Even under different boundary conditions and initial conditions, the new algorithm of LGAM led to the same $\alpha(x)$.

### 6.5 *Example 5*

The identified function $\alpha(x)$ is a two-hump function [Tadi (1998)]:

$$\alpha(x) = 1 + \exp\left(-\frac{(x-0.26)^2}{0.02}\right) + \exp\left(-\frac{(x-0.74)^2}{0.02}\right), \quad x \in (0,1), \tag{76}$$

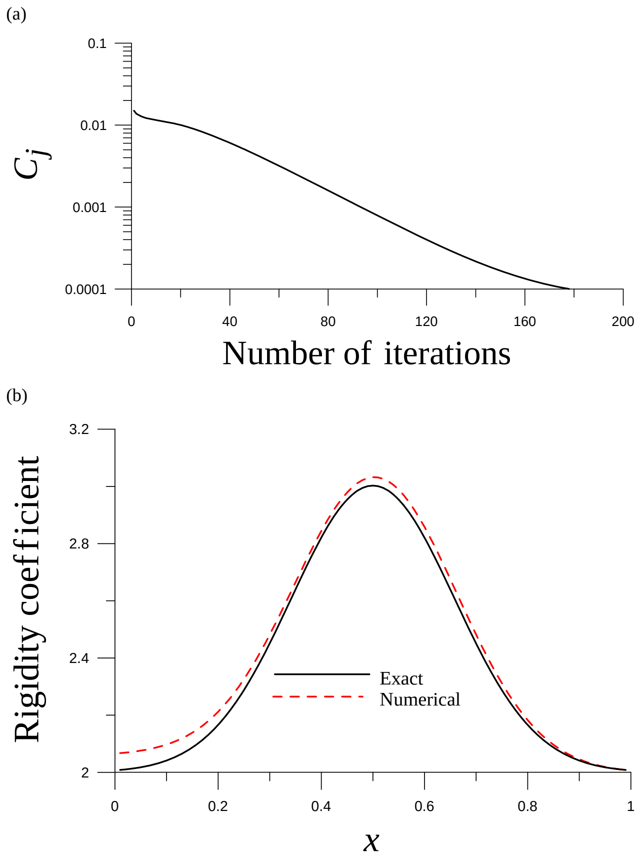

Figure 4: For example 4 : (a) the convergence rate, (b) comparing numerical and exact solutions of rigidity coefficient.

and the function $h(x,t)$ is calculated from Eq. (74) by inserting Eq. (70) for $u(x,t)$. In this identification of $\alpha(x)$ we have applied the LGAM with $\Delta x = 1/100$, $\Delta t = 0.06/50$, and $t_f = 0.06$. Fig. 6(a) shows the convergence rate under a stopping criterion $\varepsilon = 10^{-5}$, which is runned 400 iterations but not converges under the above convergence criterion. In Fig. 6(b), the zeroth, the fifth and the fiftieth iterations are plotted, which initially converges very fast with the fiftieth iteration being close to the exact solution. The numerical solution marked by the dashed line is close to the solid line of the exact solution as shown in Fig. 6(b), with the RMSE about $2.13 \times 10^{-2}$, and the maximal absolute error about $3.55 \times 10^{-2}$. As compared with the results obtained by Tadi (1998) as shown in Figs. 4 and 5 therein, the present method is better than the time-dependent regularization method proposed by Tadi

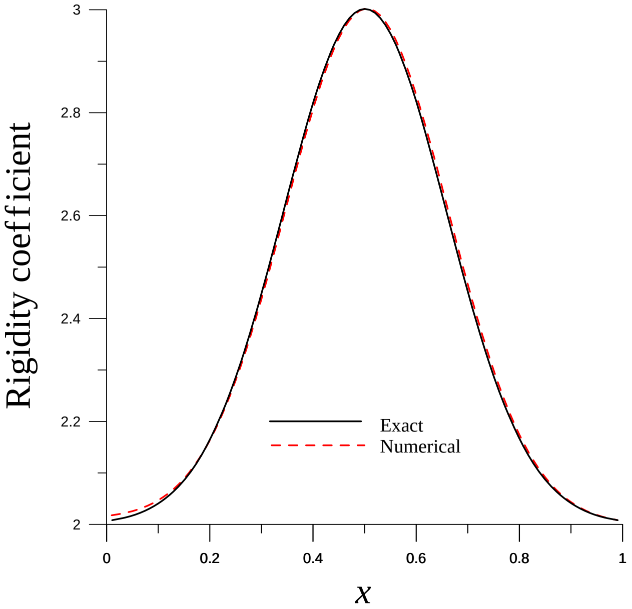

Figure 5: For example 4 comparing numerical and exact solutions of rigidity coefficient under different different boundary and initial conditions.

(1998) in three-aspect: convergent speed, accuracy, and simplicity.

## 7 Conclusions

A Lie-group adaptive method (LGAM) has been developed for the inverse scattering problem by imaging a spatially-dependent rigidity function in a one-dimensional rod. Eq. (63) is a critical equation, which plays an important role to adjust the parameter $\alpha(x)$ through iterations. The advantages of the present method are that no a priori information about the functional form of rigidity coefficient is necessary, and no extra measurement of data are required, in addition to the usual boundary conditions and initial conditions for the direct problem of wave propagation in a finite rod. In this regard the present method provides the most cheap tool to handle the inverse scattering problem. The accuracy and efficiency of the present algorithm are confirmed by comparing the estimated results with exact solutions. Through the above identifications of $\alpha(x)$ in Examples 1-5, it can be seen that when the functions $\alpha(x)$ are smooth, the convergences are fast and the estimations are rather accurate. Accordingly we can conclude that the LGAM is a powerful tool used in the reconstruction of parameter for the inverse scattering problem of wave propa-

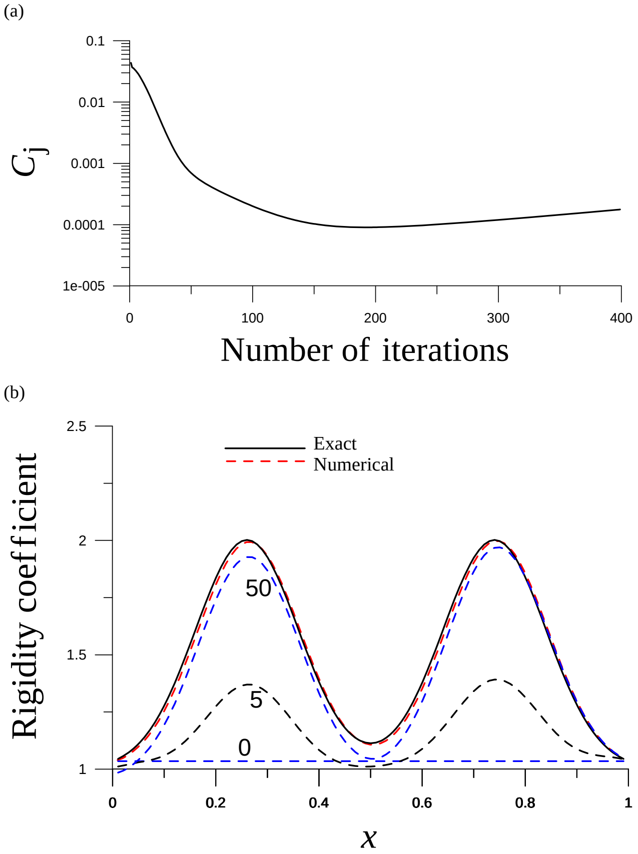

Figure 6: For example 5: (a) the convergence rate, (b) comparing numerical and exact solutions of rigidity coefficient. The zeroth, fifth and fiftieth iterations are shown for comparison.

gation. The success of the present method is hinged on a rationale that *the local in time differential governing equation (13) and the global in time algebraic equation (63) have to be self-adapted during the iteration process. This study may bring us to a new field of the inverse scattering problem of imaging the profile of material property, of which the work becomes quite easy than before.*

**Acknowledgement:** Taiwan's National Science Council project NSC-99-2221-E-002-074-MY3 granted to the author is highly appreciated.

## References

**Baev, A. V.** (1986): One formulation of the inverse boundary-value problem for the wave equation and an iterative solution method. *Dokl. Akad. Nauk SSSR*, vol. 287, pp. 818-821.

**Baev, A. V.** (1987): An iterative solution method for one inverse problem for the wave equation on an interval. *Zh. Vychisl. Mat. Mat. Fiz.*, vol. 27, pp. 1022-1031.

**Baev, A. V.** (1988): A solution method for the inverse scattering problem for the wave equation. *Zh. Vychisl. Mat. Mat. Fiz.*, vol. 28, pp. 25-33.

**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.** (2006a): 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.** (2006b): Efficient shooting methods for the second order ordinary differential equations. *CMES: Computer Modeling in Engineering & Sciences*, vol. 15, pp. 69-86.

**Liu, C.-S.** (2006c): The Lie-group shooting method for singularly perturbed two-point boundary value problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 15, pp. 179-196.

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

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

**Liu, C.-S.** (2007): Identification of temperature-dependent thermophysical properties in a partial differential equation subject to extra final measurement data. *Numer. Meth. Partial Diff. Eq.*, vol. 23, pp. 1083-1109.

**Liu, C.-S.** (2008a): An LGSM to identify nonhomogeneous heat conductivity functions by an extra measurement of temperature. *Int. J. Heat Mass Transfer*, vol. 51, pp. 2603-2613.

**Liu, C.-S.** (2008b): An LGEM to identify time-dependent heat conductivity function by an extra measurement of temperature gradient. *CMC: Computers, Materials & Continua*, vol. 7, pp. 81-95.

**Liu, C.-S.** (2009): A two-stage LGSM to identify time-dependent heat source through an internal measurement of temperature. *Int. J. Heat Mass Transfer*, vol. 52, pp. 1635-1642.

**Liu, C.-S.** (2010): A highly accurate LGSM for severely ill-posed BHCP under a large noise on the final time data. *Int. J. Heat Mass Transfer*, vol. 53, pp. 4132-4140.

**Liu, C.-S.; Atluri, S. N.** (2008): A novel fictitious time integration method for solving the discretized inverse Sturm-Liouville problems, for specified eigenvalues. *CMES: Computer Modeling in Engineering* & *Sciences*, vol. 36, pp. 261-285.

**Liu, C.-S.; Atluri, S. N.** (2010): An iterative and adaptive Lie-group method for solving the Calderón inverse problem. *CMES: Computer Modeling in Engineering & Sciences*, vol. 64, pp. 299-326.

**Majda, A.** (1976): High frequency asymptotics for the scattering matrix and the inverse problem of acoustical scattering. *Comm. Pure Appl. Math.*, vol. 29, pp. 261-291.

**Majda, A.** (1977): A representation formula for the scattering operator and the inverse problem for arbitrary bodies. *Comm. Pure Appl. Math.*, vol. 30, pp. 165-194.

**Na, S. W.; Kallivokas, L. F.** (2009): Direct time-domain soil profile reconstruction for one-dimensional semi-infinite domains. *Soil Dyna. Earthq. Eng.*, vol. 29, pp. 1016-1026.

**Tadi, M.** (1997): Explicit method for inverse wave scattering in solids. *Inverse Problems*, vol. 13, pp. 509-521.

**Tadi, M.** (1998): Evaluation of the elastic property based on boundary measurement. *Acta Mech.*, vol. 129, pp. 231-241.

**Tadi, M.** (1999): Inverse wave scattering in 2-D elastic solids. *Acta Mech.*, vol. 136, pp. 1-15.