Copyright © 2011 Tech Science Press CMES, vol.76, no.2, pp.109-132, 2011

A Dynamical Tikhonov Regularization Method for Solving Nonlinear Ill-Posed Problems

Chein-Shan Liu[1] **and Chung-Lun Kuo**[2]

Abstract: The Tikhonov method is a famous technique for regularizing ill-posed systems. In this theory a regularization parameter α needs to be determined. Based-on an invariant-manifold defined in the space of $(\mathbf{x},t)$ and from the Tikhonov minimization functional, we can derive an optimal vector driven system of nonlinear ordinary differential equations (ODEs). In the Optimal Vector Driven Algorithm (OVDA), the optimal regularization parameter α_k is presented in the iterative solution of $\mathbf{x}$, which means that a *dynamical Tikhonov regularization method* is involved in the solution of nonlinear ill-posed problem. The OVDA is an extension of the Landweber-Scherzer iterative algorithm. Numerical examples of nonlinear ill-posed systems under noise are examined, revealing that the present OVDA has a good computational efficiency and accuracy.

Keywords: Ill-posed systems, Tikhonov regularization, Dynamical Tikhonov regularization, Optimal vector driven algorithm (OVDA), Landweber-Scherzer iterative algorithm

1 Introduction

In the scientific and engineering computation, it is a key matter to solve nonlinear operator equations numerically in a stable way. In this paper we propose a robust and easily-implemented *optimal vector driven method* to solve the following nonlinear ill-posed operator equations:

$$\hat{\mathbf{F}}(\mathbf{x}) = \mathbf{y}, \tag{1}$$

where $\mathbf{x}$ is an unknown, $\mathbf{y}$ is given data or observations, and $\hat{\mathbf{F}} : D(\hat{\mathbf{F}}) \subset X \mapsto Y$ is a nonlinear operator; X and Y are both real Hilbert spaces, endowed with inner

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

[2] Department of Systems Engineering and Naval Architecture, National Taiwan Ocean University, Keelung, Taiwan

products $(\cdot,\cdot)$ and norms $\|\cdot\|$, respectively. We assume that the observation $\mathbf{y}$ is attainable, and $\mathbf{x}$ is the solution of Eq. (1). Usually, the ill-posedness means that a minor noise in the datum $\mathbf{y}$ will bring out a drastic impact on the solution $\mathbf{x}$. Therefore, we may encounter the problem that the numerical solution of Eq. (1) may deviate from the exact one to a great extent, when the nonlinear operator $\hat{\mathbf{F}}$ is severely ill-posed.

Due to the ill-posedness of Eq. (1), we need to seek some effective methods to regularize it. The Landweber iteration [Landweber (1951)] is one of the most early methods being studied extensively in the literature for the numerical solution of nonlinear ill-posed problem. It can be viewed as the steepest descent method for the minimization of

$$\min_{\mathbf{x}\in D(\hat{\mathbf{F}})} \|\hat{\mathbf{F}}(\mathbf{x})-\mathbf{y}\|^2, \tag{2}$$

and can be written as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{\partial \hat{\mathbf{F}}(\mathbf{x}_k)}{\partial \mathbf{x}}[\hat{\mathbf{F}}(\mathbf{x}_k)-\mathbf{y}], \tag{3}$$

where $\mathbf{x}_k$ denotes the numerical value of $\mathbf{x}$ at the k-th iterative step. Scherzer (1995) has provided a convergence criterion by using the Landweber iteration to solve nonlinear problem. Then, Scherzer (1998) further enhanced the stability of the Landweber iteration by adding an extra term $-\alpha_k(\mathbf{x}_k-\zeta)$:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{\partial \hat{\mathbf{F}}(\mathbf{x}_k)}{\partial \mathbf{x}}[\hat{\mathbf{F}}(\mathbf{x}_k)-\mathbf{y}] - \alpha_k(\mathbf{x}_k-\zeta). \tag{4}$$

To account of the sensitivity to noise it is usually using a regularization method to solve the ill-posed problem [Engl (1987); Kunisch and Zou (1998); Wang and Xiao (2001); Xie and Zou (2002); Resmerita (2005)], where a suitable regularization parameter is used to depress the bias in the computed solution by a better balance of approximation error and propagated data error. For a large-dimensional system the main choice is using the iterative regularization algorithm, where a regularization parameter is represented by the number of iterations. The iterative method works if an early stopping criterion is used to prevent from reconstruction of noisy components in the approximated solutions.

Regularization can be employed to solve Eq. (1), when the operator $\hat{\mathbf{F}}$ is highly ill-posed. Hansen (1992) and Hansen and O'Leary (1993) have given an illuminative explanation that the Tikhonov regularization of ill-posed problem is a trade-off between the size of the regularized solution and the quality to fit the given data:

$$\min_{\mathbf{x}\in D(\hat{\mathbf{F}})} \psi(\mathbf{x}) = \min_{\mathbf{x}\in D(\hat{\mathbf{F}})} \left[\|\hat{\mathbf{F}}(\mathbf{x})-\mathbf{y}\|^2+\alpha\|\mathbf{x}\|^2\right]. \tag{5}$$

In this theory a regularization parameter α needs to be determined.

After the poineer work of Tikhonov and Arsenin (1977), there are a lot of techniques developed to determine a suitable regularization parameter. We name a few, the L-curve [Hansen (1992), Hansen and O'Leary (1993)], the discrepancy principles [Morozov (1984,1966); Engl (1987)], and the iterative technique [Gfrerer (1987); Kunisch and Zou (1998); Lukas (1998)].

2 An invariant manifold

Usually, we can seek a finite-dimensional realization of Eq. (1) as to be

$$\mathbf{F}(\mathbf{x}) = \hat{\mathbf{F}}(\mathbf{x}) - \mathbf{y} = \mathbf{0}, \tag{6}$$

where $\mathbf{x} \in \mathbb{R}^n$ is an unknown vector.

2.1 The evolution from a vector homotopy to a scalar homotopy

For solving the nonlinear algebraic equations (NAEs) in Eq. (6) the existent homotopy method, as initiated by Davidenko (1953), represents a way to enhance the convergence from a local convergence to a global convergence. The homotopy method is based on the construction of a vector function, $\mathbf{H}(\mathbf{x},\tau)$ which is called a vector homotopy function. The homotopy function serves the objective of continuously transforming a vector function $\mathbf{G}(\mathbf{x})$, whose zero points are easily detected, into $\mathbf{F}(\mathbf{x})$ by introducing a homotopy parameter τ. The homotopy parameter τ can be treated as a time-like fictitious variable, such that $\mathbf{H}(\mathbf{x},\tau=0) = \mathbf{G}(\mathbf{x})$ and $\mathbf{H}(\mathbf{x},\tau=1) = \mathbf{F}(\mathbf{x})$. Hence we can construct an ODEs system from keeping $\mathbf{H}$ to be zero vector, whose path is named the homotopic path, and which can trace the zeros of $\mathbf{G}(\mathbf{x})$ to our desired solutions of $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ while the parameter τ reaches to 1. Among the various vector homotopy functions that are generally used, the fixed point vector homotopy function, i.e. $\mathbf{G}(\mathbf{x}) = \mathbf{x} - \mathbf{x}_0$, and the Newton vector homotopy function, i.e. $\mathbf{G}(\mathbf{x}) = \mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{x}_0)$, are simple and powerful ones that can be successfully applied to several different problems. The Newton vector homotopy function can be written as

$$\mathbf{H}(\mathbf{x},\tau) = \tau\mathbf{F}(\mathbf{x}) + (1-\tau)[\mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{x}_0)], \tag{7}$$

where $\mathbf{x}_0$ is a given initial value of $\mathbf{x}$ and $\tau \in [0,1]$. However, the resultant ODEs based-on vector homotopy function require an evaluation of the inversion of the Jacobian matrix and the computational time is very expensive for its very slow convergence of the vector homotopy function method.

Liu, Yeih, Kuo and Atluri (2009) have developed a scalar homotopy function method by converting the vector equation of $\mathbf{F} = \mathbf{0}$ to a scalar equation $\|\mathbf{F}\| = 0$ through

$$\mathbf{F} = \mathbf{0} \leftrightarrow \|\mathbf{F}\|^2 = 0, \tag{8}$$

where $\|\mathbf{F}\|^2 := \sum_{i=1}^n F_i^2$. Then, Liu, Yeih, Kuo and Atluri (2009) developed a scalar homotopy function method with

$$h(\mathbf{x},\tau) = \frac{\tau}{2}\|\mathbf{F}(\mathbf{x})\|^2 + \frac{\tau-1}{2}\|\mathbf{x}-\mathbf{x}_0\|^2 = 0 \tag{9}$$

to derive the governing ODEs for $\mathbf{x}$.

The scalar homotopy method retains the merits of the vector homotopy method, such as the global convergence, but it does not involve the complicated computation of the inversion of the Jacobian matrix. The scalar homotopy method, however, needs a very small time step to reach the fictitious time, $\tau = 1$, which results in a slow convergence, in comparison with other methods. Later, Ku, Yeih and Liu (2010) modified Eq. (9) to the Newton scalar homotopy function by

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

where the function $Q(t) > 0$ satisfies $Q(0) = 1$, monotonically increasing, and $Q(\infty) = \infty$. According to this scalar homotopy function, Ku, Yeih and Liu (2010) could derive a faster convergent algorithm for solving the NAEs. As pointed out by Liu and Atluri (2011a), Eq. (10) is indeed providing a differentiable manifold, which confines the solution path being retained on that manifold.

2.2 A Newton scalar homotopic manifold

For the nonlinear algebraic equations (NAEs) in Eq. (6), as mentioned above we can formulate a scalar Newton homotopy function:

$$h(\mathbf{x},t) = \frac{1}{2}Q(t)\|\mathbf{F}(\mathbf{x})\|^2 - \frac{1}{2}\|\mathbf{F}(\mathbf{x}_0)\|^2 = 0, \tag{11}$$

where we let $\mathbf{x}$ be a function of a fictitious time-like variable t, and its initial value is $\mathbf{x}(0) = \mathbf{x}_0$.

We expect $h(\mathbf{x},t) = 0$ to be an invariant manifold in the space of $(\mathbf{x},t)$ for a dynamical system $h(\mathbf{x}(t),t) = 0$ to be specified further. When $Q > 0$, the manifold defined by Eq. (11) is continuous, and thus the following operation of differential which being carried out on the manifold makes sense. As a consequence of the "consistency condition", and taking the time differential of Eq. (11) with respect to t and considering $\mathbf{x} = \mathbf{x}(t)$, we have

$$\frac{1}{2}\dot{Q}(t)\|\mathbf{F}(\mathbf{x})\|^2 + Q(t)(\mathbf{B}^{\mathrm{T}}\mathbf{F}) \cdot \dot{\mathbf{x}} = 0. \tag{12}$$

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

$$\dot{\mathbf{x}} = \lambda \mathbf{u}, \tag{13}$$

where

$$\mathbf{u} = \alpha \mathbf{x} + \mathbf{B}^{\mathrm{T}}\mathbf{F} \tag{14}$$

is a suitable combination of the state vector $\mathbf{x}$ and the gradient vector $\mathbf{B}^{\mathrm{T}}\mathbf{F}$. In terms of the Tikhonov regularization, we indeed take the gradient of the functional ψ as defined in Eq. (5) to be the forcing term as presented by $\mathbf{u}$. Here, the parameter α can be viewed as a regularization parameter. Upon comparing with Eq. (4) the present driven vector $\mathbf{u}$ bears a certain similarity with the modification of the Landweber iteration made by Scherzer (1998). Through many studies, e.g., Hanke, Neubauer and Scherzer (1995), Neubauer (2000), Ramlau (1998), Jin (2001), Li, Han and Wang (2007), Wang, Han and Li (2008), it is known that although the Landweber iteration has a better stable property than other regularization methods, it is convergent very slowly. In the present paper we attempt to resolve this problem from a different approach.

Inserting Eq. (13) into Eq. (12) we can derive

$$\dot{\mathbf{x}} = -q(t)\frac{\|\mathbf{F}\|^2}{\mathbf{F}^{\mathrm{T}}\mathbf{v}}\mathbf{u}, \tag{15}$$

where

$$\mathbf{A} := \mathbf{B}\mathbf{B}^{\mathrm{T}}, \tag{16}$$

$$\mathbf{v} := \mathbf{B}\mathbf{u} = \mathbf{v}_1 + \alpha \mathbf{v}_2 = \mathbf{A}\mathbf{F} + \alpha \mathbf{B}\mathbf{x}, \tag{17}$$

$$q(t) := \frac{\dot{Q}(t)}{2Q(t)}. \tag{18}$$

Hence, in our algorithm if $Q(t)$ can be guaranteed to be a monotonically increasing function of t, we may have an absolutely convergent property in solving the NAEs in Eq. (6):

$$\|\mathbf{F}(\mathbf{x})\|^2 = \frac{C}{Q(t)}, \tag{19}$$

where

$$C = \|\mathbf{F}(\mathbf{x}_0)\|^2 \tag{20}$$

is determined by the initial value $\mathbf{x}_0$. We do not need to specify the function $Q(t)$ a priori, but $\sqrt{C/Q(t)}$ merely acts as a measure of the residual error of $\mathbf{F}$ in time. Hence, we impose in our algorithm that $Q(t) > 0$ is a monotonically increasing function of t. When t is quite large, the above equation will force the residual error $\|\mathbf{F}(\mathbf{x})\|$ to tend to zero, and meanwhile the solution of Eq. (6) is obtained approximately.

3 Dynamics of the present regularization method

3.1 Discretizing, yet keeping x *on the manifold*

Now we discretize the foregoing continuous time dynamics into a discrete time dynamics:

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \beta \frac{\|\mathbf{F}\|^2}{\mathbf{F}^{\mathrm{T}}\mathbf{v}}\mathbf{u}, \tag{21}$$

where

$$\beta = q(t)\Delta t \tag{22}$$

is the steplength. Eq. (21) is obtained from the ODEs in Eq. (15) by applying the Euler scheme.

In order to keep $\mathbf{x}$ on the manifold defined by Eq. (19) we can consider the evolution of $\mathbf{F}$ along the path $\mathbf{x}(t)$ by

$$\dot{\mathbf{F}} = \mathbf{B}\dot{\mathbf{x}} = -q(t)\frac{\|\mathbf{F}\|^2}{\mathbf{F}^{\mathrm{T}}\mathbf{v}}\mathbf{v}. \tag{23}$$

Similarly we also use the Euler scheme to integrate Eq. (23), obtaining

$$\mathbf{F}(t+\Delta t) = \mathbf{F}(t) - \beta \frac{\|\mathbf{F}\|^2}{\mathbf{F}^{\mathrm{T}}\mathbf{v}}\mathbf{v}, \tag{24}$$

Taking the square-norms of both the sides of Eq. (24) and using Eq. (19) we can obtain

$$\frac{C}{Q(t+\Delta t)} = \frac{C}{Q(t)} - 2\beta\frac{C}{Q(t)} + \beta^2 \frac{C}{Q(t)}\frac{\|\mathbf{F}\|^2}{(\mathbf{F}^{\mathrm{T}}\mathbf{v})^2}\|\mathbf{v}\|^2. \tag{25}$$

Thus we can derive the following scalar equation by dividing both the sides by $C/Q(t)$:

$$a_0\beta^2 - 2\beta + 1 - \frac{Q(t)}{Q(t+\Delta t)} = 0, \tag{26}$$

where

$$a_0 := \frac{\|\mathbf{F}\|^2\|\mathbf{v}\|^2}{(\mathbf{F}^{\mathrm{T}}\mathbf{v})^2}. \tag{27}$$

As a result $h(\mathbf{x},t) = 0$, $t \in \{0,1,2,\ldots\}$ remains to be an invariant manifold in the space of $(\mathbf{x},t)$ for the discrete time dynamical system $h(\mathbf{x}(t),t) = 0$, which will be explored further in the next four sections. Liu and Atluri (2011a) first derived the formula about a_0 in Eq. (27) for a gradient-vector driven dynamical system, i.e., $\alpha = 0$.

3.2 A trial discrete dynamics

Now we specify the discrete time dynamics $h(\mathbf{x}(t),t) = 0, t \in \{0,1,2,\ldots\}$, through specifying the discrete time dynamics of $Q(t)$, $t \in \{0,1,2,\ldots\}$. Note that the discrete time dynamics is an iterative dynamics, which amounts to an iterative algorithm.

We first try the Euler scheme:

$$Q(t+\Delta t) = Q(t) + \dot{Q}(t)\Delta t. \tag{28}$$

Then from Eq. (18) we have

$$\beta = q(t)\Delta t = \frac{1}{2}[R(t) - 1]. \tag{29}$$

where the ratio $R(t)$ is defined by

$$R(t) = \frac{Q(t+\Delta t)}{Q(t)}. \tag{30}$$

As a requirement of $\dot{Q}(t) > 0$, we need $R(t) > 1$.

Thus, through some manipulations, Eq. (26) becomes

$$a_0R^3(t) - (2a_0+4)R^2(t) + (a_0+8)R(t) - 4 = 0, \tag{31}$$

which can be further written as

$$[R(t)-1]^2[a_0R(t)-4]=0. \tag{32}$$

Because $R=1$ is a double-root and does not satisfy $R>1$, we have to take another solution:

$$R(t)=\frac{4}{a_0}=\frac{4(\mathbf{F}^{\mathrm{T}}\mathbf{v})^2}{\|\mathbf{F}\|^2\|\mathbf{v}\|^2}. \tag{33}$$

By using Eq. (29), Eq. (21) can now be written as

$$\mathbf{x}(t+\Delta t)=\mathbf{x}(t)-\frac{1}{2}[R(t)-1]\frac{\|\mathbf{F}\|^2}{\mathbf{F}^{\mathrm{T}}\mathbf{v}}\mathbf{u}. \tag{34}$$

Notice, however, that this algorithm has an unfortunate fate that when the iterated a_0 starts to approach to 4 before it grows up to a large value, the algorithm stagnates at a point which is not necessarily a solution. We will avoid to follow this kind of dynamics by developing a better dynamics as follows. This indicates that the vector-driven algorithm will face this fate to lose its dynamic force if we insist the iterative orbit as being located on the manifold defined by Eq. (19). This point has been first explored in-depth by Liu and Atluri (2011a) for a gradient-driven algorithm.

3.3 A better discrete dynamics

Let

$$s=\frac{Q(t)}{Q(t+\Delta t)}=\frac{\|\mathbf{F}(\mathbf{x}(t+\Delta t))\|^2}{\|\mathbf{F}(\mathbf{x}(t))\|^2}, \tag{35}$$

which is an important quantity to assess the convergence property of numerical algorithm for solving the NAEs. A convergent algorithm must guarantee $s<1$ step-by-step, such that the residual norm $\|\mathbf{F}\|$ can tend to zero gradually, and thus the solution $\mathbf{x}$ of $\mathbf{F}(\mathbf{x})=\mathbf{0}$ can be obtained.

From Eqs. (26) and (35) it follows that

$$a_0\beta^2-2\beta+1-s=0, \tag{36}$$

where

$$a_0=\frac{\|\mathbf{F}\|^2\|\mathbf{v}\|^2}{(\mathbf{F}^{\mathrm{T}}\mathbf{v})^2}\geq 1, \tag{37}$$

by using the Cauchy-Schwarz inequality:

$$\mathbf{F}^{\mathrm{T}}\mathbf{v} \leq \|\mathbf{F}\|\|\mathbf{v}\|.$$

From Eq. (36), the solution of β is found to be

$$\beta = \frac{1-\sqrt{1-(1-s)a_0}}{a_0}, \quad \text{if } 1-(1-s)a_0 \geq 0. \tag{38}$$

Let

$$1-(1-s)a_0 = \gamma^2 \geq 0, \tag{39}$$

$$s = 1-\frac{1-\gamma^2}{a_0}. \tag{40}$$

Thus, from Eq. (38) it follows that

$$\beta = \frac{1-\gamma}{a_0}, \tag{41}$$

and from Eqs. (21) and (27) we can obtain the following algorithm:

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \eta \frac{\mathbf{F}^{\mathrm{T}}\mathbf{v}}{\|\mathbf{v}\|^2}\mathbf{u}, \tag{42}$$

where

$$\eta = 1-\gamma. \tag{43}$$

Here $0 \leq \gamma < 1$ is a parameter. Later, in the numerical examples we will explain that γ plays a major role for the *bifurcation* of discrete dynamics. Under the above condition we can prove that the new algorithm satisfies

$$\frac{\|\mathbf{F}(t+\Delta t)\|}{\|\mathbf{F}(t)\|} = \sqrt{s} < 1, \tag{44}$$

which means that the residual error is absolutely decreased.

It is interesting that in the above algorithm Δt is no longer required. Furthermore, the property in Eq. (44) is very important, since it guarantees the new algorithm to be absolutely convergent to the true solution.

3.4 Optimization of α

In the algorithm (42) we do not specify how to choose the parameter α, which is appeared in $\mathbf{u} = \alpha\mathbf{x} + \mathbf{B}^{\mathrm{T}}\mathbf{F}$ and $\mathbf{v} = \alpha\mathbf{B}\mathbf{x} + \mathbf{A}\mathbf{F}$. One way is that α is chosen by the user. However, this strategy does not always be effective. Alternatively, we can determine a suitable α such that s defined in Eq. (40) be minimized with respect to α, because a smaller s will lead to a faster convergence as shown in Eq. (44). The idea of optimizing α was first developed by Liu and Atluri (2011b) for other algorithm.

Thus, by inserting Eq. (37) for a_0 into Eq. (40) we can write s as to be

$$s = 1 - \frac{(1-\gamma^2)(\mathbf{F}\cdot\mathbf{v})^2}{\|\mathbf{F}\|^2\|\mathbf{v}\|^2}, \tag{45}$$

where $\mathbf{v}$ as defined by Eq. (17) includes a parameter α. Let $\partial s/\partial\alpha = 0$, and through some algebraic operations we can solve α by

$$\alpha = \frac{(\mathbf{v}_1\cdot\mathbf{F})(\mathbf{v}_1\cdot\mathbf{v}_2) - (\mathbf{v}_2\cdot\mathbf{F})\|\mathbf{v}_1\|^2}{(\mathbf{v}_2\cdot\mathbf{F})(\mathbf{v}_1\cdot\mathbf{v}_2) - (\mathbf{v}_1\cdot\mathbf{F})\|\mathbf{v}_2\|^2}. \tag{46}$$

Remark 1: For the usual three-dimensional vectors $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c} \in \mathbb{R}^3$, the following formula is famous:

$$\mathbf{a}\times(\mathbf{b}\times\mathbf{c}) = (\mathbf{a}\cdot\mathbf{c})\mathbf{b} - (\mathbf{a}\cdot\mathbf{b})\mathbf{c}. \tag{47}$$

Liu (2000a) has developed a Jordan algebra by extending the above formula to vectors in n-dimension:

$$[\mathbf{a},\mathbf{b},\mathbf{c}] = (\mathbf{a}\cdot\mathbf{b})\mathbf{c} - (\mathbf{c}\cdot\mathbf{b})\mathbf{a}, \;\; \mathbf{a},\mathbf{b},\mathbf{c} \in \mathbb{R}^n. \tag{48}$$

Thus α in Eq. (46) can be expressed in a neater form via

$$\alpha = \frac{[\mathbf{v}_1,\mathbf{v}_2,\mathbf{F}]\cdot\mathbf{v}_1}{[\mathbf{v}_2,\mathbf{v}_1,\mathbf{F}]\cdot\mathbf{v}_2}. \tag{49}$$

The above parameter α can be called the optimal α, because it brings us a new strategy to select the best orientation $\mathbf{u} = \alpha\mathbf{x} + \mathbf{B}^{\mathrm{T}}\mathbf{F}$ and the best steplength $\mathbf{F}\cdot[\alpha\mathbf{B}\mathbf{x} + \mathbf{A}\mathbf{F}]/\|\alpha\mathbf{B}\mathbf{x} + \mathbf{A}\mathbf{F}\|^2$ to search the solution of NAEs. Furthermore, we have an explicit form to implement α into the numerical program, and thus it is very time-saving for finding the unknown solution vector $\mathbf{x}$.

3.5 An optimal vector driven algorithm

Since the fictitious time variable is now discrete, $t \in \{0,1,2,\ldots\}$, we let $\mathbf{x}_k$ denote the numerical value of $\mathbf{x}$ at the k-th step. Thus, we arrive at a purely iterative algorithm by Eqs. (42) and (43):

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1-\gamma)\frac{\mathbf{F}_k^{\mathrm{T}}\mathbf{v}_k}{\|\mathbf{v}_k\|^2}\mathbf{u}_k. \tag{50}$$

Consequently, the following optimal vector driven algorithm (OVDA) can be derived:

Step 1: Select $0 \le \gamma < 1$, and give an initial $\mathbf{x}_0$.

Step 2: For $k = 0,1,2\ldots$ we repeat the following calculations:

$$\mathbf{v}_1^k = \mathbf{A}_k\mathbf{F}_k, \tag{51}$$

$$\mathbf{v}_2^k = \mathbf{B}_k\mathbf{x}_k, \tag{52}$$

$$\alpha_k = \frac{[\mathbf{v}_1^k, \mathbf{v}_2^k, \mathbf{F}_k]\cdot\mathbf{v}_1^k}{[\mathbf{v}_2^k, \mathbf{v}_1^k, \mathbf{F}_k]\cdot\mathbf{v}_2^k}, \tag{53}$$

$$\mathbf{u}_k = \alpha_k\mathbf{x}_k + \mathbf{B}_k^{\mathrm{T}}\mathbf{F}_k, \tag{54}$$

$$\mathbf{v}_k = \mathbf{B}_k\mathbf{u}_k, \tag{55}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1-\gamma)\frac{\mathbf{F}_k\cdot\mathbf{v}_k}{\|\mathbf{v}_k\|^2}\mathbf{u}_k. \tag{56}$$

If $\mathbf{x}_{k+1}$ converges according to a given stopping criterion with $\|\mathbf{F}_{k+1}\| < \varepsilon$, then stop; otherwise, go to **Step 2**.

In summary, we have derived a thoroughly novel algorithm for solving the NAEs. While the parameter γ is chosen by the user for a problem-dependent manner, the parameter α_k is exactly given by Eq. (53). Indeed these two parameters play the different roles of *bifurcation parameter* and *optimization parameter*, respectively. Up to here we have successfully derived a *drastical novel* algorithm from the Tikhonov regularization method, which is based on the bifurcation and optimal parameter of α. Indeed, without the help from the formula (37) we cannot perform such a powerful result. The influences of these two parameters are analyzed below through numerical examples.

Because the Tikhonov parameter α derived here is a dynamical one, which is evolving with the solution path, we may also call the present method as being a *Dynamical Tikhonov Regularization Method*, which can be used to solve the nonlinear ill-posed problem. The concept of *Dynamical Tikhonov Regularization Parameter* is the first time developed in this article, which is quite different from those determined by other methods, like as, L-curve, Morozov discrepancy princile and iterative solution; correspondingly, they are all of the static ones, whose determination needs a prior or a posterior information about the nonlinear operator equation.

4 Numerical examples

4.1 Example 1

In this example, the following nonlinear Fredholm integral equation of the first-kind is considered:

$$\int_0^1 x(s)x(t)dt = c_1\cos(c_2 s) =: h(s), \ \ c_1 > 0, \tag{57}$$

where c_1 and c_2 are constants fixed to be $c_1 = 1$ and $c_2 = 3$. Two exact solutions exist: $x(s) = \pm\cos(3s)\sqrt{3/\sin 3}$ [Polyanin and Manzhirov (2008)].

Let us discretize the interval of $[0,1]$ into $n-1$ subintervals by noting $\Delta t = 1/n$ and $\Delta s = 1/n$. Let $x_j := x(t_j)$ be a numerical value of x at a grid point t_j, and let $h_i = h(s_i)$, where $t_j = (j-1)\Delta t$ and $s_i = (i-1)\Delta s$. Through a trapezoidal rule, Eq. (57) can be discretized into

$$x_i\left[\frac{\Delta t}{2}x_1 + \Delta t\sum_{j=2}^{m_1} x_j + \frac{\Delta t}{2}x_n\right] = h_i, \ i = 1,\ldots,n, \tag{58}$$

which are nonlinear algebraic equations denoted by:

$$\hat{\mathbf{F}}(\mathbf{x}) = \mathbf{b}_1. \tag{59}$$

Here, $\mathbf{b}_1 = (h_1,\ldots,h_n)^{\mathrm{T}}$, and $\mathbf{x} = (x_1,\ldots,x_n)^{\mathrm{T}}$ is the vector of unknowns. The data h_j are corrupted by a random relative noise, such that:

$$\hat{h}_j = h_j[1+\sigma R(j)]. \tag{60}$$

Under a fixed convergence criterion $\varepsilon = 10^{-10}$ and starting from an initial condition $x_i = 0.5$, we use the following parameters $m = 50$ and $\gamma = 0$ in the OVDA to calculate the numerical solution under a noise with $\sigma = 0$. Through only 15 steps it is convergent to the true solution with a maximum error being 6.92×10^{-4} as shown by the solid lines for showing the residual error, a_0 and the numerical error, respectively, in Figs. 1(a), 1(b) and 1(c).

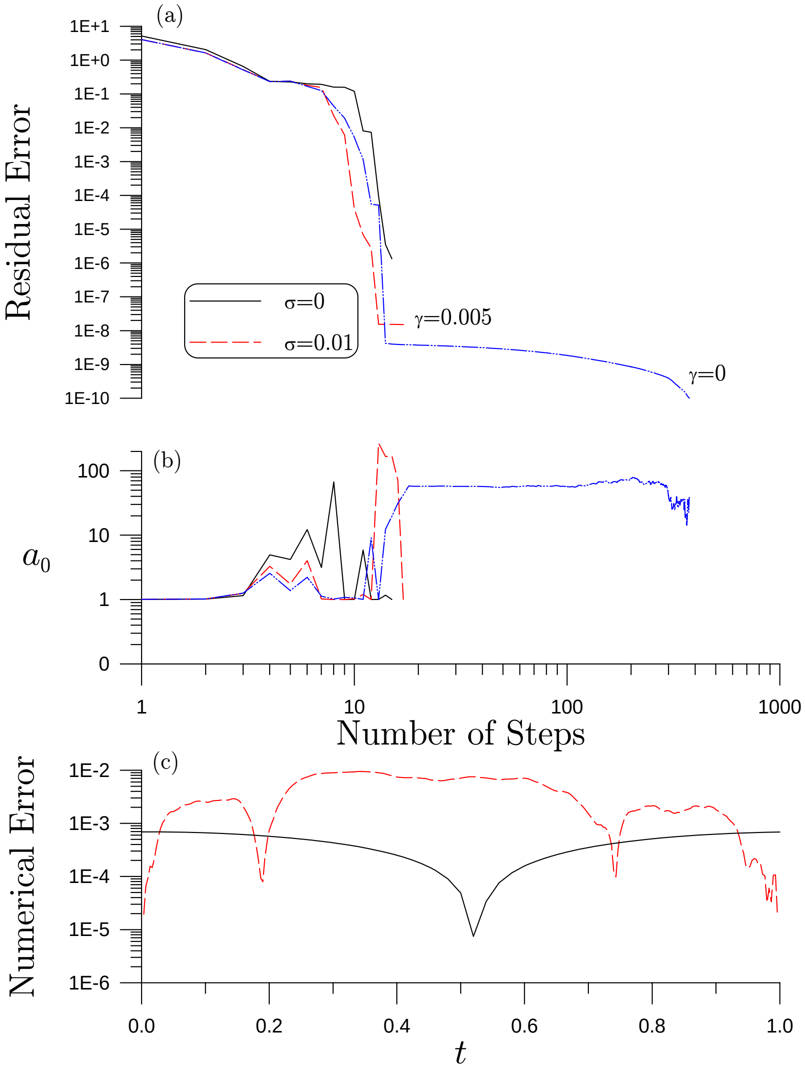

Figure 1: By applying the OVDA to example 1: (a) comparing the residual errors, (b) a_0, and (c) numerical errors.

When $\sigma = 0.01$ is added in the data, we reduce m to 30, and use the following parameter $\gamma = 0.005$ in the OVDA to calculate the numerical solution. Through only 17 steps it is convergent with a maximum error being 5.39×10^{-2} as shown by the dashed lines for showing the residual error, a_0 and the numerical error, respectively, in Figs. 1(a), 1(b) and 1(c). Then we use $\gamma = 0$ in the OVDA to calculate the numerical solution. Through 376 steps it is convergent with a maximum error being 5.39×10^{-2} as shown by the dashed-dotted lines for showing the residual error and a_0, respectively, in Figs. 1(a) and 1(b).

It is obvious that when we use $\gamma = 0$ in the OVDA it is much slow convergence than that using $\gamma = 0.005$. Now, we explain the parameter γ appeared in Eq. (50). In Fig. 1(b) we compare a_0 obtained by OVDA with $\gamma = 0$ and $\gamma = 0.005$. From Fig. 1(b) it can be seen that for the case with $\gamma = 0$, the values of a_0 tend to a constant and keep unchanged. By Eq. (37) it means that there exists an attracting set for the iterative orbit of $\mathbf{x}$ described by the following manifold:

$$\frac{\|\mathbf{F}\|^2\|\mathbf{v}\|^2}{(\mathbf{F}\cdot\mathbf{v})^2} = \text{Constant}. \tag{61}$$

Upon the iterative orbit is approached to this manifold, it is slowly to reduce the residual error as shown in Fig. 1(a) by the dashed-dotted line for the case of OVDA with $\gamma = 0$. Conversely, for the case $\gamma = 0.005$, a_0 is no more tending to a constant as shown in Fig. 1(b). Because the iterative orbit does not be attracted by an attracting manifold, the residual error as shown in Fig. 1(a) by the dashed line for the case of OVDA with $\gamma = 0.005$ can reduce very fast. Thus we can observe that when γ varies from zero to a positive value, the iterative dynamics given by Eq. (50) undergoes a Hopf bifurcation, like the ODEs behavior observed by Liu (2000b, 2007). The original stable manifold existent for $\gamma = 0$ now becomes a ghost manifold for $\gamma = 0.005$, and thus the iterative orbit generated from the present algorithm with the case $\gamma = 0.005$ does not be attracted by that manifold again.

4.2 Example 2

In this example, the following nonlinear Fredholm integral equation of the first-kind is solved:

$$\int_0^2 [x(s) - x(t)]^2 dt = 2s^4 + \frac{16}{3}s^2 + \frac{32}{5}, \quad s \in [0,2]. \tag{62}$$

The exact solution is $x(t) = t^2$.

Under a fixed convergence criterion $\varepsilon = 10^{-4}$ and starting from an initial condition $x_i = 1$, we use the following parameters $m = 50$ and $\gamma = 0.1$ in the OVDA to calculate the numerical solution under a noise with $\sigma = 0$. Through only 33 steps it is convergent to the true solution with a maximum error being 6×10^{-4} as shown by the solid lines for showing the residual error, a_0 and the numerical error, respectively, in Figs. 2(a), 2(b) and 2(c).

When $\sigma = 0.01$ is added in the data, we use the following parameter $\gamma = 0.1$ in the OVDA to calculate the numerical solution. Through only 37 steps it is convergent with a maximum error being 2.45×10^{-2} as shown by the dashed lines for showing the residual error, a_0 and the numerical error, respectively, in Figs. 2(a), 2(b) and 2(c). In Fig. 3 we compare the optimal α obtained from these two cases.

4.3 Example 3

We consider a nonlinear backward heat conduction equation:

$$u_t = k(x)u_{xx} + k'(x)u_x + u^2 + H(x,t), \tag{63}$$

$$k(x) = (x-3)^2, \;\; H(x,t) = -7(x-3)^2 e^{-t} - (x-3)^4 e^{-2t}, \tag{64}$$

with a closed-form solution being $u(x,t) = (x-3)^2 e^{-t}$. The boundary conditions and a final time condition are available from the above solution. It is known that the nonlinear backward heat conduction problem is highly ill-posed. In order to test the stability of OVDA we also add a relative noise in the final time data with an intensity $\sigma = 0.01$.

By applying the new algorithm to solve the above equation in the domain of $0 \leq x \leq 1$ and $0 \leq t \leq 1$ we fix $\Delta x = 1/n_1$ and $\Delta t = 1/n_2$, where $n_1 = 14$ and $n_2 = 10$ are numbers of nodal points used in a standard finite difference approximation of Eq. (63):

$$k(x_i)\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2} + k'(x_i)\frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} + u_{i,j}^2 + H(x_i,t_j) - \frac{u_{i,j+1} - u_{i,j}}{\Delta t} = 0. \tag{65}$$

Because a_0 defined in Eq. (27) is a very important factor of our new algorithm we show it in Fig. 4(b) for the OVDA with $\gamma = 0.15$, while the residual error is shown in Fig. 4(a), and α is shown in Fig. 4(c). We let the OVDA run 2000 steps, which attains an accurate solution with a maximum relative error being 1.785×10^{-3} as shown in Fig. 5. The present result is better than that computed by Liu and Atluri (2011b).

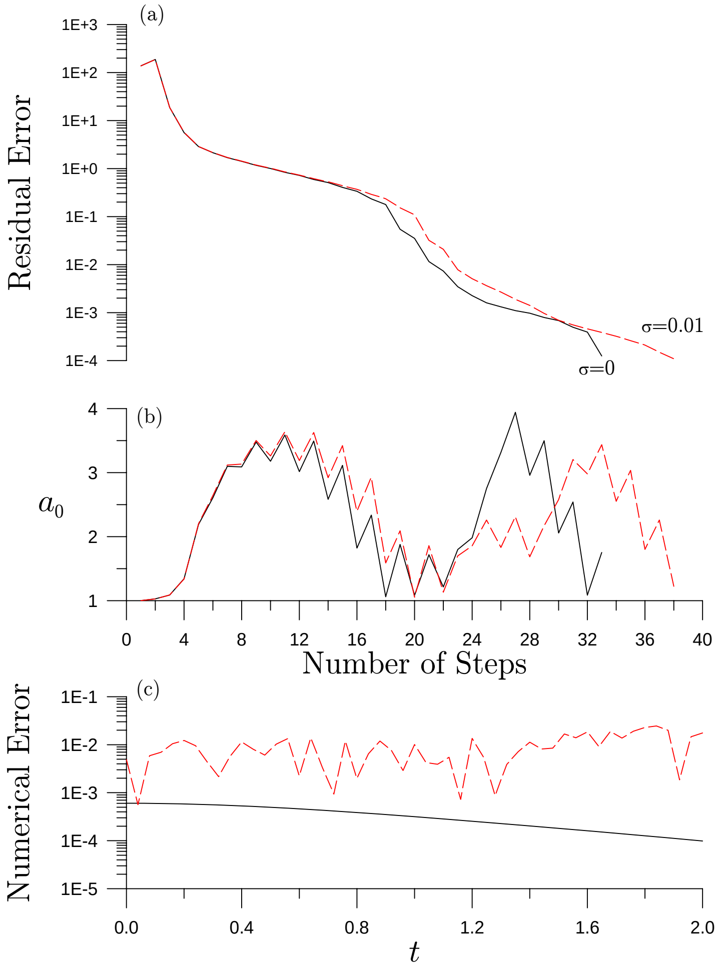

Figure 2: By applying the OVDA to example 2: (a) comparing the residual errors, (b) a_0, and (c) numerical errors.

4.4 Example 4

We consider a nonlinear Calderón inverse problem:

$$\nabla \cdot [\sigma(x,y)\nabla u(x,y)] = 0, \quad (x,y) \in \Omega, \tag{66}$$

$$u(\rho,\theta) = f(x,y), \quad (x,y) \in \partial\Omega, \tag{67}$$

$$\sigma(x,y)u_n(x,y) = g(x,y), \quad (x,y) \in \partial\Omega. \tag{68}$$

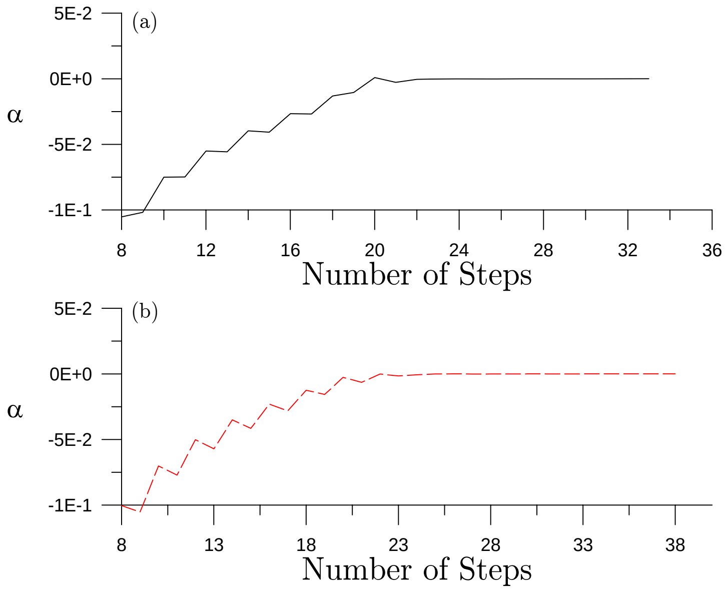

Figure 3: By applying the OVDA to example 2 comparing α for (a) no noise, and (b) with a noise.

Both $u(x,y)$ and the conductivity function $\sigma(x,y)$ are unknown, but the boundary conditions are provided and over-specified.

The following exact solutions are taken from [Farcas, Elliott, Ingham and Lesnic (2004)]:

$$u(x,y) = \frac{xy}{1+0.2x+0.4y+0.15xy}, \tag{69}$$

$$\sigma(x,y) = (1+0.2x+0.4y+0.15xy)^2, \tag{70}$$

and the boundary shape of the computation domain is described in the polar coordinates by

$$\rho(\theta) = \sqrt{\cos 2\theta + \sqrt{1.1 - \sin^2 2\theta}}. \tag{71}$$

Thus according to the suggestion by Liu and Atluri (2009) we can employ the following modified polynomial expansion method to express the solution of $u(x,y)$

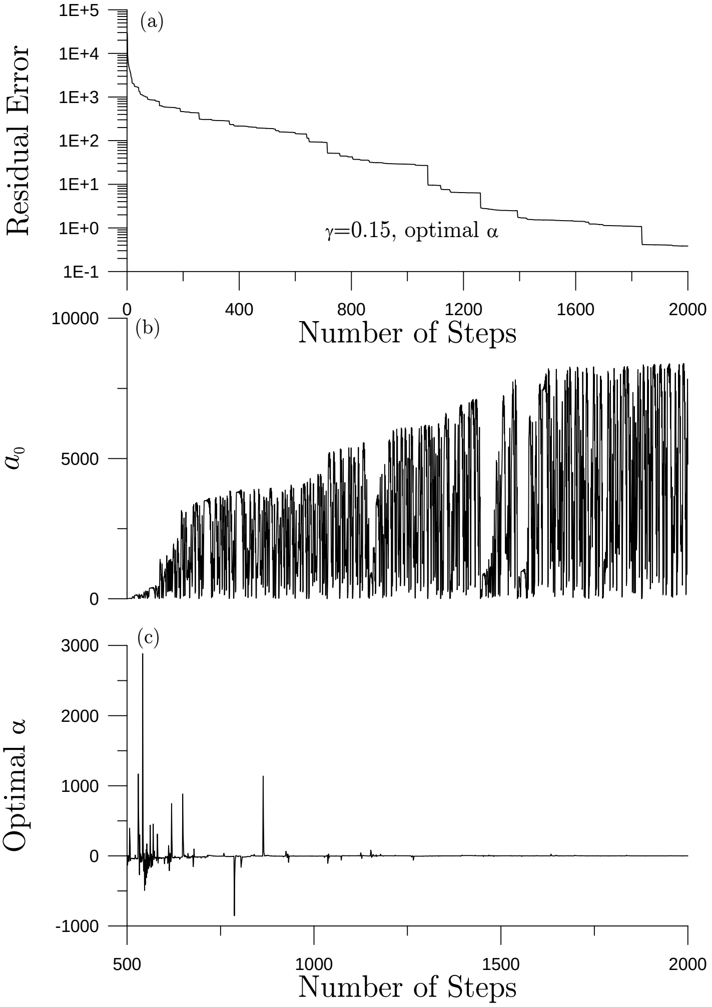

Figure 4: By applying the OVDA to example 3 showing (a) the residual error, (b) a_0, and (c) optimal α.

and $\sigma(x,y)$:

$$u(x,y)=\sum_{i=0}^{m}\sum_{j=0}^{i}a_{ji}\left(\frac{x}{X_0}\right)^{j}\left(\frac{y}{Y_0}\right)^{i-j}, \tag{72}$$

$$\sigma(x,y)=\sum_{i=0}^{m}\sum_{j=0}^{i}b_{ji}\left(\frac{x}{X_0}\right)^{j}\left(\frac{y}{Y_0}\right)^{i-j}, \tag{73}$$

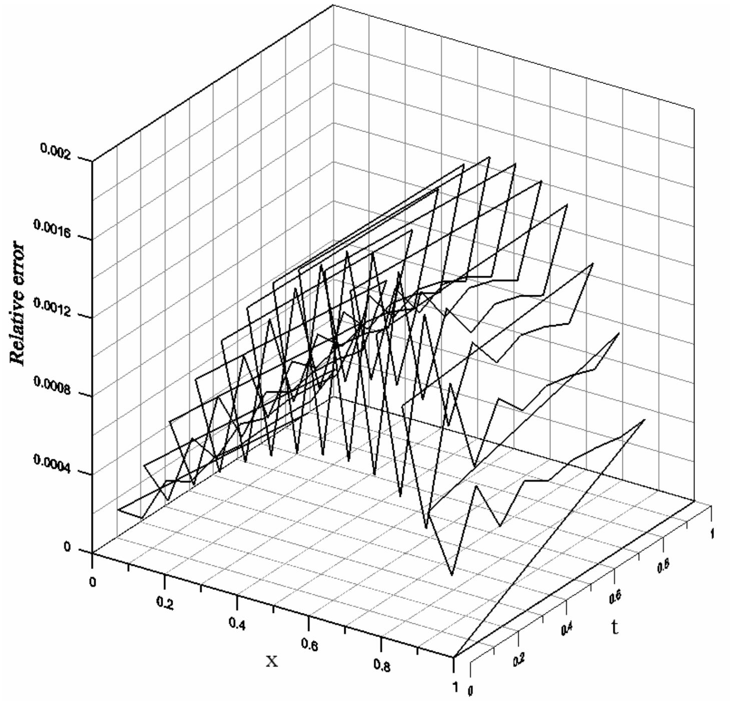

Figure 5: By applying the OVDA to example 3 showing the relative error of u over the plane of (x,t).

where m is the highest-order of the polynomial, X_0 and Y_0 are the characteristic lengths, and a_{ji} and b_{ji} are unknown coefficients. Inserting Eqs. (72) and (73) into Eqs. (66)-(68) we can obtain a nonlinear system of algebraic equations to solve a_{ji} and b_{ji}.

We apply the OVDA to solve the resultant over-determined nonlinear system of algebraic equations with parameters $m = 5$, $X_0 = 1.5$ and $Y_0 = 0.6$. We collocate 288 interior points to satisfy the governing equation and 100 boundary points to satisfy the boundary conditions. The initial guesses of a_{ji} and b_{ji} are all zero except $b_{00} = 1$. For this example we choose $\gamma = 0.15$ and consider a 5% random noise being added to the boundary conditions. We let OVDA run 10000 steps, and show a_0, s, the optimal α and the residual error in Figs. 6(a)-(d). The final regularization parameter is $\alpha = -8.2851 \times 10^{-6}$. We also compare the exact solution and the numerical solution of u and σ in Fig. 7. Even under a large noise with an intensity 5%, the results are also accurate and stable. The maximum relative error of σ is 6.82%.

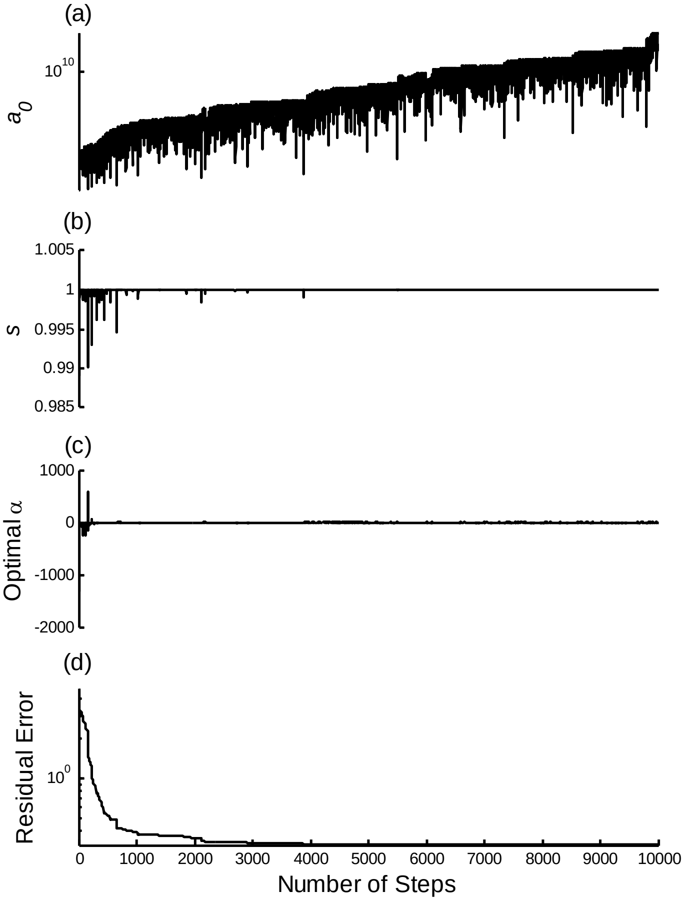

Figure 6: By applying the OVDA to example 4 showing (a) a_0, (b) s, (c) optimal α, and (d) the residual error.

5 Conclusions

A residual-norm based and optimization based algorithm, namely an Optimal Vector Driven Algorithm (OVDA), where $\dot{\mathbf{x}} = \lambda \mathbf{u}(\alpha)$ with $\mathbf{u} = \alpha \mathbf{x} + \mathbf{B}^{\mathrm{T}}\mathbf{F}$ the driving vector involving an optimal value of α and $B_{ij} = \partial F_i / \partial x_j$, was established in this paper to solve $\mathbf{F} = \mathbf{0}$. In the iteration process the driven vector $\mathbf{u}_k = \alpha_k \mathbf{x}_k + \mathbf{B}_k^{\mathrm{T}}\mathbf{F}_k$ is a dynamical Tikhonov regularization vector, which is an optimal vector involving an

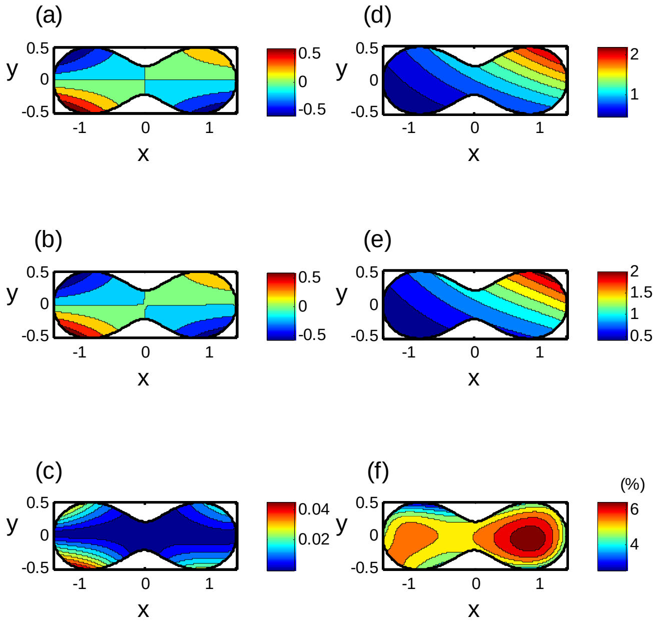

Figure 7: By applying the OVDA to example 4 showing (a)-(c) the exact solution, present result, and the absolute error of u, respectively; and (d)-(f) the exact solution, the present result, and the relative error of σ, respectively.

optimal parameter α_k. It is interesting that when the solutions were found, the optimal parameters α_k as shown by all numerical examples were also tending to zero; hence, the driven vector $\mathbf{u}_k$ was lost its dynamical force. The parameter γ is a very important factor, which is a bifurcation parameter, enabling us to switch the slow convergence to a new situation that the residual-error is quickly decreased. The optimal parameter α_k was derived exactly in terms of the Jordan algebra, and thus it is very time saving to implement the optimization technique into the numerical program. We have proved that the present algorithm is convergent automatically, and it is easy to implement, and without calculating the inversions of the Jacobian matrices. Several numerical examples of nonlinear ill-posed equations, including a nonlinear backward heat conduction problem, and a nonlinear Calderón inverse problem, were solved by the OVDA with good performance. We can conclude that this is the first time that a *Dynamical Tikhonov Regularization Method* is well-developed to solve the nonlinear ill-posed problem.

Acknowledgement: Taiwan's National Science Council project NSC-100-2221-E-002-165-MY3 granted to the first author is highly appreciated.

References

Davidenko, D. (1953): On a new method of numerically integrating a system of nonlinear equations. *Doklady Akad. Nauk SSSR*, vol. 88, pp. 601-604.

Engl, H. W. (1987): Discrepancy principles for Tikhonov regularization of ill-posed problems leading to optimal convergence rates. *J. Optim. Theory. Appl.*, vol. 52, pp. 209-215.

Farcas A.; Elliott L.; Ingham D. B.; Lesnic D. (2004): An inverse dual reciprocity method for hydraulic conductivity identification in steady ground flow. *Adv. Water Resou.*, vol. 27, pp. 223-235.

Fan, C.M.; Liu, C.-S.; Yeih, Y. C. and Chan, H. F. (2010): The Scalar Homotopy Method for Solving Non-Linear Obstacle Problem *CMC: Computers, Materials, & Continua*, Vol. 15, No. 1, pp. 67-86.

Gfrerer, H. (1987): An a-posteriori parameter choices for ordinary and iterated Tikhonov regularization of ill-posed problems leading to optimal convergence rates. *Math. Comp.*, vol. 49, pp. 507-522.

Hanke, M.; Neubauer, A.; Scherzer, O. (1995): A convergence analysis of the Landweber iteration for nonlinear ill-posed problems. *Numer. Math.*, vol. 72, pp. 21-37.

Hansen, P. C. (1992): Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Rev.*, vol. 34, pp. 561-580.

Hansen, P. C.; O'Leary, D. P. (1993): The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM J. Sci. Comput.*, vol. 14, pp. 1487-1503.

Jin, Q. (2001): A discrete scheme of Landweber iteration for solving nonlinear ill-posed problems. *J. Math. Anal. Appl.*, vol. 253, pp. 187-203.

Ku, C.-Y.; Yeih, W.; Liu, C.-S. (2010): Solving non-linear algebraic equations by a scalar Newton-homotopy continuation method. *Int. J. Nonlinear Sci. Numer. Simul.*, vol. 11, pp. 435-450.

Kunisch, K.; Zou, J. (1998): Iterative choices of regularization parameters in linear inverse problems. *Inverse Problems*, vol. 14, pp. 1247-1264.

Landweber, L. (1951): An iteration formula for Fredholm integral equations of the first kind. *Amer. J. Math.*, vol. 73, pp. 615-624.

Li, L.; Han, B.; Wang, W. (2007): R-K type Landweber method for nonlinear ill-posed problems. *J. Comp. Appl. Math.*, vol. 206, pp. 341-357.

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

Liu, C.-S. (2000b): Intermittent transition to quasiperiodicity demonstrated via a circular differential equation. *Int. J. Non-Linear Mech.*, vol. 35, pp. 931-946.

Liu, C.-S. (2007): A study of type I intermittency of a circular differential equation under a discontinuous right-hand side. *J. Math. Anal. Appl.*, vol. 331, pp. 547-566.

Liu, C.-S. (2010a): The Fictitious Time Integration Method to Solve the Space- and Time-Fractional Burgers Equations *CMC: Computers, Materials, & Continua*, Vol. 15, No. 3, pp. 221-240.

Liu, C.-S. (2010b): A Lie-Group Adaptive Method for Imaging a Space-Dependent Rigidity Coefficient in an Inverse Scattering Problem of Wave Propagation *CMC: Computers, Materials, & Continua*, Vol. 18, No. 1, pp. 1-20.

Liu, C.-S.; Atluri, S. N. (2009): A highly accurate technique for interpolations using very high-order polynomials, and its applications to some ill-posed linear problems. *CMES: Computer Modeling in Engineering and Science*, vol. 43, pp. 253-276.

Liu, C.-S.; Atluri, S. N. (2011a): Simple "residual-norm" based algorithms, for the solution of a large system of non-linear algebraic equations, which converge faster than the Newton's method. *CMES: Computer Modeling in Engineering and Science*, vol. 71, pp. 279-304.

Liu, C.-S.; Atluri, S. N. (2011b): A method for solving a system of nonlinear algebraic equations, $\mathbf{F}(\mathbf{x}) = \mathbf{0}$, using the system of ODEs with an optimum α in $\dot{\mathbf{x}} = \lambda[\alpha\mathbf{F} + (1-\alpha)\mathbf{B}^{\mathrm{T}}\mathbf{F}]$; $B_{ij} = \partial F_i/\partial x_j$. *CMES: Computer Modeling in Engineering and Science*, vol. 73, pp. 395-431.

Liu, C.-S.; Yeih, W.; Kuo, C.-L.; Atluri, S. N. (2009): A scalar homotopy method for solving an over/under-determined system of non-linear algebraic equations. *CMES: Computer Modeling in Engineering and Science*, vol. 53, pp. 47-71.

Lukas, M. A. (1998): Comparison of parameter choice methods for regularization with discrete noisy data. *Inverse Problems*, vol. 14, pp. 161-184.

Morozov, V. A. (1966): On regularization of ill-posed problems and selection of regularization parameter. *J. Comp. Math. Phys.*, vol. 6, pp. 170-175.

Morozov, V. A. (1984): Methods for Solving Incorrectly Posed Problems. Springer, New York.

Neubauer, A. (2000): On Landweber iteration for nonlinear ill-posed problem in Hilbert scales. *Numer. Math.*, vol. 85, pp. 309-328.

Polyanin, A. D.; Manzhirov, A. V. (2008): Handbook of Integral Equations. 2nd edition, Chapman & Hall/CRC, Taylor & Francis Group.

Ramlau, R. A. (1998): A modified Landweber method for inverse problems. *Numer. Funct. Anal. Optim.*, vol. 20, pp. 79-98.

Resmerita, E. (2005): Regularization of ill-posed problems in Banach spaces: convergence rates. *Inverse Problems*, vol. 21, pp. 1303-1314.

Scherzer, O. (1995): Convergence criteria of iterative methods based on Landweber iteration for solving nonlinear problems. *J. Math. Anal. Appl.*, vol. 194, pp. 911-934.

Scherzer, O. (1998): A modified Landweber iteration for solving parameter estimation problems. *Appl. Math. Optim.*, vol. 38, pp. 45-68.

Tikhonov, A. N.; Arsenin, V. Y. (1977): Solutions of Ill-Posed Problems. John-Wiley & Sons, New York.

Wang, W.; Han, B.; Li, L. (2008): A Runge-Kutta type modified Landweber method for nonlinear ill-posed operator equations. *J. Comp. Appl. Math.*, vol. 212, pp. 457-468.

Wang, Y.; Xiao, T. (2001): Fast realization algorithms for determining regularization parameters in linear inverse problems. *Inverse Problems*, vol. 17, pp. 281-291.

Xie, J.; Zou, J. (2002): An improved model function method for choosing regularization parameters in linear inverse problems. *Inverse Problems*, vol. 18, pp. 631-643.