Acta Appl Math (2013) 123:285–307
DOI 10.1007/s10440-012-9766-3

A Dynamical Tikhonov Regularization for Solving Ill-posed Linear Algebraic Systems

Chein-Shan Liu

Received: 28 March 2011 / Accepted: 26 May 2012 / Published online: 13 June 2012

Abstract The Tikhonov method is a famous technique for regularizing ill-posed linear problems, wherein a regularization parameter needs to be determined. This article, based on an invariant-manifold method, presents an adaptive Tikhonov method to solve ill-posed linear algebraic problems. The new method consists in building a numerical minimizing vector sequence that remains on an invariant manifold, and then the Tikhonov parameter can be optimally computed at each iteration by minimizing a proper merit function. In the *optimal vector method* (OVM) three concepts of optimal vector, slow manifold and Hopf bifurcation are introduced. Numerical illustrations on well known ill-posed linear problems point out the computational efficiency and accuracy of the present OVM as compared with classical ones.

Keywords Ill-posed linear system · Tikhonov regularization · Adaptive Tikhonov method · Dynamical Tikhonov regularization · Steepest descent method (SDM) · Conjugate gradient method (CGM) · Optimal vector method (OVM) · Barzilai-Borwein method (BBM)

Mathematics Subject Classification (2010) 65F10 · 65F22

1 Introduction

Consider the following linear system:

$$\mathbf{B}\mathbf{x} = \mathbf{b}_1, \tag{1}$$

where $\mathbf{B} \in \mathbb{R}^{n\times n}$ is a given matrix, which might be unsymmetric, and $\mathbf{x} \in \mathbb{R}^n$ is an unknown vector. The above equation is sometimes obtained via an n-dimensional discretization of a bounded linear operator equation under an noisy input. We only look for a generalized solution $\mathbf{x} = \mathbf{B}^\dagger \mathbf{b}_1$, where $\mathbf{B}^\dagger$ is a pseudo-inverse of $\mathbf{B}$ in the Penrose sense. When $\mathbf{B}$ is

C.-S. Liu (✉)
Department of Civil Engineering, National Taiwan University, Taipei, Taiwan
e-mail: liucs@ntu.edu.tw

severely ill-conditioned and the data are corrupted by noise, we may encounter the problem that the numerical solution of Eq. (1) deviates from the exact one to a great extent. If we only know perturbed input data $\mathbf{b}_1^\delta \in \mathbb{R}^n$ with $\|\mathbf{b}_1 - \mathbf{b}_1^\delta\| \leq \delta$, and if the problem is ill-posed, i.e., the range $R(\mathbf{B})$ is not closed or equivalently $\mathbf{B}^\dagger$ is unbounded, we have to solve the system (1) by a *regularization method.*

A measure of the ill-posedness of Eq. (1) can be performed by using the condition number of $\mathbf{B}$ [53]:

$$\mathrm{cond}(\mathbf{B}) = \|\mathbf{B}\|_F \left\|\mathbf{B}^{-1}\right\|_F, \tag{2}$$

where $\|\mathbf{B}\|_F$ denotes the Frobenius norm of $\mathbf{B}$.

For every matrix norm $\|\bullet\|$ we have $\rho(\mathbf{B}) \leq \|\mathbf{B}\|$, where $\rho(\mathbf{B})$ is a radius of the spectrum of $\mathbf{B}$. The Householder theorem states that for every $\epsilon > 0$ and every matrix $\mathbf{B}$, there exists a matrix norm $\|\mathbf{B}\|$ depending on $\mathbf{B}$ and ϵ such that $\|\mathbf{B}\| \leq \rho(\mathbf{B}) + \epsilon$. Anyway, the spectral condition number $\rho(\mathbf{B})\rho(\mathbf{B}^{-1})$ can be used as an estimation of the condition number of $\mathbf{B}$ by

$$\mathrm{cond}(\mathbf{B}) = \frac{\max_{\sigma(\mathbf{B})} |\lambda|}{\min_{\sigma(\mathbf{B})} |\lambda|}, \tag{3}$$

where $\sigma(\mathbf{B})$ is the collection of all the eigenvalues of $\mathbf{B}$. Turning back to the Frobenius norm we have

$$\|\mathbf{B}\|_F \leq \sqrt{n} \max_{\sigma(\mathbf{B})} |\lambda|. \tag{4}$$

In particular, for the symmetric case $\rho(\mathbf{B})\rho(\mathbf{B}^{-1}) = \|\mathbf{B}\|_2 \|\mathbf{B}^{-1}\|_2$. Roughly speaking, the numerical solution of Eq. (1) may lose the accuracy of k decimal points when $\mathrm{cond}(\mathbf{B}) = 10^k$.

To remedy the sensitivity to noise it is often using a regularization method to solve this sort ill-posed problem [22, 52, 58, 59], 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. Previously, the author and his co-workers have developed several methods to solve the ill-posed linear problems: using the fictitious time integration method as a filter for ill-posed linear system [31], a modified polynomial expansion method [32], the non-standard group-preserving scheme [36], a vector regularization method [37], the preconditioners and postconditioners generated from a transformation matrix, obtained by Liu et al. [39] for solving the Laplace equation with a multiple-scale Trefftz basis functions, the optimal iterative algorithms [27, 35], as well as an optimally scaled vector regularization method [28].

It is known that the iterative methods for solving the system of algebraic equations can be derived from the discretization of a certain ODEs system [2, 3, 30]. Particularly, some descent methods can be interpreted as the discretizations of gradient flows [16]. For a large scale system the main choice is using an iterative regularization algorithm, where a regularization parameter is presented by the number of iterations. The iterative method works if an early stopping criterion is used to prevent the reconstruction of noisy components in the approximate solutions.

In this article we propose a robust and easily-implemented *optimal vector method* (OVM) to solve the ill-posed linear equations system. An adaptive Tikhonov method is derived, of which the regularization parameter is adapted step-by-step and is optimized by maximizing the convergence rates in the iterative solution process. This article is arranged as follows. In Sect. 2 some backgrounds of the Tikhonov regularization techniques: the L-curve, the discrepancy principles, the generalized cross-validation technique and the iteration technique are briefly introduced. For the purpose of comparison, the classical methods of the steepest descent and the conjugate gradient, and the two-point Barzilai-Borwein algorithm are mentioned. The main contribution of this article about an invariant manifold method is presented in Sect. 3, which motivates the present study, and leads to an adaptive Tikhonov method with an optimization regularization parameter to accelerate the convergence speed. Some well known numerical examples of ill-posed linear problems are given in Sect. 4, where the comparisons with some classical methods are addressed. Finally, the conclusions are drawn in Sect. 5.

2 Mathematical Preliminaries

2.1 The Tikhonov Regularization

In Eq. (1), when $\mathbf{B}$ is highly ill-conditioned, one can use a regularization method to solve this system. Hansen [14] and Hansen and O'Leary [15] have given an illuminative explanation that the Tikhonov regularization of ill-posed linear problem is taking a trade-off between the size of the regularized solution and the quality to fit the given data:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \psi(\mathbf{x}) = \min_{\mathbf{x}\in\mathbb{R}^n}\big[\|\mathbf{B}\mathbf{x}-\mathbf{b}_1\|^2+\alpha\|\mathbf{x}\|^2\big]. \tag{5}$$

In this regularization theory a parameter α needs to be determined [17, 55]. The above minimization is equivalent to solve the following Tikhonov regularization equation:

$$\big[\mathbf{B}^{\mathrm{T}}\mathbf{B}+\alpha\mathbf{I}_n\big]\mathbf{x}=\mathbf{b}, \tag{6}$$

where $\mathbf{b}=\mathbf{B}^{\mathrm{T}}\mathbf{b}_1$, and the superscript T signifies the transpose.

If $\alpha=0$ we recover to the following normal linear equations system:

$$\mathbf{A}\mathbf{x}=\mathbf{b}, \tag{7}$$

where the coefficient matrix $\mathbf{A}:=\mathbf{B}^{\mathrm{T}}\mathbf{B}\in\mathbb{R}^{n\times n}$ is a positive matrix.

After the pioneering work of Tikhonov and Arsenin [55], a number of techniques have been developed to determine a suitable regularization parameter: the L-curve [14, 15], the discrepancy principles [8, 44, 45], and the iterative technique [12, 22, 41], just to name a few.

Here we briefly review four existent selection techniques of the regularization parameter α in Eqs. (5) and (6). They differ in the amount of a priori or a posteriori information required and in the decision criteria.

(i) *The discrepancy principle* [44] says that the regularization parameter should be chosen such that the norm of the residual vector corresponding to the regularized solution $\mathbf{x}_{\mathrm{reg}}$ is $\tau\delta$:

$$\|\mathbf{B}\mathbf{x}_{\mathrm{reg}}-\mathbf{b}_1\|=\tau\delta, \tag{8}$$

where $\tau>1$ is some predetermined real number. Note that $\mathbf{x}_{\mathrm{reg}}\longrightarrow\mathbf{x}_{\mathrm{true}}$ if $\delta\longrightarrow 0$.

(ii) *The generalized cross-validation* [13] does not depend on a priori information about the variance of noise δ. One finds the parameter α that minimizes the following GCV functional:

$$G(\alpha)=\frac{\|(\mathbf{I}_n-\mathbf{B}[\mathbf{B}^{\mathrm{T}}\mathbf{B}+\alpha\mathbf{I}_n]^{-1}\mathbf{B}^{\mathrm{T}})\mathbf{b}_1\|^2}{[\mathrm{Trace}(\mathbf{I}_n-\mathbf{B}[\mathbf{B}^{\mathrm{T}}\mathbf{B}+\alpha\mathbf{I}_n]^{-1}\mathbf{B}^{\mathrm{T}})]^2}. \tag{9}$$

(iii) For *the L-curve*, the plot of the norm of $\mathbf{x}_{\mathrm{reg}}$ versus the corresponding residual norm for each of a set of the regularization parameter values, was introduced by Hansen [14]. The best regularization parameter should lie on the corner of the L-curve, since for values higher than this, the residual norm increases without reducing the norm of $\mathbf{x}_{\mathrm{reg}}$ much, while for values smaller than this, the norm $\|\mathbf{x}_{\mathrm{reg}}\|$ increases rapidly without much decreasing the residual norm.

(iv) *The iteration technique* was introduced by Engl [9] and Gfrerer [12]. Basically, they constructed an iterative sequence:

$$\left[\mathbf{B}^{\mathrm{T}}\mathbf{B}+\alpha\mathbf{I}_n\right]\mathbf{x}^{j}_{\alpha,\delta}=\mathbf{b}+\alpha\mathbf{x}^{j-1}_{\alpha,\delta},\quad j=1,\dots,m, \tag{10}$$

starting from the initial $\mathbf{x}^0_{\alpha,\delta}=\mathbf{0}$, and then inserted the convergent solution of Eq. (10) into Eq. (8) to iteratively solve an implicit and nonlinear algebraic equation for finding the best α.

As pointed out by Kilmer and O'Leary [20] many of the above algorithms are needlessly complicated. In this article we introduce a dynamical Tikhonov regularization method for solving the ill-posed linear problem, and the regularization parameter α can be generated automatically by basing on an optimization vector embedded in an invariant manifold [38].

2.2 The Steepest Descent and Conjugate Gradient Methods

Solving Eq. (7) by the steepest descent method [19, 46] is equivalent to solving the following minimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n}\varphi(\mathbf{x})=\min_{\mathbf{x}\in\mathbb{R}^n}\left[\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}-\mathbf{b}^{\mathrm{T}}\mathbf{x}\right]. \tag{11}$$

Thus one can derive the following steepest descent method (SDM):

(i) Give an initial $\mathbf{x}_0$.
(ii) For $k=0,1,2,\dots$, we repeat the following computations:

$$\mathbf{r}_k=\mathbf{A}\mathbf{x}_k-\mathbf{b}, \tag{12}$$

$$\eta_k=\frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^{\mathrm{T}}\mathbf{A}\mathbf{r}_k}, \tag{13}$$

$$\mathbf{x}_{k+1}=\mathbf{x}_k-\eta_k\mathbf{r}_k. \tag{14}$$

If $\mathbf{x}_{k+1}$ converges, satisfying a given stopping criterion $\|\mathbf{r}_{k+1}\|<\varepsilon$, then stop; otherwise, go to step (ii).

For the SDM the residual vector $\mathbf{r}_k$ is the steepest descent direction of the functional φ at the point $\mathbf{x}_k$. But when $\|\mathbf{r}_k\|$ is rather small the computed $\mathbf{r}_k$ may deviate from the actual steepest descent direction to a great extent due to a round-off error of computing machine, which usually leads to the numerical instability of SDM.

An improvement of the SDM is the conjugate gradient method (CGM), which enhances the searching direction of the minimum by imposing the orthogonality of the residual vectors at each iterative step [19]. The algorithm of the CGM is summarized as follows:

(i) Give an initial $\mathbf{x}_0$ and then compute $\mathbf{r}_0=\mathbf{A}\mathbf{x}_0-\mathbf{b}$ and set $\mathbf{p}_1=\mathbf{r}_0$.
(ii) For $k=1,2,\dots$, we repeat the following computations:

$$\eta_k=\frac{\|\mathbf{r}_{k-1}\|^2}{\mathbf{p}_k^{\mathrm{T}}\mathbf{A}\mathbf{p}_k}, \tag{15}$$

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \eta_k \mathbf{p}_k, \tag{16}$$

$$\mathbf{r}_k = \mathbf{A}\mathbf{x}_k - \mathbf{b}, \tag{17}$$

$$\beta_k = \frac{\|\mathbf{r}_k\|^2}{\|\mathbf{r}_{k-1}\|^2}, \tag{18}$$

$$\mathbf{p}_{k+1} = \beta_k \mathbf{p}_k + \mathbf{r}_k. \tag{19}$$

If $\mathbf{x}_k$ converges according to a given stopping criterion $\|\mathbf{r}_k\| < \varepsilon$, then stop; otherwise, go to step (ii).

It is well known that the SDM and CGM are effective for solving the well-posed linear systems. For ill-posed linear systems they are however vulnerable to noise disturbance with numerical instability. Recently, Liu [26] has developed a relaxed steepest descent method from the viewpoint of an invariant manifold in terms of a squared residual norm; furthermore, Liu and Atluri [35] proposed an optimal descent vector method for solving the ill-conditioned linear systems. Both methods are better than the SDM and the CGM against noise, which is randomly imposed on the ill-conditioned linear system.

2.3 The Steplengths in the Steepest Descent Method

In addition to the CGM, several modifications to the SDM have been made. These modifications stimulated a new interest in the SDM because it is recognized that the gradient vector itself is not a bad choice of the solution search direction, but rather that the steplength originally dictated by the SDM is to blame for the slow convergence behavior. Barzilai and Borwein [1] were the first, who presented a new choice of steplength through two-point stepsize. The algorithm of the Barzilai-Borwein method (BBM) is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{(\Delta\mathbf{r}_{k-1})^{\mathrm{T}} \Delta\mathbf{x}_{k-1}}{\|\Delta\mathbf{r}_{k-1}\|^2} \mathbf{r}_k, \tag{20}$$

where $\Delta\mathbf{r}_{k-1} = \mathbf{r}_k - \mathbf{r}_{k-1}$ and $\Delta\mathbf{x}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}$ with initial guesses $\mathbf{r}_0 = \mathbf{0}$ and $\mathbf{x}_0 = \mathbf{0}$. Although it does not guarantee the continuous descent of the values of the minimum functional, the BBM was able to produce a substantial improvement of the convergence speed in a certain test. The results of BBM have motivated many researches on the SDM [4–6, 10, 11, 49–51, 60]. Chehab and Laminie [3] have modified the BBM together with a shooting method, and they found that the coupled method of shooting/BBM converges faster than the BBM.

In this article we will approach this steplength problem from a quite different point of view of the *invariant-manifold* and *bifurcation*, and propose a new strategy to modify the steplength. Moreover, instead of the static regularization parameter α, which as previously demonstrated is frequently determined by the L-curve, discrepancy principle, etc., here we propose a *dynamical regularization parameter* in the adaptive Tikhonov regularization method, which is the best trade-off between residual vector and state vector in the ODEs for describing the evolution of state vector. We will demonstrate that the newly developed optimal vector method (OVM) outperforms better than the SDM, the CGM, as well as the BBM for solving the ill-posed linear problems.

3 An Invariant Manifold Method

3.1 Motivation

For solving the nonlinear algebraic equations $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ the existent vector homotopy method, as initiated by Davidenko [7], represents a way to enhance the convergence from a local one to a global one. The vector homotopy method is based on the construction of a vector homotopy function $\mathbf{H}(\mathbf{x}, \tau)$, which serves an 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 fictitious time-like 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 by keeping $\mathbf{H}$ to be a 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 often 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 simpler ones that can be successfully applied to solve the nonlinear problems. The Newton vector homotopy function can be written as

$$\mathbf{H}(\mathbf{x}, \tau) = \tau\mathbf{F}(\mathbf{x}) + (1 - \tau)\big[\mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{x}_0)\big], \tag{21}$$

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

Liu et al. [40] have developed a scalar homotopy function method by converting the vector equation $\mathbf{F} = \mathbf{0}$ into a scalar equation $\|\mathbf{F}\| = 0$ through

$$\mathbf{F} = \mathbf{0} \leftrightarrow \|\mathbf{F}\|^2 = 0, \tag{22}$$

where $\|\mathbf{F}\|^2 := \sum_{i=1}^n F_i^2$. Then, Liu et al. [40] developed a scalar homotopy function method with

$$h(\mathbf{x}, \tau) = \frac{\tau}{2}\big\|\mathbf{F}(\mathbf{x})\big\|^2 + \frac{\tau - 1}{2}\|\mathbf{x} - \mathbf{x}_0\|^2 = 0 \tag{23}$$

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

The scalar homotopy method retains the major merit of the vector homotopy method as to be global convergence, and 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 in order to keep $\mathbf{x}$ on the manifold (23), such that in order to reach $\tau = 1$, it results in a slow convergence. Later, Ku et al. [21] modified Eq. (23) to the Newton scalar homotopy function by

$$h(\mathbf{x}, t) = \frac{Q(t)}{2}\big\|\mathbf{F}(\mathbf{x})\big\|^2 - \frac{1}{2}\big\|\mathbf{F}(\mathbf{x}_0)\big\|^2 = 0, \tag{24}$$

where the function $Q(t) > 0$, satisfies $Q(0) = 1$, is monotonically increasing, and $Q(\infty) = \infty$. According to this scalar homotopy function, Ku et al. [21] could derive a faster convergent algorithm for solving the nonlinear algebraic equations. As pointed out by Liu and Atluri [33], Eq. (24) is indeed providing a differentiable manifold

$$Q(t)\big\|\mathbf{F}(\mathbf{x})\big\|^2 = C, \tag{25}$$

where C is a constant, to confine the solution path being retained on that manifold.

Stemming from this idea of embedding the ODEs on an invariant manifold, now we introduce an invariant manifold pertaining to Eq. (7). From Eqs. (11) and (7) it is easy to prove that the minimum is

$$\min_{\mathbf{x}\in\mathbb{R}^n} \varphi(\mathbf{x}) = \varphi\big(\mathbf{x}^*\big) = -\frac{1}{2}\mathbf{x}^{*\mathrm{T}}\mathbf{A}\mathbf{x}^* < 0, \tag{26}$$

where $\mathbf{A}$ is a positive definite matrix and $\mathbf{x}^*$ is a non-zero solution of Eq. (7).

We take

$$\phi(\mathbf{x}) := \varphi(\mathbf{x}) + c_0 = \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x} - \mathbf{b}^{\mathrm{T}}\mathbf{x} + c_0, \tag{27}$$

where c_0 is a constant such that $\phi(\mathbf{x}) > 0$. Obviously, the minima of $\phi(\mathbf{x})$ and $\varphi(\mathbf{x})$ occur at the same place $\mathbf{x} = \mathbf{x}^*$.

There are several regularization methods available for solving Eq. (7) when $\mathbf{A}$ is ill-conditioned. In this article we consider a dynamical regularization method for Eq. (7) by investigating the evolutional behavior of $\mathbf{x}$ governed by the ordinary differential equations (ODEs) defined on an invariant manifold formed from $\phi(\mathbf{x})$:

$$h(\mathbf{x}, t) := Q(t)\phi(\mathbf{x}) = C, \tag{28}$$

where it is required that $Q(t) > 0$ and the constant $C = Q(0)\phi(\mathbf{x}(0)) > 0$. This equation is similar to Eq. (25). Here, we do not need to specify the function $Q(t)$ *a priori*, for $C/Q(t)$ merely acting as a measure of the decreasing of ϕ in time. We let $Q(0) = 1$, and C is determined by the initial condition $\mathbf{x}(0) = \mathbf{x}_0$ with

$$C = \phi(\mathbf{x}_0). \tag{29}$$

In our algorithm if $Q(t)$ can be guaranteed to be an increasing function of t, we may have an absolutely convergent property in finding the minimum of ϕ through the following equation:

$$\phi(t) = \frac{C}{Q(t)}. \tag{30}$$

When t is large enough the above equation can enforce the functional ϕ to tend to its minimum. We expect $h(\mathbf{x}, t) = C$ to be an invariant manifold in the space of $(\mathbf{x}, t)$ for a dynamical system $h(\mathbf{x}(t), t) = C$ to be specified further. Hence, for the requirement of the consistency condition $dh/dt = 0$, we have

$$\dot{Q}(t)\phi(\mathbf{x}) + Q(t)(\mathbf{A}\mathbf{x} - \mathbf{b}) \cdot \dot{\mathbf{x}} = 0, \tag{31}$$

by taking the differentiation of Eq. (28) with respect to t and considering $\mathbf{x} = \mathbf{x}(t)$. The overdot indicates the differential with respect to t. Note that $\dot{Q}(t) > 0$.

3.2 A Vector Driving Method

We suppose that $\mathbf{x}$ is governed by a vector-driving-flow:

$$\dot{\mathbf{x}} = -\lambda\mathbf{u}, \tag{32}$$

where λ is to be determined. Inserting Eq. (32) into Eq. (31) we can solve λ by

$$\lambda = \frac{q(t)\phi}{\mathbf{r}\cdot\mathbf{u}}, \tag{33}$$

where

$$q(t) := \frac{\dot{Q}(t)}{Q(t)}, \tag{34}$$

and

$$\mathbf{r} := \mathbf{A}\mathbf{x} - \mathbf{b} \tag{35}$$

is a residual vector.

According to the Tikhonov regularization method in Eq. (5) we further suppose that the driving vector $\mathbf{u}$ is given by the gradient vector of ψ, that is,

$$\mathbf{u} = \mathbf{r} + \alpha\mathbf{x}, \tag{36}$$

where α is a parameter to be determined below by an optimal equation. Here, we assert that the driving vector $\mathbf{u}$ is a suitable combination of the residual vector $\mathbf{r}$ and the state vector $\mathbf{x}$ being weighted by α.

Thus, inserting Eq. (33) into Eq. (32) we can obtain an ODEs system for $\mathbf{x}$ defined by

$$\dot{\mathbf{x}} = -q(t)\frac{\phi}{\mathbf{r}\cdot\mathbf{u}}\mathbf{u}. \tag{37}$$

It deserves to notice that the dynamical system in Eq. (37) is time-dependent and nonlinear, which is quite different from that of the so-called Dynamical Systems Method (DSM), which was previously developed by Ramm [47, 48], Hoang and Ramm [17, 18], and Sweilam et al. [54]. The ODEs system appeared in the DSM for solving the ill-posed linear problems is time-independent and linear.

3.3 Discretizing, Yet Keeping x on the Manifold

Now we discretize the foregoing continuous time dynamics in Eq. (37) into a discretized time dynamics. By applying the Euler method to Eq. (37) we can obtain the following algorithm:

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \beta\frac{\phi}{\mathbf{r}\cdot\mathbf{u}}\mathbf{u}, \tag{38}$$

where

$$\beta = q(t)\Delta t, \tag{39}$$

and Δt is a time increment.

In order to keep $\mathbf{x}$ on the invariant manifold defined by Eq. (30) with ϕ defined by Eq. (27) we can insert the above $\mathbf{x}(t+\Delta t)$ into

$$\frac{1}{2}\mathbf{x}^{\mathrm{T}}(t+\Delta t)\mathbf{A}\mathbf{x}(t+\Delta t) - \mathbf{b}^{\mathrm{T}}\mathbf{x}(t+\Delta t) + c_0 = \frac{C}{Q(t+\Delta t)}, \tag{40}$$

and obtain

$$\frac{C}{Q(t+\Delta t)} - c_0 = \frac{1}{2}\mathbf{x}^{\mathrm{T}}(t)\mathbf{A}\mathbf{x}(t) - \mathbf{b}^{\mathrm{T}}\mathbf{x}(t) + \beta\phi\frac{[\mathbf{b}-\mathbf{A}\mathbf{x}(t)]^{\mathrm{T}}\mathbf{u}}{\mathbf{r}\cdot\mathbf{u}} + \beta^2\phi^2\frac{\mathbf{u}^{\mathrm{T}}\mathbf{A}\mathbf{u}}{2(\mathbf{r}\cdot\mathbf{u})^2}. \tag{41}$$

Thus by Eqs. (35), (30) and (27) and through some manipulations we can derive the following scalar equation:

$$\frac{1}{2}a_0\beta^2 - \beta + 1 = \frac{Q(t)}{Q(t+\Delta t)}, \tag{42}$$

where

$$a_0 := \frac{\phi \mathbf{u}^{\mathrm{T}}\mathbf{A}\mathbf{u}}{(\mathbf{r}\cdot\mathbf{u})^2} \ge \frac{1}{2}. \tag{43}$$

This inequality can be achieved by choosing a suitable value of c_0.

As a result $h(\mathbf{x}, t) = C$, $t \in \{0, 1, 2, \ldots\}$, remains to be an invariant manifold in the space of $(\mathbf{x}, t)$ for the discrete time dynamical system $h(\mathbf{x}(t), t) = C$, which will be further explored in the following three subsections.

3.4 A Worser Dynamics

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

We first try the Euler scheme,

$$Q(t+\Delta t) = Q(t) + \dot{Q}(t)\Delta t, \tag{44}$$

and by Eqs. (34) and (39) we can derive

$$\frac{Q(t)}{Q(t+\Delta t)} = \frac{1}{1+\beta}. \tag{45}$$

Inserting it into Eq. (42) we come to a cubic equation for β:

$$a_0\beta^2(1+\beta) - 2\beta(1+\beta) + 2(1+\beta) = 2, \tag{46}$$

where $\beta = 0$ is a double root, which leads to $\dot{Q} = 0$, contradicting to $\dot{Q} > 0$. However, it allows another non-zero solution of β:

$$\beta = \frac{2}{a_0} - 1. \tag{47}$$

Inserting the above β into Eq. (38) we can obtain

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \left[\frac{2}{a_0} - 1\right]\frac{\phi}{\mathbf{r}\cdot\mathbf{u}}\mathbf{u}. \tag{48}$$

Notice, however, that this algorithm has an unfortunate fate in that when the iterative value of a_0 starts to approach to 2 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 and develop a better dynamics below.

3.5 A Better Dynamics

The above observation hints us that we must abandon the solution of β provided by Eq. (47); otherwise, we only have an algorithm which cannot work continuously, upon it stagnates at one point.

Let $s = Q(t)/Q(t+\Delta t)$. By Eq. (42) we can derive

$$\frac{1}{2}a_0\beta^2 - \beta + 1 - s = 0, \tag{49}$$

and we can take the solution of β to be

$$\beta = \frac{1-\sqrt{1-2(1-s)a_0}}{a_0}, \quad \text{if } 1-2(1-s)a_0 \geq 0. \tag{50}$$

Let

$$1-2(1-s)a_0 = \gamma^2 \geq 0, \tag{51}$$

$$s = 1 - \frac{1-\gamma^2}{2a_0}, \tag{52}$$

such that the condition $1-2(1-s)a_0 \geq 0$ in Eq. (50) is automatically satisfied, and thus we have

$$\beta = \frac{1-\gamma}{a_0}. \tag{53}$$

Here $0 \leq \gamma < 1$ is a parameter.

So far, we have derived a better discrete dynamics by inserting Eq. (53) for β into Eq. (38), which is given by

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \frac{1-\gamma}{a_0}\frac{\phi}{\mathbf{r}\cdot\mathbf{u}}\mathbf{u}. \tag{54}$$

It deserves to note the difference between Eqs. (54) and (48). Furthermore, by inserting Eq. (43) for a_0 into Eq. (54) we have

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - (1-\gamma)\frac{\mathbf{r}\cdot\mathbf{u}}{\mathbf{u}\cdot\mathbf{A}\mathbf{u}}\mathbf{u}; \tag{55}$$

it is interesting that in the above algorithm ϕ disappears.

The main mathematical property of the above algorithm is that the functional ϕ is stepwisely decreasing, i.e., $\phi(t+\Delta t) < \phi(t)$. From Eqs. (43) and (52) it follows that

$$s < 1, \tag{56}$$

due to $0 \leq \gamma < 1$ and $a_0 \geq 1/2$. Then, by $s = Q(t)/Q(t+\Delta t)$ and Eq. (30) we can prove

$$\frac{\phi(t+\Delta t)}{\phi(t)} = s < 1, \tag{57}$$

which means that along the iterative path the functional ϕ is monotonically decreasing to its minimum. Such that we can find the solution $\mathbf{x}$ of Eq. (1) with a convergence rate:

$$\frac{1}{s} = \frac{\phi(t)}{\phi(t+\Delta t)} > 1 \quad \text{(Convergence rate)}, \tag{58}$$

which can be made as larger as possible with the following optimal technique.

3.6 Optimization of α

In the algorithm (55) we do not specify how to choose the parameter α in the descent vector $\mathbf{u}$ defined by Eq. (36). One way is that α is chosen by the user; however, this is somewhat an *ad hoc* strategy. Much better, we can determine a suitable α such that s as defined by Eq. (52) is minimized with respect to α, because a smaller s will lead to a faster convergence as shown by Eq. (58).

Thus by inserting Eq. (43) for a_0 into Eq. (52) we can write s as to be

$$s = 1 - \frac{(1-\gamma^2)(\mathbf{r}\cdot\mathbf{u})^2}{2\phi\mathbf{u}\cdot(\mathbf{A}\mathbf{u})}, \tag{59}$$

where $\mathbf{u}$ as defined by Eq. (36) includes a parameter α. Let $\partial s/\partial\alpha = 0$, and through some algebraic operations we can solve α by

$$\alpha = \frac{g_1 g_4 - g_2 g_3}{g_2 g_4 - g_1 g_5}, \tag{60}$$

where

$$g_1 = \|\mathbf{r}\|^2, \tag{61}$$

$$g_2 = \mathbf{r}\cdot\mathbf{x}, \tag{62}$$

$$g_3 = \mathbf{r}\cdot(\mathbf{A}\mathbf{r}), \tag{63}$$

$$g_4 = \mathbf{r}\cdot(\mathbf{A}\mathbf{x}), \tag{64}$$

$$g_5 = \mathbf{x}\cdot(\mathbf{A}\mathbf{x}). \tag{65}$$

The parameter α given by Eq. (60) is the optimal one in the sense that it maximizes the convergence rate in Eq. (58); and correspondingly, the descent vector $\mathbf{u} = \mathbf{r} + \alpha\mathbf{x}$ is an *optimal vector* in solving the ill-posed linear problem.

3.7 An Algorithm Derived from the Optimal-Vector Method

Since the fictitious time-like variable t is now discretized as $t \in \{0, 1, 2, \ldots\}$, we let $\mathbf{x}_k$ denote the numerical value of $\mathbf{x}$ at the k-th time step. Thus, we arrive at a purely iterative algorithm by Eq. (55):

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1-\gamma)\frac{\mathbf{r}_k\cdot\mathbf{u}_k}{\mathbf{u}_k\cdot\mathbf{A}\mathbf{u}_k}\mathbf{u}_k, \tag{66}$$

which leaves a parameter γ being determined by the user. The value of $0 \le \gamma < 1$ is problem dependent, and a suitable choice of the relaxed parameter γ can greatly accelerate the convergence speed.

Consequently, we have developed a novel algorithm based on the optimal vector method (OVM):

(i) Select $0 \le \gamma < 1$, and give an initial $\mathbf{x}_0$.

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

$$
\begin{aligned}
g_1^k &= \|\mathbf{r}_k\|^2, \\
g_2^k &= \mathbf{r}_k \cdot \mathbf{x}_k, \\
g_3^k &= \mathbf{r}_k \cdot (\mathbf{A}\mathbf{r}_k), \\
g_4^k &= \mathbf{r}_k \cdot (\mathbf{A}\mathbf{x}_k), \\
g_5^k &= \mathbf{x}_k \cdot (\mathbf{A}\mathbf{x}_k), \\
\alpha_k &= \frac{g_1^k g_4^k - g_2^k g_3^k}{g_2^k g_4^k - g_1^k g_5^k}, \\
\mathbf{u}_k &= \mathbf{r}_k + \alpha_k \mathbf{x}_k, \\
\mathbf{x}_{k+1} &= \mathbf{x}_k - (1-\gamma)\frac{\mathbf{r}_k \cdot \mathbf{u}_k}{\mathbf{u}_k \cdot \mathbf{A}\mathbf{u}_k}\mathbf{u}_k.
\end{aligned}
\tag{67}
$$

If $\|\mathbf{r}_{k+1}\| < \varepsilon$ then stop; otherwise, go to step (ii).

In the above, if $g_2^k g_4^k - g_1^k g_5^k = 0$ we can take $\alpha_k = 0$. Corresponding to the static parameter α as used previously in the Tikhonov regularization method, the present α_k can be labelled as a *dynamical regularization parameter*, and meanwhile, the present algorithm is one kind of the *dynamical (adaptive) Tikhonov regularization method* for solving the ill-posed linear systems. Below we give numerical examples to test the performance of the present optimal-vector method (OVM). When both the two parameters γ and α are zeroes, the present OVM reduces to the SDM introduced in Sect. 2.2. If $\alpha = 0$, the OVM reduces to the relaxed SDM [26, 29]. Recently, Liu and Atluri [34], and Liu and Kuo [38] have developed the counterparts of the present algorithm for solving the nonlinear algebraic equations, which were categorized into the *optimal iterative algorithms*.

4 Numerical Examples

In this section we assess the performance of the newly developed Optimal Vector Method (OVM) for solving some well-known ill-posed linear problems under the disturbance of noise on the given data. The algorithm in Sect. 3.7 is easily being programmed in the Fortran code, which is suitable to run the program in a portable PC machine. The plots to show the numerical results are all carried out in a PC with the Grapher system.

4.1 Example 1: A Simple but Highly Ill-conditioned Linear System

In this example we consider a two-dimensional but highly ill-conditioned linear system:

$$
\begin{bmatrix} 2 & 6 \\ 2 & 6.0001 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 8 \\ 8.0001 \end{bmatrix}. \tag{68}
$$

The condition number of this system is $\text{cond}(\mathbf{A}) = 1.596 \times 10^{11}$, where $\mathbf{A} = \mathbf{B}^{\mathrm{T}}\mathbf{B}$ and $\mathbf{B}$ denotes the coefficient matrix. The exact solution is $(x, y) = (1, 1)$.

Under a convergence criterion $\varepsilon = 10^{-12}$, and starting from an initial point $(x_0, y_0) = (10, 10)$ we apply the CGM and the Optimal Vector Method (OVM) with $\gamma = 0$ to solve the above equation. When the CGM spent 4 steps, the OVM spent only 2 steps, of which the

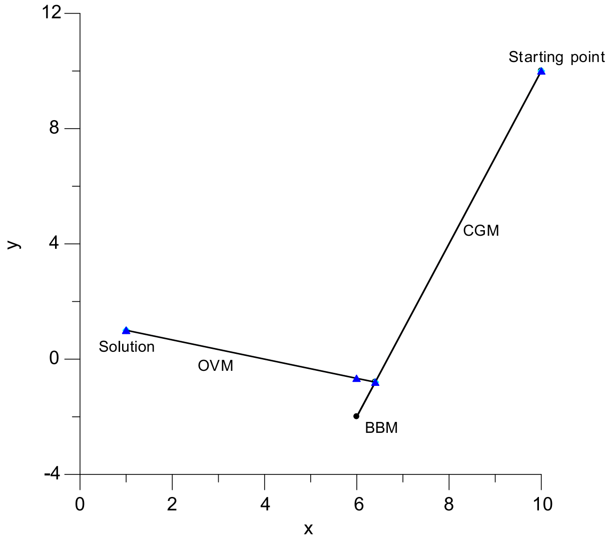

Fig. 1 For example 1 comparing the iterative paths obtained by the CGM, OVM and BBM

iterative paths are compared in Fig. 1. The maximum error obtained by the CGM is 1.94×10^{-5}, which is larger than 8.129×10^{-6} that obtained by the OVM. For this example, under a convergence criterion $\varepsilon = 10^{-8}$ the BBM converges very fast with only three iterations, but it leads to an incorrect solution $(x, y) = (6.400031543, -0.7999954938)$ as shown in Fig. 1.

4.2 Example 2: The Hilbert Matrix

In this example we consider a highly ill-conditioned linear equation (7) with $\mathbf{A}$ given by

$$A_{ij} = \frac{1}{i+j-1}, \tag{69}$$

which is well known as the Hilbert matrix. The ill-posedness of Eq. (7) with the above $\mathbf{A}$ increases very fast with n. Todd [56] has proven that the asymptotic of condition number of the Hilbert matrix is

$$\mathcal{O}\left(\frac{(1+\sqrt{2})^{4n+4}}{\sqrt{n}}\right).$$

In order to compare the numerical solutions with exact solution we suppose that $x_1 = x_2 = \cdots = x_n = 1$ to be the exact solution, and then by Eq. (69) we have

$$b_i = \sum_{j=1}^{n} \frac{1}{i+j-1} + \sigma R(i), \tag{70}$$

where we consider a noise being imposed on the data with $R(i)$ the random numbers in $[-1, 1]$.

We calculate this problem for the case with $n = 50$. The resulting linear equations system is highly ill-conditioned, since the condition number is very large up to 1.1748×10^{19}.

We fix the noise $\sigma = 10^{-8}$ in order to apply the SDM and the CGM. Both algorithms are divergent when the noise level is larger than $\sigma = 10^{-8}$. With a stopping criterion

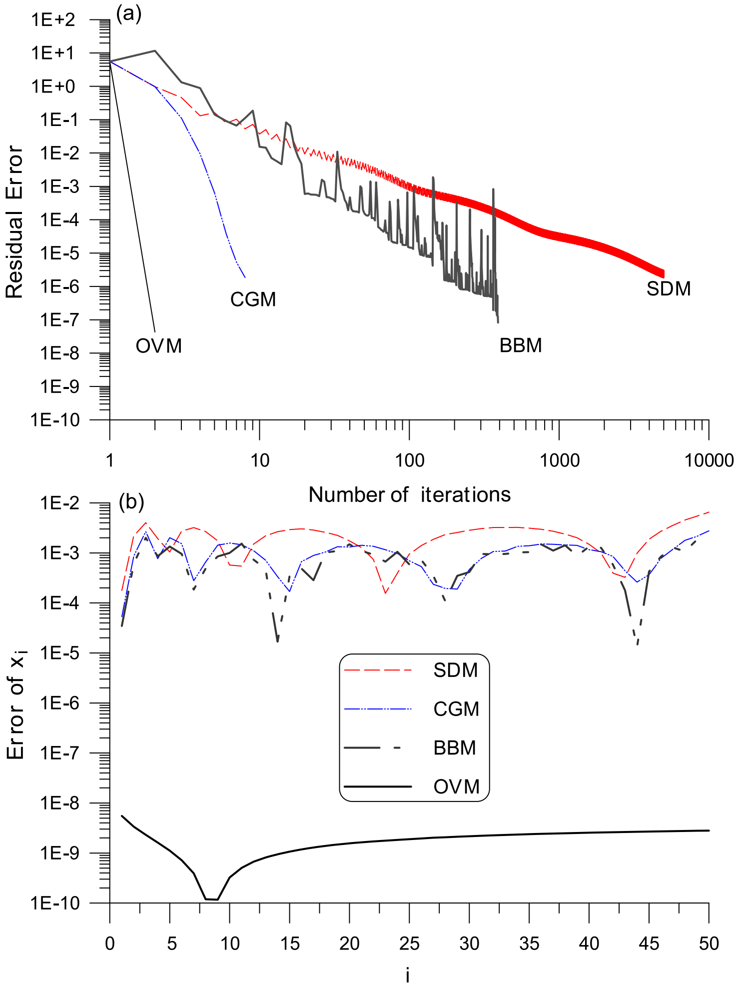

Fig. 2 For example 2 of a Hilbert linear system: (**a**) comparing the residual errors of SDM, CGM, BBM and OVM, and (**b**) showing the numerical errors

$\varepsilon = 10^{-7}$ and starting from the initial condition $x_i = 0.5$, the SDM over 5000 iterations does not converge to the exact solution very accurately, as shown in Fig. 2(a) for the residual error and Fig. 2(b) for the numerical error by the thin dashed lines, whose maximum error is 6.48×10^{-3}. At the same time, the CGM spent 9 iterations as shown in Fig. 2(a) for the residual error and Fig. 2(b) for the numerical error by the dashed-dotted lines, whose maximum error is 2.76×10^{-3}. The BBM spent 390 iterations as shown in Fig. 2(a) for the residual error and Fig. 2(b) for the numerical error by the thick dashed lines, whose maximum error is 2.2×10^{-3}. Conversely, the OVM with $\gamma = 0$ converges very fast with 2 iterations, as shown in Fig. 2(a) for the residual error and Fig. 2(b) for the numerical error by the solid lines, whose maximum error is 5.5×10^{-9}.

4.3 The Fredholm Integral Equations

Another possible application of the present OVM algorithm is solving the first-kind linear Fredholm integral equation:

$$\int_a^b K(s,t)x(t)dt = h(s), \quad s \in [c,d], \tag{71}$$

where $K(s,t)$ and $h(s)$ are known functions and $x(t)$ is an unknown function. We also suppose that $h(s)$ is perturbed by random noise. Some numerical methods to solve the above equation are discussed in [23, 42, 43, 57], and references therein.

To demonstrate other applications of the OVM, we further consider a singularly perturbed Fredholm integral equation, because it is known to be severely ill-posed:

$$\epsilon x(s) + \int_a^b K(s,t)x(t)dt = h(s), \quad s \in [c,d]. \tag{72}$$

When $\epsilon = 0$, the above equation reduces to Eq. (71). So we only consider the discretization of Eq. (72). Let us discretize the intervals of $[a,b]$ and $[c,d]$ into m_1 and m_2 subintervals by noting $\Delta t = (b-a)/m_1$ and $\Delta s = (c-d)/m_2$. Let $x_j := x(t_j)$ be a numerical value of x at a grid point t_j, and let $K_{i,j} = K(s_i, t_j)$ and $h_i = h(s_i)$, where $t_j = a + (j-1)\Delta t$ and $s_i = c + (i-1)\Delta s$. Through the use of a trapezoidal rule on the integral term, Eq. (72) can be discretized into

$$\epsilon x_i + \frac{\Delta t}{2} K_{i,1} x_1 + \Delta t \sum_{j=2}^{m_1} K_{i,j} x_j + \frac{\Delta t}{2} K_{i,m_1+1} x_{m_1+1} = h_i, \quad i = 1, \ldots, m_2 + 1, \tag{73}$$

which are linear algebraic equations denoted by:

$$\mathbf{B}\mathbf{x} = \mathbf{b}_1, \tag{74}$$

where $\mathbf{B}$ is a rectangular matrix with dimensions $(m_2+1) \times (m_1+1)$. Here, $\mathbf{b}_1 = (h_1, \ldots, h_{m_2+1})^{\mathrm{T}}$, and

$$\mathbf{x} = (x_1, \ldots, x_{m_1+1})^{\mathrm{T}} \tag{75}$$

is the unknown vector. The data h_j may be corrupted by noise, such that,

$$\hat{h}_j = h_j + \sigma R(j). \tag{76}$$

4.3.1 A First-Kind Fredholm Integral Equation

We consider the problem of finding $x(t)$ in the following equation:

$$\int_0^1 \left[\sin(s+t) + e^t \cos(s-t)\right] x(t)dt = 1.4944 \cos s + 1.4007 \sin s, \quad s \in [0,1], \tag{77}$$

where $x(t) = \cos t$ is the exact solution. We use the following parameters $m_1 = m_2 = 60$, $\gamma = 0.01$ and $\varepsilon = 10^{-5}$ in the OVM to calculate the numerical solution under a noise with $\sigma = 0.01$. Through only 9 steps it is convergent to the true solution with the maximum error

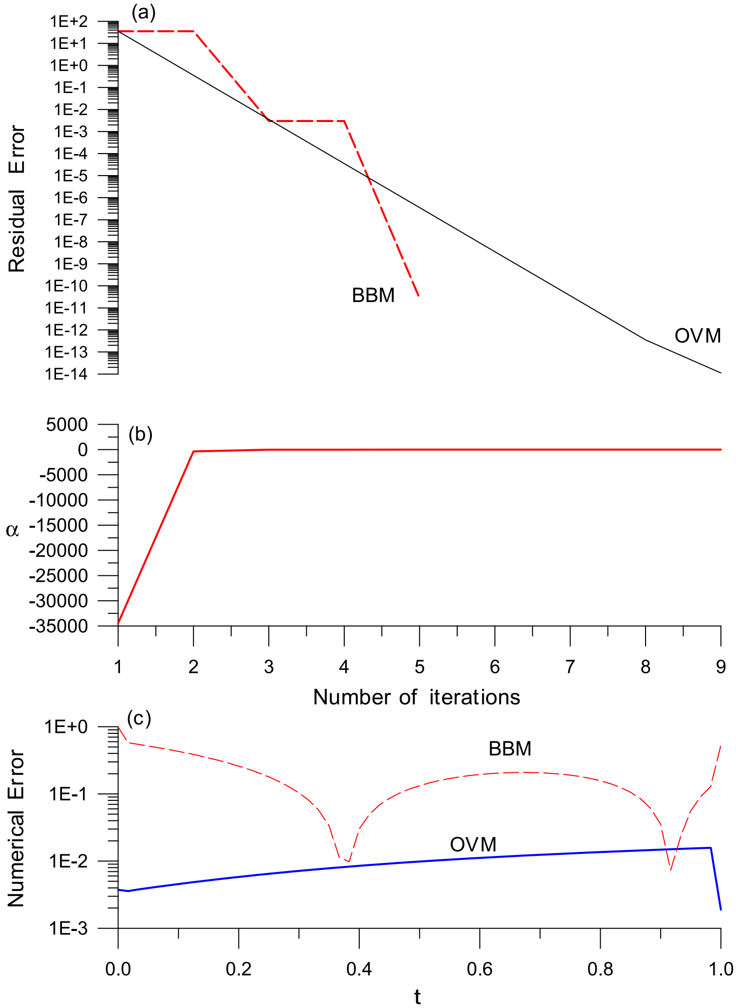

Fig. 3 For a first-kind Fredholm integral equation solved by the OVM and BBM: showing (**a**) residual errors, (**b**) dynamical regularization parameter, and (**c**) numerical errors

0.0244 as shown in Fig. 3. Even under a large noise our calculated result is better than that calculated by Maleknejad et al. [43]. The present OVM can adjust the parameter α as shown in Fig. 3(b) to that near to the relaxed SDM [26] after the first few steps. At the last four steps α is given by 1.05492×10^{-7}, -8.03471×10^{-12}, 4.89208×10^{-14} and 6.15574×10^{-16}. At the same time, when we apply the BBM to this problem, we find that it is convergent with five iterations as shown in Fig. 3(a), but its numerical error is quite large as shown in Fig. 3(c) with the maximum error being 0.987.

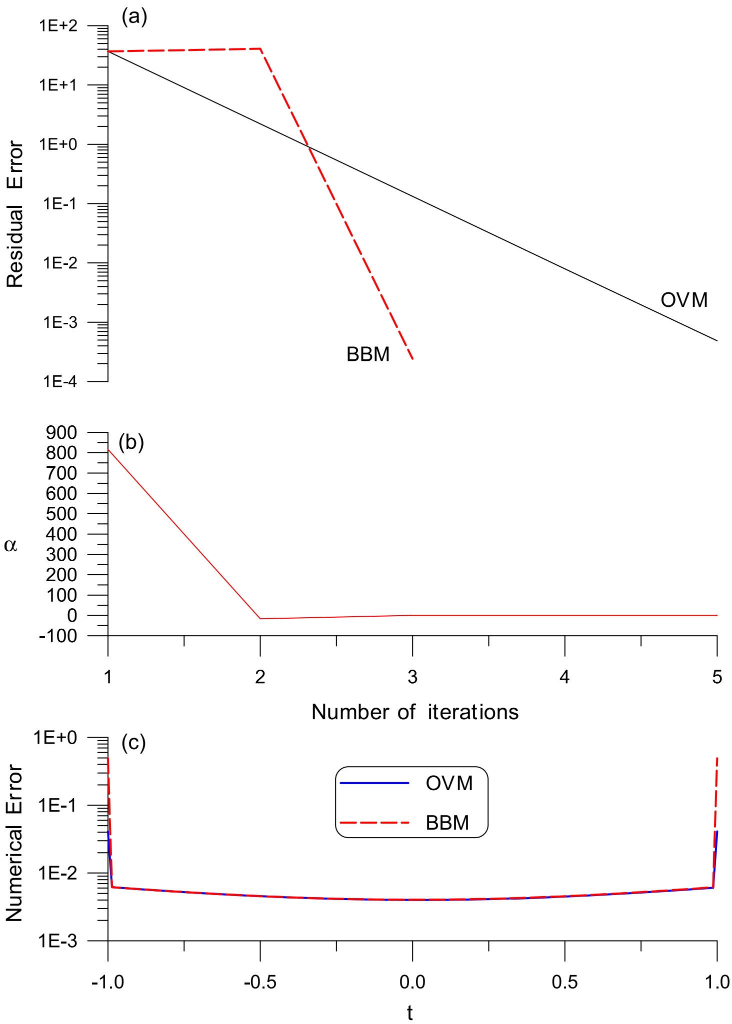

Fig. 4 For a second-kind Fredholm integral equation solved by the OVM and BBM: showing (**a**) residual errors, (**b**) dynamical regularization parameter, and (**c**) numerical errors

4.3.2 A Second-Kind Fredholm Integral Equation

Next, we consider the problem of finding $x(t)$ in the following equation:

$$\int_{-1}^{1} \cosh(s+t)x(t)dt - 0.01x(s) = \cosh s, \quad s \in [-1, 1], \tag{78}$$

where $x(t) = 2\cosh t/(2 + \sinh 2 - 0.02)$ is the exact solution. We use the following parameters $m_1 = m_2 = 150$, $\gamma = 0.06$ and $\varepsilon = 10^{-3}$ in the OVM to calculate the numerical solution under a noise with $\sigma = 0.001$. Through only 5 steps it is convergent to the true solution with the maximum error 0.041 as shown in Fig. 4. In addition that near the two ends of the interval, the most numerical solution is quite accurate in the three orders. The present algorithm of OVM can adjust the parameter α as shown in Fig. 4(b) to that near to the relaxed SDM [26] after the first few steps. At the last three steps α is given by 0.02104, 0.00143471 and 0.908795×10^{-4}. When we apply the BBM to this problem, we find that it is convergent very fast with only three iterations as shown in Fig. 4(a), and its numerical error is almost coincident with that of the OVM as shown in Fig. 4(c); however, the error of the BBM at two ends is quite large with the maximum error being 0.4938.

4.4 A Finite-Difference Equation

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

$$\begin{aligned} &-u''(x) = f(x), \quad 0 < x < 1, \\ &u(0) = a, \qquad u(1) = b, \end{aligned} \tag{79}$$

we can derive a linear equations system

$$\mathbf{Au} = \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} (\Delta x)^2 f(\Delta x) + a \\ (\Delta x)^2 f(2\Delta x) \\ \vdots \\ (\Delta x)^2 f((n-1)\Delta x) \\ (\Delta x)^2 f(n\Delta x) + b \end{bmatrix}, \tag{80}$$

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. The above matrix $\mathbf{A}$ is known as a central difference matrix.

Taking the inversion of $\mathbf{A}$ in Eq. (80) we can obtain the unknown vector $\mathbf{u}$. However, there exhibits a great difficulty when $\mathbf{A}$ has a large condition number. The eigenvalues of $\mathbf{A}$ are found to be

$$4\sin^2 \frac{k\pi}{2(n+1)}, \quad k = 1, 2, \ldots, n, \tag{81}$$

which together with the symmetry of $\mathbf{A}$ indicates that $\mathbf{A}$ is positive definite, and

$$\operatorname{cond}(\mathbf{A}) = \frac{\sin^2 \frac{n\pi}{2(n+1)}}{\sin^2 \frac{\pi}{2(n+1)}} \tag{82}$$

maybe get a large number when the grid number n is very large.

In this numerical test we fix $n = 300$ and thus the condition number of $\mathbf{A}$ is 36475. Let us consider the boundary value problem in Eq. (79) with $f(x) = \sin \pi x$. The exact solution is

$$u(x) = a + (b-a)x + \frac{1}{\pi^2} \sin \pi x, \tag{83}$$

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

A random noise with intensity $\sigma = 0.0001$ is added into the right-hand side of Eq. (80). Upon compared with the given data, which are in the order of $(1/300)^2 \approx 1.11 \times 10^{-5}$, the noise is ten times larger than the given data. We find that the CGM leads to an unstable solution which does not converge within 5000 iterations even under a loose convergence

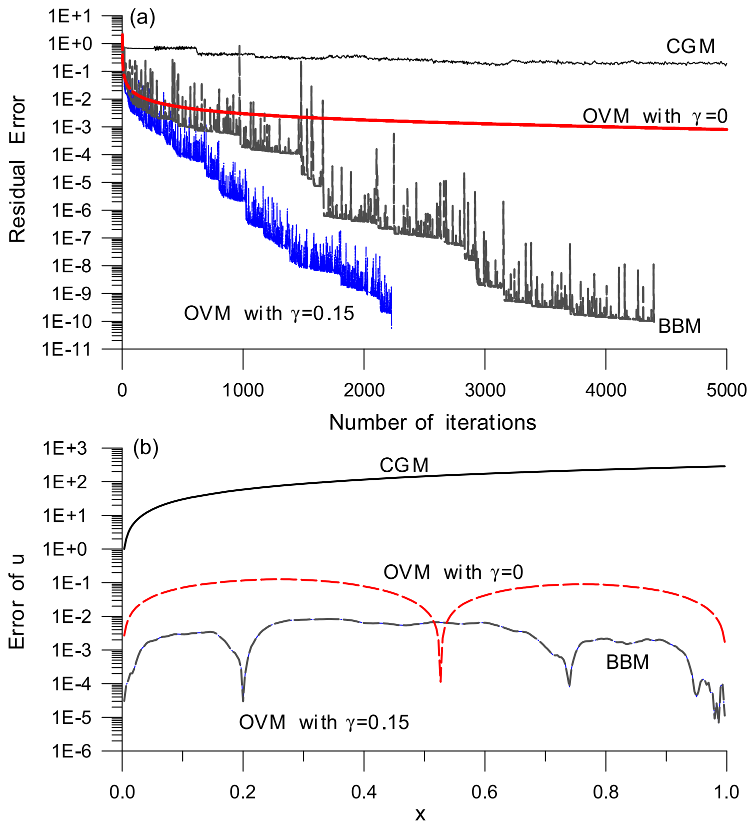

Fig. 5 For a second-order differential equation: (**a**) comparing the residual errors of CGM, BBM, OVM with $\gamma = 0$, and OVM with $\gamma = 0.15$, and (**b**) comparing numerical errors

criterion with $\varepsilon = 0.01$. We plot the residual error curve with respect to the number of iterations in Fig. 5(a) by the solid line for the CGM, whose error as shown in Fig. 5(b) by the solid line has the maximum value 286.87.

Then we apply the OVM with $\gamma = 0$ to this problem. Under a convergence criterion $\varepsilon = 10^{-10}$, we plot the residual error curve in Fig. 5(a) by the dashed line, whose error as shown in Fig. 5(b) by the dashed line has the maximum value 0.125. Obviously, the OVM with $\gamma = 0$ is convergent slowly over 5000 iterations. In order to improve the convergence speed we apply the OVM with $\gamma = 0.15$ to this problem. Under a convergence criterion $\varepsilon = 10^{-10}$, the OVM with $\gamma = 0.15$ is convergent with 2226 iterations, and we plot the residual error curve in Fig. 5(a) by the dashed-dotted line, whose error as shown in Fig. 5(b) by the dashed-dotted line has the maximum value 8.41×10^{-3}. The BBM under the above convergence criterion can converge with 4399 iterations as shown in Fig. 5(a), while its numerical error as shown in Fig. 5(b) is coincident with that of the OVM with $\gamma = 0.15$. However, the OVM is convergent faster than the BBM two times.

For the OVM with $\gamma = 0$, the value of α is plotted in Fig. 6(a) by the dashed line, which is much fast tending to the value in the order of 10^{-6}. For the OVM with $\gamma = 0.15$, the value of α is plotted in Fig. 6(b) by the dashed line, which exhibits an intermittent behavior [24, 25], and then it is fast tending to the value in the order of 10^{-6}.

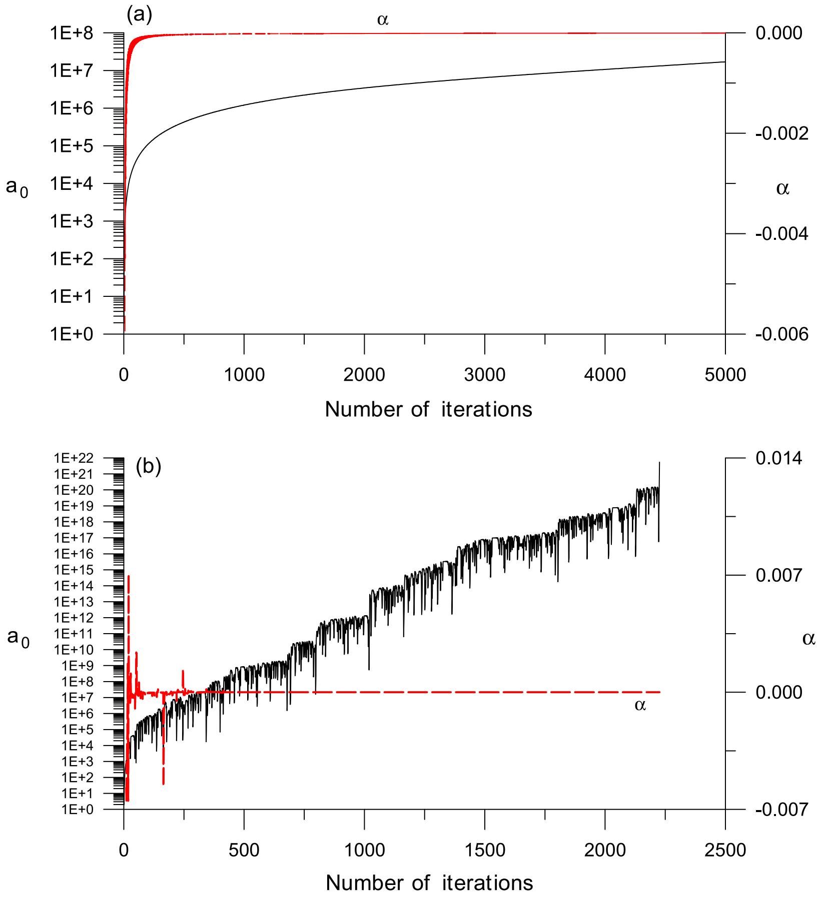

Fig. 6 For a second-order differential equation showing the dynamical regularization parameter α and a_0 for (**a**) OVM with $\gamma = 0$, and (**b**) for OVM with $\gamma = 0.15$

Now, we explain the parameter γ appeared in Eq. (67). In Figs. 6(a) and 6(b) we compare a_0 obtained by OVM with $\gamma = 0$ and $\gamma = 0.15$. From Fig. 6(a) 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. (43) it means that there exists an attracting set for the iterative orbit of **x** described by the following manifold:

$$\frac{\phi \mathbf{u}^{\mathrm{T}} \mathbf{A} \mathbf{u}}{(\mathbf{r} \cdot \mathbf{u})^2} = \text{Constant}. \tag{84}$$

Upon the iterative orbit is approached to this manifold, it is slowly to reduce the residual error as shown in Fig. 5(a) by the dashed line for the case of OVM with $\gamma = 0$. The manifold defined by Eq. (84) is a *slow manifold*. Conversely, for the case $\gamma = 0.15$, a_0 is no more tending to a constant as shown in Fig. 6(b). Because the iterative orbit is not attracted by an attracting slow manifold, the residual error as shown in Fig. 5(a) by the dashed-dotted line for the case of OVM with $\gamma = 0.15$ can be reduced very fast. For the latter case the new algorithm of OVM can give very accurate numerical solution with a residual error tending to 10^{-5}. Thus we can observe that when γ varies from zero to a positive value, the iterative dynamics given by Eq. (67) undergoes a Hopf bifurcation, like as the ODEs behavior observed by Liu [24, 25]. The original stable slow manifold existent for $\gamma = 0$ now becomes a ghost manifold for $\gamma = 0.15$, and thus the iterative orbit generated from the algorithm with the value $\gamma = 0.15$ does not be attracted by that slow manifold again, and instead of the intermittency occupies, leading to an irregularly jumping behaviors of a_0 and of the residual error as shown respectively in Fig. 6(b) by the solid line and Fig. 5(a) by the dashed-dotted line.

5 Conclusions

In the present article, a new Tikhonov-like method has been introduced; that the new method is based on computing the approximate solution of ill-posed linear problem on a proper invariant manifold, allows a natural stability; this approach is completed by computing adaptively and optimally the regularization parameter which is a central difficulty in the Tikhonov-like approaches of the ill-posed linear problems. The resultant algorithm is coined as an *optimal vector method* (OVM) for effectively solving the ill-posed linear problems. Several ill-posed numerical examples were examined, which revealed that the OVM has a better computational efficiency and accuracy than the classical methods of SDM, CGM as well as the BBM, even for the highly ill-conditioned linear equations, which are randomly disturbed by a large noise being imposed on the given data.

Acknowledgements Taiwan's National Science Council project NSC-100-2221-E-002-165-MY3 and the 2011 Outstanding Research Award, and the anonymous referee comments to improve the quality of this article are highly appreciated.

References

1. Barzilai, J., Borwein, J.M.: Two point step size gradient methods. IMA J. Numer. Anal. **8**, 141–148 (1988)
2. Bhaya, A., Kaszkurewicz, E.: Control Perspectives on Numerical Algorithms and Matrix Problems. Advances in Design and Control, vol. 10. SIAM, Philadelphia (2006)
3. Chehab, J.-P., Laminie, J.: Differential equations and solution of linear systems. Numer. Algorithms **40**, 103–124 (2005)
4. Dai, Y.H., Liao, L.H.: R-linear convergence of the Barzilai and Borwein gradient method. IMA J. Numer. Anal. **22**, 1–10 (2002)
5. Dai, Y.H., Yuan, Y.: Alternate minimization gradient method. IMA J. Numer. Anal. **23**, 377–393 (2003)
6. Dai, Y.H., Yuan, J.Y., Yuan, Y.: Modified two-point stepsize gradient methods for unconstrained optimization. Comput. Optim. Appl. **22**, 103–109 (2002)
7. Davidenko, D.: On a new method of numerically integrating a system of nonlinear equations. Dokl. Akad. Nauk SSSR **88**, 601–604 (1953)
8. Engl, H.W.: Discrepancy principles for Tikhonov regularization of ill-posed problems leading to optimal convergence rates. J. Optim. Theory Appl. **52**, 209–215 (1987)
9. Engl, H.W.: On the choice of the regularization parameter for iterated Tikhonov regularization of ill-posed problems. J. Approx. Theory **49**, 55–63 (1987)
10. Fletcher, R.: On the Barzilai-Borwein gradient method. In: Qi, L., Teo, K., Yang, X. (eds.) Optimization and Control with Applications, pp. 235–256. Springer, New York (2005)
11. Friedlander, A., Martinez, J.M., Molina, B., Raydan, M.: Gradient method with retards and generalizations. SIAM J. Numer. Anal. **36**, 275–289 (1999)
12. Gfrerer, H.: An a posteriori parameter choice for ordinary and iterated Tikhonov regularization of ill-posed problems leading to optimal convergence rates. Math. Comput. **49**, 507–522 (1987)
13. Goloub, B., Heath, M., Wahba, G.: Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics **21**, 215–223 (1979)

14. Hansen, P.C.: Analysis of discrete ill-posed problems by means of the L-curve. SIAM Rev. **34**, 561–580 (1992)
15. Hansen, P.C., O'Leary, D.P.: The use of the L-curve in the regularization of discrete ill-posed problems. SIAM J. Sci. Comput. **14**, 1487–1503 (1993)
16. Helmke, U., Moore, J.B.: Optimization and Dynamical Systems. Springer, Berlin (1994)
17. Hoang, N.S., Ramm, A.G.: Solving ill-conditioned linear algebraic systems by the dynamical systems method (DSM). Inverse Probl. Sci. Eng. **16**, 617–630 (2008)
18. Hoang, N.S., Ramm, A.G.: Dynamical systems gradient method for solving ill-conditioned linear algebraic systems. Acta Appl. Math. **111**, 189–204 (2010)
19. Jacoby, S.L.S., Kowalik, J.S., Pizzo, J.T.: Iterative Methods for Nonlinear Optimization Problems. Prentice-Hall, Englewood Cliffs (1972)
20. Kilmer, M.E., O'Leary, D.P.: Choosing regularization parameter in iterative methods for ill-posed problems. SIAM J. Matrix Anal. Appl. **22**, 1204–1221 (2001)
21. Ku, C.-Y., Yeih, W., Liu, C.-S.: Solving non-linear algebraic equations by a scalar Newton-homotopy continuation method. Int. J. Nonlinear Sci. Numer. Simul. **11**, 435–450 (2010)
22. Kunisch, K., Zou, J.: Iterative choices of regularization parameters in linear inverse problems. Inverse Probl. **14**, 1247–1264 (1998)
23. Landweber, L.: An iteration formula for Fredholm integral equations of the first kind. Am. J. Math. **73**, 615–624 (1951)
24. Liu, C.-S.: Intermittent transition to quasiperiodicity demonstrated via a circular differential equation. Int. J. Non-Linear Mech. **35**, 931–946 (2000)
25. Liu, C.-S.: A study of type I intermittency of a circular differential equation under a discontinuous right-hand side. J. Math. Anal. Appl. **331**, 547–566 (2007)
26. Liu, C.-S.: A revision of relaxed steepest descent method from the dynamics on an invariant manifold. Comput. Model. Eng. Sci. **80**, 57–86 (2011)
27. Liu, C.-S.: The concept of best vector used to solve ill-posed linear inverse problems. Comput. Model. Eng. Sci. **83**, 499–525 (2012)
28. Liu, C.-S.: Optimally scaled vector regularization method to solve ill-posed linear problems. Appl. Math. Comput. **218**, 10602–10616 (2012)
29. Liu, C.-S.: Modifications of steepest descent method and conjugate gradient method against noise for ill-posed linear systems. Commun. Numer. Anal. **2012**, cna-00115 (2012). 24 pages
30. Liu, C.-S., Atluri, S.N.: A novel time integration method for solving a large system of non-linear algebraic equations. Comput. Model. Eng. Sci. **31**, 71–83 (2008)
31. Liu, C.-S., Atluri, S.N.: A fictitious time integration method for the numerical solution of the Fredholm integral equation and for numerical differentiation of noisy data, and its relation to the filter theory. Comput. Model. Eng. Sci. **41**, 243–261 (2009)
32. Liu, C.-S., Atluri, S.N.: A highly accurate technique for interpolations using very high-order polynomials, and its applications to some ill-posed linear problems. Comput. Model. Eng. Sci. **43**, 253–276 (2009)
33. Liu, C.-S., Atluri, S.N.: 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. Comput. Model. Eng. Sci. **71**, 279–304 (2011)
34. Liu, C.-S., Atluri, S.N.: An iterative algorithm for solving a system of nonlinear algebraic equations, $\mathbf{F}(\mathbf{x}) = \mathbf{0}$, using the system of ODEs with an optimum α in $\dot{\mathbf{x}} = \lambda[\alpha\mathbf{F} + (1-\alpha)\mathbf{B}^{\mathrm{T}}\mathbf{F}]$; $B_{ij} = \partial F_i/\partial x_j$. Comput. Model. Eng. Sci. **73**, 395–431 (2011)
35. Liu, C.-S., Atluri, S.N.: 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. Comput. Model. Eng. Sci. **80**, 275–298 (2011)
36. Liu, C.-S., Chang, C.W.: Novel methods for solving severely ill-posed linear equations system. J. Mar. Sci. Technol. **17**, 216–227 (2009)
37. Liu, C.-S., Hong, H.K., Atluri, S.N.: Novel algorithms based on the conjugate gradient method for inverting ill-conditioned matrices, and a new regularization method to solve ill-posed linear systems. Comput. Model. Eng. Sci. **60**, 279–308 (2010)
38. Liu, C.-S., Kuo, C.-L.: A dynamical Tikhonov regularization method for solving nonlinear ill-posed problems. Comput. Model. Eng. Sci. **76**, 109–132 (2011)
39. Liu, C.-S., Yeih, W., Atluri, S.N.: On solving the ill-conditioned system $\mathbf{Ax} = \mathbf{b}$: general-purpose conditioners obtained from the boundary-collocation solution of the Laplace equation, using Trefftz expansions with multiple length scales. Comput. Model. Eng. Sci. **44**, 281–311 (2009)
40. Liu, C.-S., Yeih, W., Kuo, C.-L., Atluri, S.N.: A scalar homotopy method for solving an over/under-determined system of non-linear algebraic equations. Comput. Model. Eng. Sci. **53**, 47–71 (2009)

41. Lukas, M.A.: Comparison of parameter choice methods for regularization with discrete noisy data. Inverse Probl. **14**, 161–184 (1998)
42. Maleknejad, K., Mahmoudi, Y.: Numerical solution of linear integral equation by using hybrid Taylor and block-pulse functions. Appl. Math. Comput. **149**, 799–806 (2004)
43. Maleknejad, K., Mollapourasl, R., Nouri, K.: Convergence of numerical solution of the Fredholm integral equation of the first kind with degenerate kernel. Appl. Math. Comput. **181**, 1000–1007 (2006)
44. Morozov, V.A.: On regularization of ill-posed problems and selection of regularization parameter. J. Comput. Math. Phys. **6**, 170–175 (1966)
45. Morozov, V.A.: Methods for Solving Incorrectly Posed Problems. Springer, New York (1984)
46. Ostrowski, A.M.: Solution of Equations in Euclidean and Banach Spaces, 3rd edn. Academic Press, New York (1973)
47. Ramm, A.G.: Dynamical System Methods for Solving Operator Equations. Elsevier, Amsterdam (2007)
48. Ramm, A.G.: Dynamical systems method for solving linear ill-posed problems. Ann. Pol. Math. **95**, 253–272 (2009)
49. Raydan, M.: On the Barzilai and Borwein choice of steplength for the gradient method. IMA J. Numer. Anal. **13**, 321–326 (1993)
50. Raydan, M.: The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J. Optim. **7**, 26–33 (1997)
51. Raydan, M., Svaiter, B.F.: Relaxed steepest descent and Cauchy-Barzilai-Borwein method. Comput. Optim. Appl. **21**, 155–167 (2002)
52. Resmerita, E.: Regularization of ill-posed problems in Banach spaces: convergence rates. Inverse Probl. **21**, 1303–1314 (2005)
53. Stewart, G.: Introduction to Matrix Computations. Academic Press, New York (1973)
54. Sweilam, N.H., Nagy, A.M., Alnasr, M.H.: An efficient dynamical systems method for solving singularly perturbed integral equations with noise. Comput. Math. Appl. **58**, 1418–1424 (2009)
55. Tikhonov, A.N., Arsenin, V.Y.: Solutions of Ill-posed Problems. Wiley, New York (1977)
56. Todd, J.: The condition of finite segments of the Hilbert matrix. In: Taussky, O. (ed.) The Solution of Systems of Linear Equations and the Determination of Eigenvalues. Nat. Bur. of Standards Appl. Math. Series, vol. 39, pp. 109–116 (1954)
57. Wang, W.: A new mechanical algorithm for solving the second kind of Fredholm integral equation. Appl. Math. Comput. **172**, 946–962 (2006)
58. Wang, Y., Xiao, T.: Fast realization algorithms for determining regularization parameters in linear inverse problems. Inverse Probl. **17**, 281–291 (2001)
59. Xie, J., Zou, J.: An improved model function method for choosing regularization parameters in linear inverse problems. Inverse Probl. **18**, 631–643 (2002)
60. Yuan, Y.: A new stepsize for the steepest descent method. J. Comput. Math. **24**, 149–156 (2006)