 CMES, vol.84, no.4, pp.383-403, 2012

A Globally Optimal Iterative Algorithm to Solve an Ill-Posed Linear System

Chein-Shan Liu[1]

Abstract: An iterative algorithm based on the critical descent vector is proposed to solve an ill-posed linear system: $\mathbf{Bx}=\mathbf{b}$. We define a future cone in the Minkowski space as an invariant manifold, wherein the discrete dynamics evolves. A critical value α_c in the critical descent vector $\mathbf{u}=\alpha_c\mathbf{r}+\mathbf{B}^{\mathrm{T}}\mathbf{r}$ is derived, which renders the largest convergence rate as to be the **globally optimal iterative algorithm** (GOIA) among all the numerically iterative algorithms with the descent vector having the form $\mathbf{u}=\alpha\mathbf{r}+\mathbf{B}^{\mathrm{T}}\mathbf{r}$ to solve the ill-posed linear problems. Some numerical examples are used to reveal the superior performance of the GOIA.

Keywords: Ill-posed linear system, Globally optimal iterative algorithm (GOIA), Future cone, Invariant-manifold

1 Introduction

In this paper we numerically solve

$$\mathbf{Bx}=\mathbf{b}, \tag{1}$$

where $\mathbf{x}\in\mathbb{R}^n$ is an unknown vector, to be determined from a given coefficient matrix $\mathbf{B}\in\mathbb{R}^{n\times n}$ and the input $\mathbf{b}\in\mathbb{R}^n$. Eq. (1) might be an ill-posed system if the condition number cond($\mathbf{B}$) of $\mathbf{B}$ is quite large. We transform Eq. (1) to a system of nonlinear ODEs for $\mathbf{x}(t)$ where t is a time-like variable, and then derive a *globally optimal iterative algorithm* (GOIA), by solving those nonlinear ODEs for $\mathbf{x}(t)$ using the forward Euler scheme.

Instead of Eq. (1), we can solve a normal linear system:

$$\mathbf{Cx}=\mathbf{b}_1, \tag{2}$$

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

where

$$\mathbf{b}_1 = \mathbf{B}^{\mathrm{T}}\mathbf{b},$$
$$\mathbf{C} = \mathbf{B}^{\mathrm{T}}\mathbf{B} > \mathbf{0}. \tag{3}$$

Liu (2011a, 2012a) has derived the following relaxed steepest descent method (RSDM) to solve Eq. (2):
(i) Give an initial $\mathbf{x}_0$.
(ii) For $k = 0, 1, 2, \ldots$, we repeat the following computations:

$$\mathbf{R}_k = \mathbf{C}\mathbf{x}_k - \mathbf{b}_1, \tag{4}$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1-\gamma)\frac{\|\mathbf{R}_k\|^2}{\mathbf{R}_k^{\mathrm{T}}\mathbf{C}\mathbf{R}_k}\mathbf{R}_k. \tag{5}$$

If $\mathbf{x}_{k+1}$ converges according to a given stopping criterion $\|\mathbf{R}_{k+1}\| < \varepsilon$, then stop; otherwise, go to step (ii). A relaxation parameter γ appeared in Eq. (5) can enhance the convergence speed, of which the relaxation parameter γ is determined by the user.

There are some methods that converge significantly faster than the steepest descent method (SDM); unlike the conjugate gradient method (CGM), they insist their search directions to be the gradient vector at each iteration [Barzilai and Borwein (1988); Friedlander, Martinez, Molina and Raydan (1999); Raydan and Svaiter (2002); Dai and Yuan (2003); Dai, Hager, Schittkowsky and Zhang (2006); Ascher, van den Doel, Hunag and Svaiter (2009); Liu (2011a, 2012a)].

The SDM performs poorly, yielding iteration counts that grow linearly with cond($\mathbf{C}$) [Akaike (1959); Forsythe (1968); Nocedal, Sartenar and Zhu (2002)]. The slowness of SDM has to do with the choice of the steepest descent direction as well as the steplength. Liu and Atluri (2011) have explored a variant of the SDM by developing an optimal iterative algorithm driven by an optimal descent vector to solve Eq. (1). The resulting scheme [OIA/ODV] is given as follows:
(i) Select $0 \le \gamma < 1$, and give an initial $\mathbf{x}_0$.

(ii) For $k = 0, 1, 2 \ldots$, we repeat the following computations:

$$\begin{aligned}
&\mathbf{r}_k = \mathbf{B}\mathbf{x}_k - \mathbf{b}, \\
&\mathbf{v}_1^k = \mathbf{A}\mathbf{r}_k, \\
&\mathbf{v}_2^k = \mathbf{B}\mathbf{r}_k, \\
&\alpha_k = \frac{[\mathbf{v}_1^k, \mathbf{r}_k, \mathbf{v}_2^k] \cdot \mathbf{v}_1^k}{[\mathbf{v}_2^k, \mathbf{r}_k, \mathbf{v}_1^k] \cdot \mathbf{v}_2^k}, \qquad (6) \\
&\mathbf{u}_k = \alpha_k \mathbf{r}_k + \mathbf{B}^{\mathrm{T}} \mathbf{r}_k, \\
&\mathbf{v}_k = \mathbf{v}_1^k + \alpha_k \mathbf{v}_2^k, \\
&\mathbf{x}_{k+1} = \mathbf{x}_k - (1-\gamma) \frac{\mathbf{r}_k \cdot \mathbf{v}_k}{\|\mathbf{v}_k\|^2} \mathbf{u}_k. \qquad (7)
\end{aligned}$$

If $\mathbf{x}_{k+1}$ converges according to a given stopping criterion $\|\mathbf{r}_{k+1}\| < \varepsilon$, then stop; otherwise, go to step (ii). In above $\mathbf{A} := \mathbf{B}\mathbf{B}^{\mathrm{T}}$, and the following Jordan algebra for vectors in n-dimension:

$$[\mathbf{a}, \mathbf{b}, \mathbf{c}] = (\mathbf{a} \cdot \mathbf{b})\mathbf{c} - (\mathbf{c} \cdot \mathbf{b})\mathbf{a}, \quad \mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^n \qquad (8)$$

is developed by Liu (2000).

We will modify the descent direction as well as the steplength from a theoretical foundation of future cone, and the concept of *critical descent vector*. The remaining parts of this paper are arranged as follows. In Section 2 we start from a future cone in the Minkowski space to derive a system of nonlinear ODEs for the numerical solution of Eq. (1). Then, a genuine dynamics on the future cone is constructed in Section 3, resulting in a globally optimal iterative algorithm. The numerical examples are given in Section 4 to display some advantages of the newly developed iterative algorithm of GOIA. Finally, the conclusions are drawn in Section 5.

2 A future cone in the Minkowski space

For system (1), which is expressed in terms of the residual vector:

$$\mathbf{r} = \mathbf{B}\mathbf{x} - \mathbf{b} \qquad (9)$$

with $\mathbf{r} = \mathbf{0}$ in the final solution, we can formulate a scalar function:

$$h(\mathbf{x}, t) = \frac{1}{2} Q(t) \|\mathbf{r}(\mathbf{x})\|^2 - \frac{1}{2} \|\mathbf{r}_0\|^2 = 0, \qquad (10)$$

where we let $\mathbf{x}$ be a function of a time-like variable t, with initial values $\mathbf{x}(0) = \mathbf{x}_0$ and $\mathbf{r}_0 = \mathbf{r}(\mathbf{x}_0)$, and $Q(t) > 0$ is a monotonically increasing function of t. In terms

of

$$\mathbf{X} = \begin{bmatrix} \frac{\mathbf{r}}{\|\mathbf{r}_0\|} \\ \frac{1}{\sqrt{Q(t)}} \end{bmatrix}, \tag{11}$$

Eq. (10) represents a positive cone:

$$\mathbf{X}^{\mathrm{T}}\mathbf{g}\mathbf{X} = 0 \tag{12}$$

in the Minkowski space $\mathbb{M}^{n+1}$, endowed with an indefinite Minkowski metric tensor:

$$\mathbf{g} = \begin{bmatrix} \mathbf{I}_n & \mathbf{0}_{n\times 1} \\ \mathbf{0}_{1\times n} & -1 \end{bmatrix}, \tag{13}$$

where $\mathbf{I}_n$ is the $n \times n$ identity matrix. Because the last component $1/\sqrt{Q(t)}$ of $\mathbf{X}$ is positive, the cone in Eq. (12) is a future cone [Liu (2001)] as shown in Fig. 1.

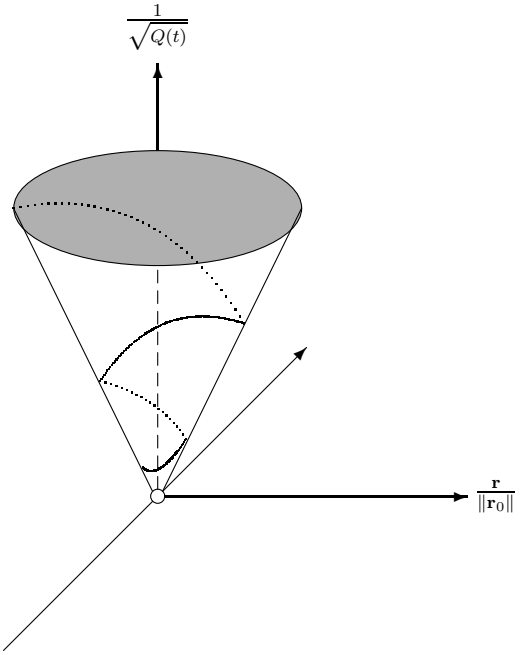

Figure 1: The construction of cone in the Minkowski space for solving ill-posed linear system signifies a conceptual breakthrough.

When $Q > 0$, the manifold defined by Eq. (10) is continuous and differentiable, and thus as a consequence of the consistency condition, we have

$$\frac{1}{2}\dot{Q}(t)\|\mathbf{r}(\mathbf{x})\|^2 + Q(t)\mathbf{R}\cdot\dot{\mathbf{x}} = 0, \tag{14}$$

which is obtained by taking the differential of Eq. (10) with respect to t and considering $\mathbf{x} = \mathbf{x}(t)$ and $h(\mathbf{x},t) = 0$ for all t. Corresponding to $\mathbf{r}$ in Eq. (9),

$$\mathbf{R} := \mathbf{B}^{\mathrm{T}}\mathbf{r} \tag{15}$$

is the steepest descent vector.

We suppose that the evolution of $\mathbf{x}$ is driven by a vector $\mathbf{u}$:

$$\dot{\mathbf{x}} = \lambda\mathbf{u}, \tag{16}$$

where

$$\mathbf{u} = \alpha\mathbf{r} + \mathbf{R} \tag{17}$$

is a suitable combination of the weighted residual vector $\alpha\mathbf{r}$ and the steepest descent vector $\mathbf{R}$. Note that $\mathbf{u}$ is not a vector along the *normal* to the hyper-surface $h(\mathbf{x},t) = 0$, and instead of it is a non-normal descent vector.

Inserting Eq. (16) into Eq. (14) we can derive a nonlinear ODEs system:

$$\dot{\mathbf{x}} = -q(t)\frac{\|\mathbf{r}\|^2}{\mathbf{r}^{\mathrm{T}}\mathbf{v}}\mathbf{u}, \tag{18}$$

where

$$\mathbf{A} := \mathbf{B}\mathbf{B}^{\mathrm{T}}, \tag{19}$$

$$\mathbf{v} := \mathbf{B}\mathbf{u} = \mathbf{v}_1 + \alpha\mathbf{v}_2 = \mathbf{A}\mathbf{r} + \alpha\mathbf{B}\mathbf{r}, \tag{20}$$

$$q(t) := \frac{\dot{Q}(t)}{2Q(t)}. \tag{21}$$

Hence, in our algorithm, if $Q(t)$ can be guaranteed to be a monotonically increasing function of t, we have an absolutely convergent property in solving the linear equations system (1):

$$\|\mathbf{r}(\mathbf{x})\|^2 = \frac{\|\mathbf{r}_0\|^2}{Q(t)}. \tag{22}$$

Thus, we can observe that the path of $\mathbf{X}$ gradually moves down to the vertex point along the cone defined by Eq. (12) as schematically shown in Fig. 1.

3 Dynamics on the future cone

3.1 Discretizing and keeping $\mathbf{x}$ on the manifold

Now we discretize the continuous dynamics in Eq. (18) into a discrete dynamics by applying the forward Euler scheme:

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \beta \frac{\|\mathbf{r}\|^2}{\mathbf{r}^{\mathrm{T}}\mathbf{v}}\mathbf{u}, \tag{23}$$

where

$$\beta = q(t)\Delta t \tag{24}$$

is the stepsize. Correspondingly, $\mathbf{u}$ is a search direction endowed with a steplength $\beta\|\mathbf{r}\|^2/(\mathbf{r}^{\mathrm{T}}\mathbf{v})$.

In order to keep $\mathbf{x}$ on the manifold (22) we can consider the evolution of $\mathbf{r}$ along the path $\mathbf{x}(t)$ by

$$\dot{\mathbf{r}} = \mathbf{B}\dot{\mathbf{x}} = -q(t)\frac{\|\mathbf{r}\|^2}{\mathbf{r}^{\mathrm{T}}\mathbf{v}}\mathbf{v}. \tag{25}$$

Similarly we use the forward Euler scheme to integrate Eq. (25), obtaining

$$\mathbf{r}(t+\Delta t) = \mathbf{r}(t) - \beta \frac{\|\mathbf{r}\|^2}{\mathbf{r}^{\mathrm{T}}\mathbf{v}}\mathbf{v}, \tag{26}$$

from which by taking the square-norms of both sides and using Eq. (22) we can obtain

$$\frac{C}{Q(t+\Delta t)} = \frac{C}{Q(t)} - 2\beta\frac{C}{Q(t)} + \beta^2\frac{C}{Q(t)}\frac{\|\mathbf{r}\|^2}{(\mathbf{r}^{\mathrm{T}}\mathbf{v})^2}\|\mathbf{v}\|^2. \tag{27}$$

Thus, dividing both sides by $C/Q(t)$ leads to

$$a_0\beta^2 - 2\beta + 1 - \frac{Q(t)}{Q(t+\Delta t)} = 0, \tag{28}$$

where

$$a_0 := \frac{\|\mathbf{r}\|^2\|\mathbf{v}\|^2}{(\mathbf{r}^{\mathrm{T}}\mathbf{v})^2} \geq 1. \tag{29}$$

As a result $h(\mathbf{x},t) = 0$, $t \in \{0,1,2,\ldots\}$ remains to be an invariant-manifold in the Minkowski space for the discrete time dynamical system $h(\mathbf{x}(t),t) = 0$ on the future cone.

3.2 A discrete dynamics

Let

$$s = \frac{Q(t)}{Q(t+\Delta t)} = \frac{\|\mathbf{r}(\mathbf{x}(t+\Delta t))\|^2}{\|\mathbf{r}(\mathbf{x}(t))\|^2}. \tag{30}$$

From Eqs. (28) and (30) it follows that

$$a_0\beta^2 - 2\beta + 1 - s = 0. \tag{31}$$

From Eq. (31), we can take the solution of β to be

$$\beta = \frac{1-\sqrt{1-(1-s)a_0}}{a_0}, \text{ if } 1-(1-s)a_0 \geq 0. \tag{32}$$

Let

$$1-(1-s)a_0 = \gamma^2 \geq 0, \tag{33}$$

$$s = 1 - \frac{1-\gamma^2}{a_0}, \tag{34}$$

such that the condition $1-(1-s)a_0 \geq 0$ in Eq. (32) is automatically satisfied. Thus, from Eqs. (32) and (33) it follows that

$$\beta = \frac{1-\gamma}{a_0}, \tag{35}$$

and from Eqs. (23) and (29) we can further derive

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - (1-\gamma)\frac{\mathbf{r}^{\mathrm{T}}(t)\mathbf{v}(t)}{\|\mathbf{v}(t)\|^2}\mathbf{u}(t), \tag{36}$$

where

$$0 \leq \gamma < 1 \tag{37}$$

is a relaxation parameter chosen by the user.

Under the above conditions (29) and (37), from Eqs. (30) and (34) we can prove that the new algorithm satisfies

$$\text{Convergence Rate} := \frac{\|\mathbf{r}(t)\|}{\|\mathbf{r}(t+\Delta t)\|} = \frac{1}{\sqrt{s}} > 1. \tag{38}$$

The property in Eq. (38) is very important, since it guarantees the new algorithm to be absolutely convergent to the true solution. *Smaller s implies Faster convergence speed.*

3.3 A critical descent vector $\mathbf{u}=\alpha_c\mathbf{r}+\mathbf{R}$

Up to here we not yet specify how to choose the parameter α in the algorithm (36). Previously, Liu and Atluri (2011) derived the optimal α by letting $\partial s/\partial\alpha=0$ (or equivalently, $\partial a_0/\partial\alpha=0$). Here, we try another approach and attempt to develop a more powerful selection of α; however, before that we give a remark about the best choice of the descent vector $\mathbf{u}$.

Remark 1: The best choice of $\mathbf{u}$ would be $\mathbf{u}=\mathbf{B}^{-1}\mathbf{r}$, which by Eq. (20) leads to $\mathbf{v}=\mathbf{r}$, and by Eq. (29) further leads to the smallest value of $a_0=1$. If $\mathbf{u}=\mathbf{B}^{-1}\mathbf{r}$ is realizable, we have $s=\gamma^2$ by Eq. (34), and thus from Eq. (38) we have an infinite convergence rate by letting $\gamma=0$. In this regard $\mathbf{B}^{-1}\mathbf{r}$ is the *best descent vector* for a numerical algorithm used to solve Eq. (1). However, if one has such the best descent vector $\mathbf{u}=\mathbf{B}^{-1}\mathbf{r}$ at hand, Eq. (1) is already solved, and one no longer needs any numerical method. Obviously, in order to use $\mathbf{u}=\mathbf{B}^{-1}\mathbf{r}$ we have faced the same difficulty to find $\mathbf{B}^{-1}$ as that to find the solution of $\mathbf{x}$ in Eq. (1) by $\mathbf{x}=\mathbf{B}^{-1}\mathbf{b}$.

Below, we introduce a new method to approximate the *best descent vector* according to the following equivalent relation:

$$a_0=1\equiv\mathbf{u}=\mathbf{B}^{-1}\mathbf{r}. \tag{39}$$

Motivated by the above remark about the best descent vector, which is however very difficult to be realized in practice, we can slightly relax the requirement of the value of a_0 to be 1, which means that we can relax the choice of $\mathbf{u}=\mathbf{B}^{-1}\mathbf{r}$ by Eq. (39). Instead of, we can determine a suitable α such that a_0 defined by Eq. (29) is given by

$$a_0:=\frac{\|\mathbf{r}\|^2\|\mathbf{v}\|^2}{(\mathbf{r}^{\mathrm{T}}\mathbf{v})^2}=a_s. \tag{40}$$

When a_s is near to 1, the convergence speed is very fast. Liu (2012b) first used the above idea to develop some effective numerical algorithms to solve the ill-posed linear problems. Inserting Eq. (20) for $\mathbf{v}$ into the above equation, and through some elementary operations we can derive a quadratic equation to solve α:

$$e_1\alpha^2+e_2\alpha+e_3=0, \tag{41}$$

where

$$e_1:=\|\mathbf{r}\|^2\|\mathbf{v}_2\|^2-a_s(\mathbf{r}\cdot\mathbf{v}_2)^2, \tag{42}$$

$$e_2:=2\|\mathbf{r}\|^2\mathbf{v}_1\cdot\mathbf{v}_2-2a_s\mathbf{r}\cdot\mathbf{v}_1\mathbf{r}\cdot\mathbf{v}_2, \tag{43}$$

$$e_3:=\|\mathbf{r}\|^2\|\mathbf{v}_1\|^2-a_s(\mathbf{r}\cdot\mathbf{v}_1)^2. \tag{44}$$

If the following condition is satisfied

$$D := e_2^2 - 4e_1e_3 \geq 0, \tag{45}$$

then a **real solution** of α is found to be

$$\alpha = \frac{\sqrt{D} - e_2}{2e_1}. \tag{46}$$

Inserting Eqs. (42)-(44) into the critical equation:

$$D = e_2^2 - 4e_1e_3 = 0, \tag{47}$$

we can derive an algebraic equation to determine which a_s is the lowest bound of Eq. (45). This lowest bound is a critical value denoted by a_c, and for all $a_s \geq a_c$ it can satisfy Eq. (45). From Eq. (47) through some operations, the critical value of a_c can be solved as:

$$a_c = \frac{\|\mathbf{r}\|^2[\|\mathbf{v}_1\|^2\|\mathbf{v}_2\|^2 - (\mathbf{v}_1 \cdot \mathbf{v}_2)^2]}{\|[\mathbf{v}_1, \mathbf{r}, \mathbf{v}_2]\|^2}. \tag{48}$$

Then insert it for a_s into Eqs. (42) and (43) we can obtain a critical value α_c for α by Eq. (46):

$$\alpha_c = \frac{a_c\mathbf{r} \cdot \mathbf{v}_1\mathbf{r} \cdot \mathbf{v}_2 - \|\mathbf{r}\|^2\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{r}\|^2\|\mathbf{v}_2\|^2 - a_c(\mathbf{r} \cdot \mathbf{v}_2)^2}, \tag{49}$$

where $D = 0$ was used in view of Eq. (47). Hence,

$$\mathbf{u} = \alpha_c\mathbf{r} + \mathbf{R} \tag{50}$$

is a critical descent vector. **Due to its criticality, if one attempts to find a better descent vector than that in Eq. (50), there would be no real solution of α, and hence no real descent vector of $\mathbf{u}$**.

3.4 The globally optimal iterative algorithm

According to the above theory we can derive a globally optimal iterative algorithm (GOIA) to solve Eq. (1) by
(i) Select $0 \leq \gamma < 1$, and give an initial $\mathbf{x}_0$.

(ii) For $k = 0, 1, 2 \ldots$, we repeat the following computations:

$$\begin{aligned}
&\mathbf{r}_k = \mathbf{B}\mathbf{x}_k - \mathbf{b}, \\
&\mathbf{v}_1^k = \mathbf{A}\mathbf{r}_k, \\
&\mathbf{v}_2^k = \mathbf{B}\mathbf{r}_k, \\
&a_c^k = \frac{\|\mathbf{v}_1^k\|^2\|\mathbf{v}_2^k\|^2 - (\mathbf{v}_1^k \cdot \mathbf{v}_2^k)^2}{\|[\mathbf{v}_1^k, \mathbf{r}_k, \mathbf{v}_2^k]\|^2}, \\
&\alpha_c^k = \frac{a_c^k \mathbf{r}_k \cdot \mathbf{v}_1^k \mathbf{r}_k \cdot \mathbf{v}_2^k - \mathbf{v}_1^k \cdot \mathbf{v}_2^k}{\|\mathbf{v}_2^k\|^2 - a_c^k (\mathbf{r}_k \cdot \mathbf{v}_2^k)^2}, \\
&\mathbf{u}_k = \alpha_c^k \mathbf{r}_k + \mathbf{B}^{\mathrm{T}} \mathbf{r}_k, \\
&\mathbf{v}_k = \mathbf{v}_1^k + \alpha_c^k \mathbf{v}_2^k, \\
&\mathbf{x}_{k+1} = \mathbf{x}_k - (1-\gamma)\frac{\mathbf{r}_k \cdot \mathbf{v}_k}{\|\mathbf{v}_k\|^2}\mathbf{u}_k.
\end{aligned} \tag{51}$$

If $\mathbf{x}_{k+1}$ converges according to a given stopping criterion $\|\mathbf{r}_{k+1}\| < \varepsilon$, then stop; otherwise, go to step (ii). This algorithm is better than the algorithm OIA/ODV given by Eq. (7). In the above, a common factor $\|\mathbf{r}_k\|^2$ in a_c^k and α_c^k is cancelled out.

4 Numerical examples

In order to assess the performance of the newly developed method of the globally optimal iterative algorithm (GOIA), let us investigate the following examples.

4.1 Example 1

When we apply a central difference scheme to the following two-point boundary value problem:

$$\begin{aligned}
&-u''(x) = f(x), \;\; 0 < x < 1, \\
&u(0) = a, \;\; u(1) = b,
\end{aligned} \tag{52}$$

we can derive a linear equations system

$$\mathbf{B}\mathbf{u} = \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} (\Delta x)^2 f(\Delta x) + a \\ (\Delta x)^2 f(2\Delta x) \\ \vdots \\ (\Delta x)^2 f((n-1)\Delta x) \\ (\Delta x)^2 f(n\Delta x) + b \end{bmatrix}, \tag{53}$$

where $\Delta x = 1/(n+1)$ is the spatial length, and $u_i = u(i\Delta x),\ i = 1,\ldots,n$, are unknown values of $u(x)$ at the grid points $x_i = i\Delta x$. $u_0 = a$ and $u_{n+1} = b$ are the given boundary conditions.

In this numerical test we fix $n = 200$ and thus the condition number of $\mathbf{B}$ is 16210.7. Let us consider the boundary value problem in Eq. (52) with $f(x) = \sin \pi x$. The exact solution is

$$u(x) = a + (b-a)x + \frac{1}{\pi^2}\sin \pi x, \tag{54}$$

where we fix $a = 1$ and $b = 2$.

A relative random noise with intensity $\sigma = 0.01$ is added into the data on the right-hand side of Eq. (53). Under a moderate convergence criterion with $\varepsilon = 10^{-7}$, we find that the CGM converges with 1332 iterations as shown in Fig. 2(a) by the solid line, and the maximum error as shown in Fig. 2(b) is 3.13×10^{-5}. Under the same convergence criterion we find that the GOIA with $\gamma = 0.25$ converges more faster with 1121 iterations as shown in Fig. 2(a) by the dashed line, and the maximum error as shown in Fig. 2(b) is 1.15×10^{-5}. The present GOIA is better than the CGM for a mildly ill-conditioned system.

4.2 Example 2

In order to compare the GOIA with the OIA/ODV [Liu and Atluri (2011)], we solve the following Laplace equation:

$$\begin{aligned} &u_{xx} + u_{yy} = 0,\ \ 0 < x < 1,\ \ 0 < y < 1, \\ &u(x,y) = \sin x \cosh y. \end{aligned} \tag{55}$$

The boundary conditions can be derived from the exact solution. Under a finite-difference discretization with $\Delta x = \Delta y = 1/16$ and under the same convergence criterion $\varepsilon = 10^{-6}$, we compare the residual errors in Fig. 3(a), where the OIA/ODV with $\gamma = 0.05$ is convergent with 68 iterations, and the GOIA with $\gamma = 0.06$ is convergent with 66 iterations. The convergence rates are compared in Fig. 3(b). The above two algorithms provide very accurate solutions with the maximum error being 2.73×10^{-5}.

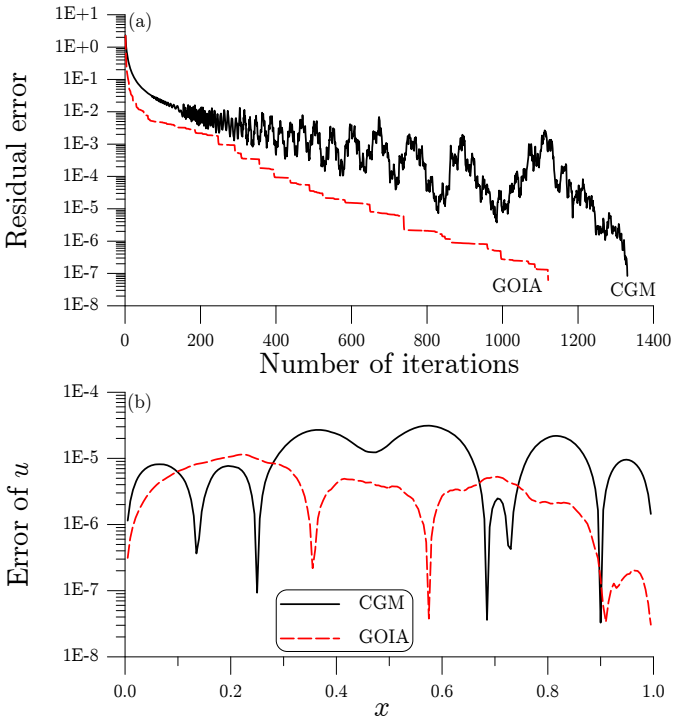

Figure 2: For example 1: comparing (a) the residual errors, and (b) the numerical errors obtained by the CGM and GOIA.

4.3 Example 3

Finding an n-degree polynomial function $p(x) = a_0 + a_1 x + \ldots + a_n x^n$ to best match a continuous function $f(x)$ in the interval of $x \in [0,1]$:

$$\min_{\deg(p)\le n} \int_0^1 [f(x) - p(x)]^2 dx, \tag{56}$$

leads to a problem governed by Eq. (1), where $\mathbf{B}$ is the $(n+1)\times(n+1)$ Hilbert matrix, defined by

$$B_{ij} = \frac{1}{i+j-1}, \tag{57}$$

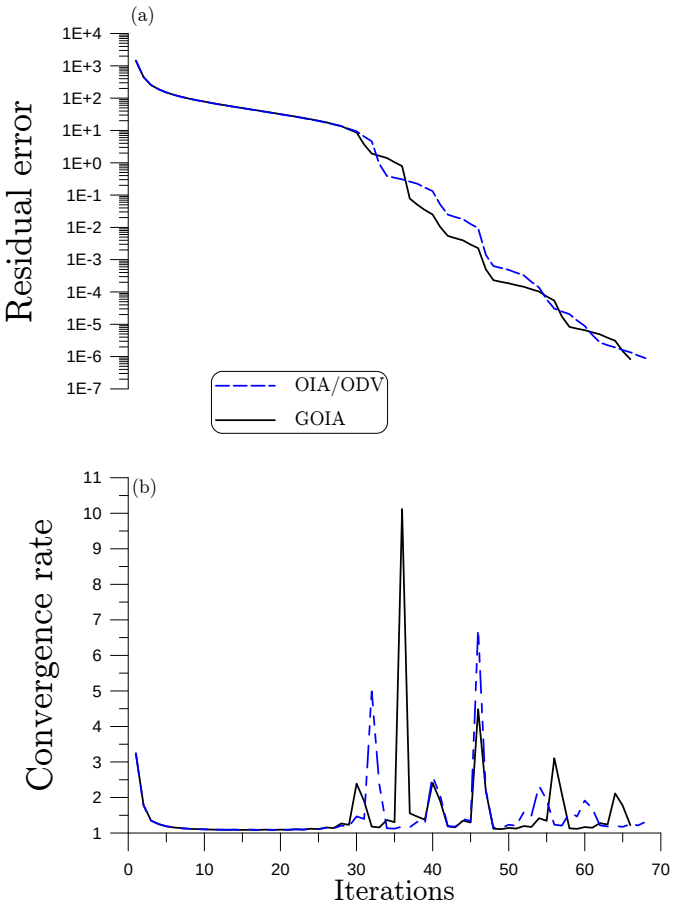

Figure 3: For the Laplace equation: comparing (a) the residual errors, and (b) the convergence rates obtained by the OIA/ODV and GOIA.

$\mathbf{x}$ is composed of the $n+1$ coefficients $a_0, a_1, \ldots, a_n$ appeared in $p(x)$, and

$$\mathbf{b} = \begin{bmatrix} \int_0^1 f(x)dx \\ \int_0^1 xf(x)dx \\ \vdots \\ \int_0^1 x^n f(x)dx \end{bmatrix} \tag{58}$$

is uniquely determined by the function $f(x)$.

In this example we consider a highly ill-conditioned linear system (1) with $\mathbf{B}$ given

by Eq. (57). The ill-posedness of Eq. (1) with the above **B** increases very fast with an exponential growth with n.

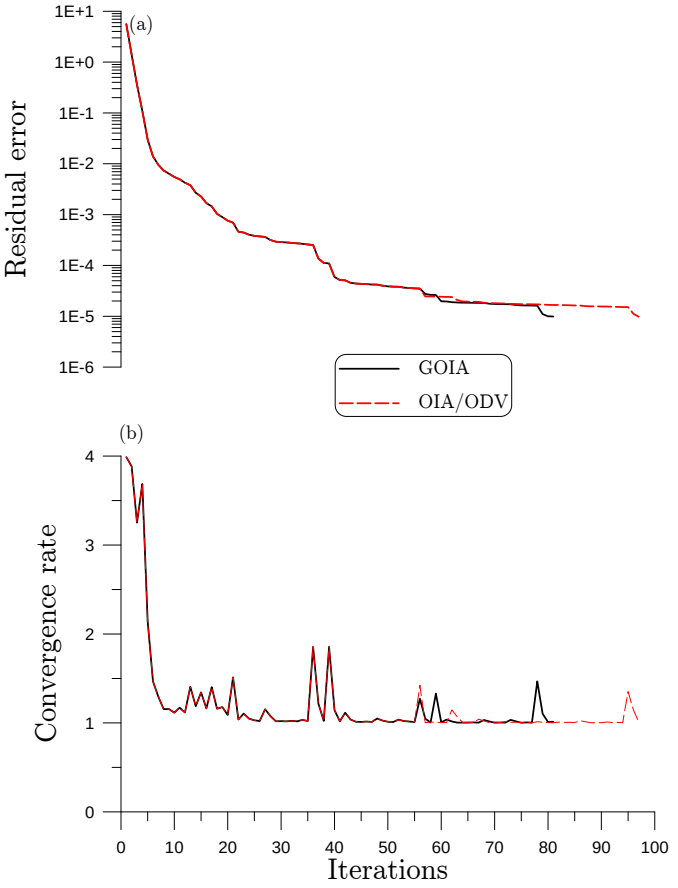

Figure 4: For the Hilbert linear problem, comparing (a) the residual errors, and (b) the convergence rates obtained by the OIA/ODV and GOIA.

Let us consider $n = 50$. For this problem the condition number is about 1.1748×10^{19}. We consider a constant solution $x_1 = x_2 = \ldots = x_{50} = 1$. The noise being imposed on the given data is fixed to be $\sigma = 10^{-8}$, and the convergence criterion $\varepsilon = 10^{-5}$ is fixed. We compare the residual errors in Fig. 4(a), where the OIA/ODV with $\gamma = 0.25$ is convergent with 97 iterations, and the GOIA with $\gamma = 0.25$ is convergent with 81 iterations. The convergence rates are compared in Fig. 4(b). The above two algorithms yield very accurate solutions with the maximum error being 1.05×10^{-2}.

4.4 Example 4

When the backward heat conduction problem (BHCP) is considered in a spatial interval of $0 < x < \ell$ by subjecting to the boundary conditions at two ends of a slab:

$$u_t(x,t) = \alpha u_{xx}(x,t), \;\; 0 < t < T, \;\; 0 < x < \ell, \tag{59}$$

$$u(0,t) = u_0(t), \;\; u(\ell,t) = u_\ell(t), \tag{60}$$

we solve u under a final time condition:

$$u(x,T) = u^T(x). \tag{61}$$

The fundamental solution to Eq. (59) is given as follows:

$$K(x,t) = \frac{H(t)}{2\sqrt{\alpha\pi t}} \exp\left(\frac{-x^2}{4\alpha t}\right), \tag{62}$$

where $H(t)$ is the Heaviside function.

The method of fundamental solutions (MFS) has a broad application in engineering computations. In the MFS the solution of u at the field point $\mathbf{z} = (x,t)$ can be expressed as a linear combination of the fundamental solutions $U(\mathbf{z},\mathbf{s}_j)$:

$$u(\mathbf{z}) = \sum_{j=1}^{n} c_j U(\mathbf{z},\mathbf{s}_j), \;\; \mathbf{s}_j = (\eta_j,\tau_j) \in \Omega^c, \tag{63}$$

where n is the number of source points, c_j are unknown coefficients, and $\mathbf{s}_j$ are source points being located in the complement Ω^c of $\Omega = [0,\ell] \times [0,T]$. For the heat conduction equation we have

$$U(\mathbf{z},\mathbf{s}_j) = K(x-\eta_j, t-\tau_j). \tag{64}$$

It is known that the distribution of source points in the MFS has a great influence on the accuracy and stability. In a practical application of MFS to solve the BHCP, the source points are uniformly located on two vertical straight lines parallel to the t-axis and one horizontal line over the final time, which was adopted by Hon and Li (2009) and Liu (2011b), showing a large improvement than the line location of source points below the initial time. After imposing the boundary conditions and the final time condition on Eq. (63) we can obtain a linear equations system:

$$\mathbf{Bx} = \mathbf{b}, \tag{65}$$

where

$$B_{ij} = U(\mathbf{z}_i, \mathbf{s}_j), \ \ \mathbf{x} = (c_1, \cdots, c_n)^{\mathrm{T}},$$
$$\mathbf{b} = (u_\ell(t_i),\ i = 1, \ldots, m_1; u^T(x_j),\ j = 1, \ldots, m_2; u_0(t_k),\ k = m_1, \ldots, 1)^{\mathrm{T}}, \quad (66)$$

and $n = 2m_1 + m_2$.

Since the BHCP is highly ill-posed, the ill-condition of the coefficient matrix **B** in Eq. (65) is serious. To overcome the ill-posedness of Eq. (65) we can use the new method to solve this problem. Here we compare the numerical solution with an exact solution:

$$u(x,t) = \cos(\pi x)\exp(-\pi^2 t).$$

For the case with $T = 1$ the value of final time data is in the order of 10^{-4}, which is much small in a comparison with the value of the initial temperature $u_0(x) = \cos(\pi x)$ to be retrieved, which is $O(1)$. We solve this problem by the OIA/ODV with $\gamma = 0.25$, and the GOIA with $\gamma = 0.1$. As shown in Fig. 5(a), both the OIA/ODV and the GOIA do not converge within 20000 iterations under the convergence criterion $\varepsilon = 10^{-2}$. We have added a relative random noise with an intensity $\sigma = 10\%$ on the final time data, of which we compare the initial time data computed by the OIA/ODV and the GOIA with the exact one in Fig. 5(b) by showing the numerical errors, of which the maximum error of OIA/ODV is smaller than 0.017 and that for the GOIA is smaller than 0.00757. It indicates that the present iterative algorithms are robust against noise, and can provide very accurate numerical results.

4.5 Example 5

We solve the Cauchy problem of the Laplace equation under the incomplete boundary conditions:

$$\Delta u = u_{rr} + \frac{1}{r}u_r + \frac{1}{r^2}u_{\theta\theta} = 0, \ \ r < \rho, \ \ 0 \le \theta \le 2\pi, \quad (67)$$
$$u(\rho, \theta) = h(\theta), \ \ 0 \le \theta \le \pi, \quad (68)$$
$$u_n(\rho, \theta) = g(\theta), \ \ 0 \le \theta \le \pi, \quad (69)$$

where $h(\theta)$ and $g(\theta)$ are given functions, and $\rho = \rho(\theta)$ is a given contour to describe the boundary shape. The contour in the polar coordinates is specified by $\Gamma = \{(r, \theta) | r = \rho(\theta),\ 0 \le \theta \le 2\pi\}$, which is the boundary of the problem domain Ω, and n denotes the outward normal direction. We need to find the boundary data on the lower half contour for the completeness of boundary data.

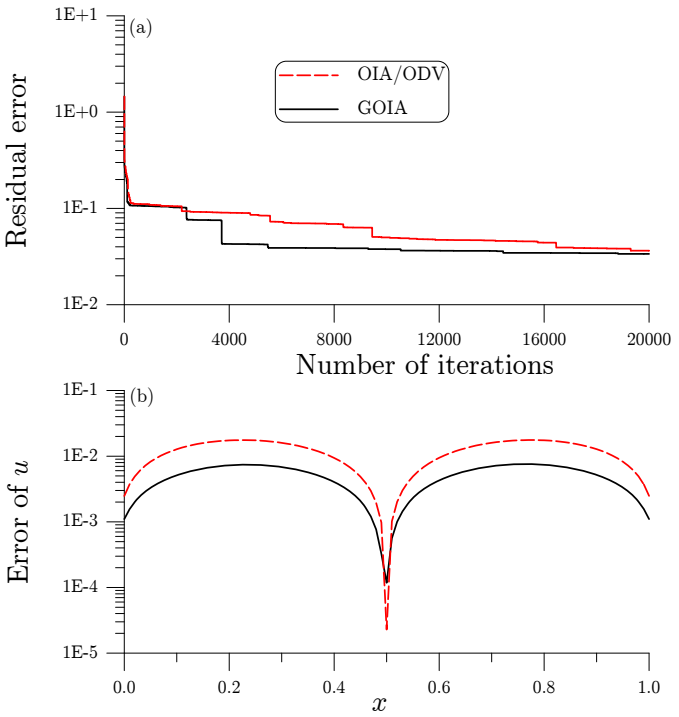

Figure 5: For the BHCP, comparing (a) the residual errors, and (b) the numerical errors obtained by the OIA/ODV and GOIA.

In the MFS the trial solution of u at the field point $\mathbf{z} = (r\cos\theta, r\sin\theta)$ can be expressed as a linear combination of the fundamental solutions $U(\mathbf{z},\mathbf{s}_j)$:

$$u(\mathbf{z}) = \sum_{j=1}^{n} c_j U(\mathbf{z},\mathbf{s}_j), \quad \mathbf{s}_j \in \Omega^c, \tag{70}$$

where n is the number of source points, c_j are the unknown coefficients, $\mathbf{s}_j$ are the source points, and Ω^c is the complementary set of Ω. For the Laplace equation (67) we have the fundamental solutions:

$$U(\mathbf{z},\mathbf{s}_j) = \ln r_j, \quad r_j = \|\mathbf{z}-\mathbf{s}_j\|. \tag{71}$$

In the practical application of MFS, usually the source points are distributed uniformly on a circle with a radius R, such that after imposing the boundary conditions (68) and (69) on Eq. (70) we can obtain a linear equations system:

$$\mathbf{Bx} = \mathbf{b}, \tag{72}$$

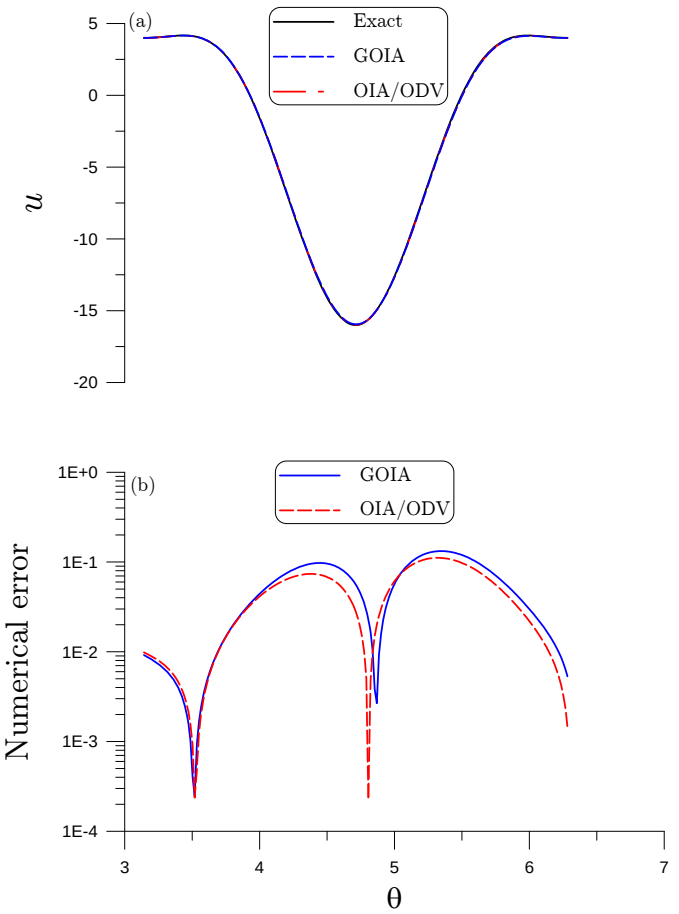

Figure 6: For the Cauchy problem, (a) comparing the numerical solutions obtained by the OIA/ODV and GOIA with the exact one, and (b) showing the numerical errors.

where

$$\begin{aligned}
&\mathbf{z}_i = (z_i^1, z_i^2) = (\rho(\theta_i)\cos\theta_i, \rho(\theta_i)\sin\theta_i),\\
&\mathbf{s}_j = (s_j^1, s_j^2) = (R\cos\theta_j, R\sin\theta_j),\\
&B_{ij} = \ln\|\mathbf{z}_i - \mathbf{s}_j\|, \quad \text{if } i \text{ is odd},\\
&B_{ij} = \frac{\eta(\theta_i)}{\|\mathbf{z}_i - \mathbf{s}_j\|^2}\left(\rho(\theta_i) - s_j^1\cos\theta_i - s_j^2\sin\theta_i\right.\\
&\quad \left. - \frac{\rho'(\theta_i)}{\rho(\theta_i)}[s_j^1\sin\theta_i - s_j^2\cos\theta_i]\right), \text{if } i \text{ is even},\\
&\mathbf{x} = (c_1, \ldots, c_n)^{\mathrm{T}}, \quad \mathbf{b} = (h(\theta_1), g(\theta_1), \ldots, h(\theta_m), g(\theta_m))^{\mathrm{T}},
\end{aligned} \tag{73}$$

in which $n = 2m$, and

$$\eta(\theta) = \frac{\rho(\theta)}{\sqrt{\rho^2(\theta) + [\rho'(\theta)]^2}}. \tag{74}$$

This example poses a great challenge to test the efficiency of linear equations solver, because the Cauchy problem is highly ill-posed. We fix $n = 60$ and take a circle with a constant radius $R = 15$ to distribute the source points. We apply both the OIA/ODV and the GOIA to solve Eq. (72) within 5000 iterations, where a noise with intensity $\sigma = 10\%$ is imposed on the given data. While in Fig. 6(a) we compare the numerical solutions obtained by the OIA/ODV and the GOIA with the exact data given by $u = \rho^2 \cos(2\theta)$, $\pi \le \theta < 2\pi$, where $\rho = \sqrt{10 - 6\cos(2\theta)}$, the numerical errors are plotted in Fig. 6(b). It can be seen that both the OIA/ODV and the GOIA provide very accurate solutions.

5 Conclusions

In the present paper we were based on a critical descent vector to derive a *globally optimal iterative algorithm* (GOIA), which can largely accelerate the convergence speed in the numerical solution of ill-posed linear system $\mathbf{Bx} = \mathbf{b}$. It was demonstrated that the critical value α_c in the critical descent vector leads to the **largest convergence rate** among all the descent vectors specified by $\mathbf{u} = \alpha\mathbf{r} + \mathbf{B}^{\mathrm{T}}\mathbf{r}$. Due to its criticality, if one attempts to find a better descent vector than $\mathbf{u} = \alpha_c\mathbf{r} + \mathbf{B}^{\mathrm{T}}\mathbf{r}$, there would be no real descent vector of $\mathbf{u}$. Hence, in the present framework of the future cone and giving the descent vector by $\mathbf{u} = \alpha\mathbf{r} + \mathbf{B}^{\mathrm{T}}\mathbf{r}$, the present GOIA is the **globally optimal iterative algorithm** to solve Eq. (1).

Acknowledgement: Taiwan's National Science Council project NSC-100-2221-E-002-165-MY3 and the 2011 Outstanding Research Award, as well as the 2011 Taiwan Research Front Award from Thomson Reuters granted to the author are highly appreciated.

References

Akaike, H. (1959): On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. *Ann. Inst. Stat. Math. Tokyo*, vol. 11, pp. 1-16.

Ascher, U.; van den Doel, K.; Hunag, H.; Svaiter, B. (2009): Gradient descent and fast artificial time integration. *M2AN*, vol. 43, pp. 689-708.

Barzilai, J.; Borwein, J. M. (1988): Two point step size gradient methods. *IMA J. Numer. Anal.*, vol. 8, pp. 141-148.

Dai, Y. H.; Yuan, Y. (2003): Alternate minimization gradient method. *IMA J. Numer. Anal.*, vol. 23, pp. 377-393.

Dai, Y. H.; Hager, W.; Schittkowsky, K.; Zhang, H. (2006): A cyclic Barzilai-Borwein method for unconstrained optimization. *IMA J. Numer. Anal.*, vol. 26, pp. 604-627.

Forsythe, G. E. (1968): On the asymptotic directions of the s-dimensional optimum gradient method. *Numer. Math.*, vol. 11, pp. 57-76.

Friedlander, A.; Martinez, J.; Molina, B.; Raydan, M. (1999): Gradient method with retard and generalizations. *SIAM J. Num. Anal.*, vol. 36, pp. 275-289.

Hon, Y.C.; Li, M. (2009): A discrepancy principle for the source points location in using the MFS for solving the BHCP. *Int. J. Comput. Meth.*, vol. 6, pp. 181-197.

Liu, C.-S. (2000): A Jordan algebra and dynamic system with associator as vector field. *Int. J. Non-Linear Mech.*, vol. 35, pp. 421-429.

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. (2011a): A revision of relaxed steepest descent method from the dynamics on an invariant manifold. *CMES: Computer Modeling in Engineering & Sciences*, vol. 80, pp. 57-86.

Liu, C.-S. (2011b): The method of fundamental solutions for solving the backward heat conduction problem with conditioning by a new post-conditioner. *Num. Heat Transfer, B: Fundamentals*, vol. 60, pp. 57-72.

Liu, C.-S. (2012a): Modifications of steepest descent method and conjugate gradient method against noise for ill-posed linear system. *Commun. Numer. Anal.*, vol. 2012, Article ID cna-00115, 24 pages.

Liu, C.-S. (2012b): The concept of best vector used to solve ill-posed linear inverse problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 83, pp. 499-525.

Liu, C.-S.; Atluri, S. N. (2011): An iterative method using an optimal descent vector, for solving an ill-conditioned system $\mathbf{Bx} = \mathbf{b}$, better and faster than the conjugate gradient method. *CMES: Computer Modeling in Engineering & Sciences*, vol. 80, pp. 275-298.

Nocedal, J.; Sartenar, A.; Zhu, C. (2002): On the behavior of the gradient norm

in the steepest descent method. *Comp. Optim. Appl.*, vol. 22, pp. 5-35.

Raydan, M.; Svaiter, B. F. (2002): Relaxed steepest descent and Cauchy-Barzilai-Borwein method. *Comp. Optim. Appl.*, vol. 21, pp. 155-167.