

Accelerated Bidirectional Modifications of the Steepest Descent Method for Ill-posed Linear Algebraic Systems with Dynamics-theoretical and Optimization Interpretation

Li-Wei Liu¹, Hong-Ki Hong¹, & Chein-Shan Liu¹

¹ Department of Civil Engineering, National Taiwan University, Taipei 10617, Taiwan

Correspondence: Li-Wei Liu, Department of Civil Engineering, National Taiwan University, Taipei 10617, Taiwan. Tel: 886-2-3366-4264. E-mail: liweiliu@ntu.edu.tw

Received: September 3, 2015 Accepted: September 23, 2015 Online Published: November 3, 2015

doi:10.5539/jmr.v7n4p112 URL: <http://dx.doi.org/10.5539/jmr.v7n4p112>

Abstract

It is well known that the classical numerical algorithm of the steepest descent method (SDM) is effective for well-posed linear systems, but performs poorly for ill-posed ones. In this paper we propose accelerated and/or bidirectional modifications of SDM, namely the accelerated steepest descent method (ASDM), the bidirectional method (2DM), and the accelerated bidirectional method (A2DM). The starting point is a manifold defined in terms of a minimum functional and a fictitious time variable; nevertheless, in the end the fictitious time variable disappears so that we arrive at purely iterative algorithms. The proposed algorithms are justified by dynamics-theoretical and optimization interpretation. The accelerator plays a prominent role of switching from the situation of slow convergence to a new situation that the functional tends to decrease stepwise in an intermittent and ceaseless manner. Three examples of solving ill-posed systems are examined and comparisons are made with exact solutions and with the existing algorithms of the SDM, the Barzilai-Borwein method, and the random SDM, revealing that the new algorithms of ASDM and A2DM have better computational efficiency and accuracy even for the highly ill-posed systems.

Keywords: Ill-posed linear equations, accelerated steepest descent method, accelerated bidirectional method.

1. Introduction

1.1 Ill-posed Problems and Remedy

In this paper we propose robust and easily implemented new methods to solve the system of linear algebraic equations

$$\mathbf{Ax} = \mathbf{b}, \quad (1)$$

where the coefficient matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is a given positive definite matrix, which may be ill-conditioned; the right-hand side vector $\mathbf{b} \in \mathbb{R}^n$ is the input data, which may be corrupted by noise; and $\mathbf{x} \in \mathbb{R}^n$ is the unknown vector to be sought for. We thus encounter the problem that the numerical solution to Eq. (1) may deviate from the exact one to a great extent, when \mathbf{A} is highly ill-conditioned and/or \mathbf{b} is disturbed by noise. In some occasions (for example, design code regulations require engineers to consider hundreds of load combinations \mathbf{b}), the unknown vectors \mathbf{x} may be solved for preferably by first inverting \mathbf{A} , but this again causes a great difficulty when \mathbf{A} has a large condition number.

Many numerical methods used in the computational mechanics are confronted with the ill-posed linear algebraic equations, as demonstrated in the literature (Liu, 2007a; Liu, 2007b; Liu, 2007c; Lorentz & Benallal, 2005). Indeed, the solution of ill-posed linear algebraic equations is an important issue for many engineering and scientific problems. A good numerical method to solve Eq. (1) is not only important in its own right but also beneficial in applications such as (i) the optimization problems including linear programming and nonlinear programming, (ii) the Newton's, quasi-Newton's and homotopy methods for systems of nonlinear equations, and (iii) the finite difference, boundary element and finite element methods for partial differential equations.

To account of the sensitivity to noise a common practice is to invoke a regularization method to tackle this sort of ill-posed problems (Kunisch & Zou, 1998; Resmerita, 2005; Wang & Xiao, 2001; Xie & Zou, 2002), where a suitable regularization parameter is utilized to depress the bias in the computed solution via a better balance of the approximation error and the propagated data error. Developed were several techniques after the pioneering work of Tikhonov & Arsenin (1977). For a large scale system the main choice is to use the iterative regularization algorithms, where a regularized parameter is represented by the number of iterations. The iterative algorithms work if an early stopping criterion is erected to prevent from reconstruction of noisy components in the approximated solutions. To solve the ill-posed linear problems, several methods such as using the fictitious time integration method as an ill-conditioning filter (Liu & Atluri, 2009a), a modified

polynomial expansion method (Liu & Atluri, 2009b), the nonstandard group preserving scheme (Liu & Chang, 2009), and a natural regularization (Liu, Hong, & Alturi, 2010) have been developed.

A quantitative measure of the ill-posedness of Eq. (1) is the condition number of the square matrix \mathbf{A} (Stewart, 1973):

$$\text{cond}(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\|, \quad (2)$$

where $\|\mathbf{A}\|$ is the Frobenius norm of \mathbf{A} . For an arbitrary $\epsilon > 0$, there exists a matrix norm $\|\mathbf{A}\|$ such that $\rho(\mathbf{A}) \leq \|\mathbf{A}\| \leq \rho(\mathbf{A}) + \epsilon$, where $\rho(\mathbf{A})$ is the radius of the spectrum of \mathbf{A} . Therefore, the condition number of \mathbf{A} can be estimated by

$$\text{cond}(\mathbf{A}) = \frac{\max_{\sigma(\mathbf{A})} |\lambda|}{\min_{\sigma(\mathbf{A})} |\lambda|}, \quad (3)$$

where $\sigma(\mathbf{A})$ is the collection of the eigenvalues of \mathbf{A} .

The linear algebraic system (1) with \mathbf{A} having a large condition number usually suffers from the numerical instability problem that an arbitrary small perturbation on the right-hand side \mathbf{b} may lead to a large perturbation to the solution \mathbf{x} on the left-hand side. Roughly speaking, the numerical solution of Eq. (1) may lose the accuracy of k decimal points in \mathbf{x} when $\text{cond}(\mathbf{A}) = 10^k$. The problems of ill-conditioned \mathbf{A} appear in several fields. A famous example of highly ill-conditioned matrices is the Hilbert matrix. It arises naturally in finding an n -degree polynomial function $p(x) = a_0 + a_1x + \dots + a_nx^n$ to best match a continuous function $f(x) \in C[0, 1]$ in the interval of $x \in [0, 1]$:

$$\min_{\deg(p) \leq n} \int_0^1 |f(x) - p(x)|^2 dx. \quad (4)$$

The minimization problem (4) leads to Eq. (1), where \mathbf{A} is the $(n+1) \times (n+1)$ Hilbert matrix, whose elements are of the form

$$A_{ij} = \frac{1}{i+j-1}, \quad (5)$$

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

$$\mathbf{b} = \begin{bmatrix} \int_0^1 f(x) dx \\ \int_0^1 xf(x) dx \\ \vdots \\ \int_0^1 x^n f(x) dx \end{bmatrix} \quad (6)$$

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

1.2 The Steepest Descent Method

Solving Eq. (1) by the steepest descent method (Jacoby, 1972; Ostrowski, 1973) is equivalent to solving the minimization problem:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \varphi(\mathbf{x}), \quad (7)$$

where

$$\varphi(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{b}^T \mathbf{x},$$

in which the superscript T signifies the transpose. By using the Ritz variational principle one can derive the following SDM algorithm.

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

$$\begin{aligned} \mathbf{r}_k &= \mathbf{A} \mathbf{x}_k - \mathbf{b}, \\ \alpha_k &= \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}, \\ \mathbf{x}_{k+1} &= \mathbf{x}_k - \alpha_k \mathbf{r}_k, \end{aligned} \quad (8)$$

where $\mathbf{x}_k := \mathbf{x}(k\Delta t)$ and $\mathbf{r}_k = \mathbf{r}(k\Delta t)$. If \mathbf{x}_k converges according to a given stopping criterion $\|\mathbf{r}_k\| < \epsilon$, then stop; otherwise, go to step (ii).

For the SDM the residual vector \mathbf{r}_k is the steepest descent direction of the function 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 the SDM.

1.3 Steplengths in the SDM, BBM, and RSDM

Modifications to the SDM have recurred. These modifications, although somewhat ad hoc in some cases, have stimulated a new interest in the SDM because it is recognized that the gradient vector itself is not a bad choice of the direction, but rather that the steplength originally dictated by the classical SDM is to blame for the slow convergence behavior. Barzilai & Borwein (1988) were the first, who presented a new choice of steplength through two-point stepsize. The algorithm of the Barzilai-Borwein method (BBM) is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{(\Delta \mathbf{r}_{k-1})^T \Delta \mathbf{x}_{k-1}}{\|\Delta \mathbf{r}_{k-1}\|^2} \mathbf{r}_k, \quad (9)$$

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 failing to guarantee the descent of the minimum functional values, they were able to produce a substantial improvement of the convergence speed in certain tests. Their results have initiated many researches on the SDM, for example, Raydan (1993, 1997), Friedlander *et al.* (1999), Raydan & Svaiter (2002), Dai & Liao (2002), Dai *et al.* (2002), Dai & Yuan (2003), Fletcher (2005), Yuan (2006), *etc.* Among these efforts, a very interesting modification of steplength was the random SDM (abbreviated to RSDM) proposed by Raydan & Svaiter (2002), namely

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \theta_k \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \mathbf{r}_k, \quad (10)$$

where θ_k are random numbers in $[0, 2]$.

These algorithms (8), (9), and (10) may be summarized as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{r}_k,$$

in which

$$\alpha_k = \begin{cases} \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} & \text{for SDM,} \\ \frac{(\Delta \mathbf{r}_{k-1})^T \Delta \mathbf{x}_{k-1}}{\|\Delta \mathbf{r}_{k-1}\|^2} & \text{for BBM,} \\ \theta_k \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \text{ with random } \theta_k \in [0, 2] & \text{for RSDM.} \end{cases}$$

It can be seen that they are different mainly in steplengths.

2. A New Stage—Invariant Manifold

In this paper we approach this steplength problem from a quite different point of view of dynamics on an invariant manifold in the product space of the state space and a fictitious time axis, and propose new strategies.

From Eqs. (1) and (7) it is easy to prove that the minimum is

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

because \mathbf{A} is positive definite, where \mathbf{x}^* is the solution of Eq. (1). We take

$$\phi(\mathbf{x}) := \varphi(\mathbf{x}) + c_0 = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{b}^T \mathbf{x} + c_0, \quad (11)$$

where c_0 is a constant such that $\phi(\mathbf{x}) > 0$. Therefore, solving Eq. (1) by the steepest descent method amounts to solving the minimization problem:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \phi(\mathbf{x}), \quad (12)$$

where $\phi(\mathbf{x})$ is given by Eq. (11). It is obvious that 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 treating Eq. (1) when \mathbf{A} is ill-conditioned. In this paper we consider an iterative regularization method for Eq. (1) by investigating the evolutionary behavior of \mathbf{x} governed by a dynamical system defined on a manifold

$$h(\mathbf{x}, t) := Q(t)\phi(\mathbf{x}) = C(t) \quad (13)$$

formed from $\phi(\mathbf{x})$ with the requirements that $Q(t)$ be positive ($Q(t) > 0$) and strictly increasing ($Q(t_2) > Q(t_1)$ if $t_2 > t_1$) and that $C(t) > 0$. The requirement that $Q(t)$ be a strictly increasing function of a fictitious time variable t enables us to

have an absolutely convergent property in solving Eq. (1). We will utilize such a strictly increasing positive $Q(t)$ to play the role of an accelerator in the dynamics of the iterative methods. We expect $h(\mathbf{x}, t) = C(t)$ to be an *invariant* manifold in the space of (\mathbf{x}, t) for a dynamical system to be specified further. Note that by the above implicit function we in fact have required \mathbf{x} to be a function $\mathbf{x}(t)$ of t , that is, we are considering the trajectory

$$h(\mathbf{x}(t), t) = Q(t)\phi(\mathbf{x}(t)) = C(t) \quad (14)$$

in the invariant manifold (13). Therefore, when taking the differentiation of Eq. (14) with respect to t , we have to fulfill the *consistency* condition

$$\dot{Q}(t)\phi(\mathbf{x}(t)) + Q(t)\mathbf{r}^T(t)\dot{\mathbf{x}}(t) - \dot{C}(t) = 0, \quad (15)$$

where

$$\mathbf{r} = \mathbf{Ax} - \mathbf{b} \quad (16)$$

is the residual vector and an overdot indicates differentiation with respect to t .

Let there be a gradient flow in the state space of (\mathbf{x}) ,

$$\dot{\mathbf{x}} = -\alpha \frac{\partial \phi}{\partial \mathbf{x}} = -\alpha \mathbf{r}, \quad (17)$$

where α is to be determined. Inserting Eq. (17) into Eq. (15) and using Eq. (16) we get

$$\alpha = \frac{\dot{Q}\phi - \dot{C}}{Q\|\mathbf{r}\|^2}. \quad (18)$$

With this we obtain an evolution equation for $\mathbf{x}(t)$:

$$\dot{\mathbf{x}} = \frac{\dot{C} - \dot{Q}\phi}{Q\|\mathbf{r}\|^2} \mathbf{r}. \quad (19)$$

This is the governing equation for the dynamical system defined on the invariant manifold (13). Since, in order to be qualified as an accelerator, $Q(t)$ has been chosen to be a strictly increasing positive function of t , we have an absolutely convergent property in solving Eq. (1) by searching the minimum of ϕ through the following equation,

$$\phi(\mathbf{x}(t)) = \frac{C(t)}{Q(t)}. \quad (20)$$

When t is large the above equation with the aid of the accelerator $Q(t)$ helps the functional ϕ to attain faster its minimum, and meanwhile the solution of Eq. (1) is obtained quicker.

3. Dynamics of Iterative Algorithms

In this section we derive a discrete analogue of the foregoing continuous time dynamics and develop accelerated algorithms based on the discrete time dynamics. Since the fictitious time variable is now discrete, we arrive at purely iterative algorithms with discrete $k \in \{0, 1, 2, \dots\}$ superseding continuous t .

Here and following, we use the symbols $\mathbf{x}_k = \mathbf{x}(k\Delta t)$, $Q_k = Q(k\Delta t)$, $C_k = C(k\Delta t)$, $\alpha_k = \Delta t \alpha(k\Delta t)$, and $\phi_k = \phi(\mathbf{x}_k)$, etc. Hence the trajectory (14) on the invariant manifold (13) becomes

$$Q_{k+1}\phi_{k+1} = C_{k+1}. \quad (21)$$

3.1 Accelerated SDM

We use the Euler scheme to discretize the evolution equation (17),

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{r}_k. \quad (22)$$

Upon denoting $\phi_{k+1} = \phi(\mathbf{x}_{k+1}) = \phi(\mathbf{x}_k - \alpha_k \mathbf{r}_k)$, the minimization problem (12) becomes

$$\min_{\alpha_k} \phi_{k+1}, \quad (23)$$

which leads to

$$\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}. \quad (24)$$

Hence

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \mathbf{r}_k. \quad (25)$$

It is recognized that this is indeed the algorithm (8) of the steepest descent method (SDM). The trajectory

$$\phi_{k+1} = C_{k+1}, \quad (26)$$

of \mathbf{x}_{k+1} generated by Eq. (25) is kept on the invariant manifold if and only if

$$C_{k+1} = C_k - \frac{\mathbf{r}_k^T \mathbf{r}_k}{2\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}. \quad (27)$$

Despite the optimal property (23) and the globally convergent property, the SDM algorithm (25) converges linearly and performs poorly in ill-posed cases (Akaike, 1959). To accelerate the convergent iterations we stick to such C_{k+1} of Eq. (27) but replace $Q_{k+1} = 1$ implied in Eq. (26) as comparing with equation given below by the accelerator $Q_1 > 1$ and $Q_{k+1} > Q_k$. According to Eq. (27), we choose $\mathbf{r}_k^T \mathbf{r}_k$ and $\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k$ be the factors of the accelerator. After derivation, we find the minimization problem

$$\min_{\alpha_k} Q_{k+1} \phi_{k+1} \quad (28)$$

with the best accelerator

$$Q_{k+1} = \exp \left[\frac{2\gamma \mathbf{r}_k^T \mathbf{r}_k \mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}{(1 - \gamma^2)(\mathbf{r}_k^T \mathbf{r}_k)^2 + 2\phi_k \mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \alpha_k \right] \quad (29)$$

and Eq. (22) leads to

$$\alpha_k = (1 - \gamma) \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}. \quad (30)$$

The range of γ contained in Eq. (29) is $0 \leq \gamma < 1$ which is investigated in the next subsection.

Inserting Eq. (30) into Eq. (22) we have

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1 - \gamma) \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \mathbf{r}_k. \quad (31)$$

The trajectory (21) of \mathbf{x}_{k+1} obtained by Eq. (31) stays on the invariant manifold if and only if C_{k+1} and Q_{k+1} satisfy Eqs. (27) and (29), respectively.

Then, we devise the following algorithm:

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

$$\mathbf{r}_k = \mathbf{A} \mathbf{x}_k - \mathbf{b}, \quad (32)$$

$$\alpha_k = (1 - \gamma) \frac{\|\mathbf{r}_k\|^2}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}, \quad (33)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{r}_k. \quad (34)$$

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

For convenience the algorithm is referred to as the accelerated steepest descent method (ASDM)¹. If instead $\gamma = 0$ the above algorithm becomes the classical steepest descent method (SDM). The SDM and ASDM are unidirectional methods in the sense that at each step the optimizations of their algorithms are performed within an one-dimensional space in which the optimal steplengths are found.

3.2 How Does the Accelerator Work?

In the last subsection, we propose an accelerated modification of SDM. For this modification, it is easy to obtain

$$\frac{\phi_{k+1}}{\phi_k} = 1 - (1 - \gamma^2) \frac{(\mathbf{r}_k^T \mathbf{r}_k)^2}{(\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k)(\mathbf{r}_k^T \mathbf{A}^{-1} \mathbf{r}_k)}$$

¹This algorithm had been proposed in (Liu, 2012) from the different point of view and named "MSDM". Here we rename it "ASDM" basing on the concept of accelerator.

by specifying $c_0 = \frac{1}{2} \mathbf{x}^* \mathbf{A} \mathbf{x}^*$ where $\mathbf{A} \mathbf{x}^* = \mathbf{b}$. Upon using the Kantorovich inequality, we get the upper bound of $\frac{\phi_{k+1}}{\phi_k}$ as follows:

$$\frac{\phi_{k+1}}{\phi_k} \leq 1 - (1 - \gamma^2) \frac{4 \text{cond}(\mathbf{A})}{(1 + \text{cond}(\mathbf{A}))^2}. \quad (35)$$

From inequality (35) it follows that $\gamma \in (-1, 1)$ is sufficient for a strictly decreasing sequence $\{\phi_k\}$. However, $\gamma \in (-1, 0)$ leads to $Q_k < 1$ based on Eq. (29); therefore, we have to choose $\gamma \in (0, 1)$ in order to have an accelerator $Q_0 > 1$ and $Q_{k+1} > Q_k$.

The θ_k in the algorithm (10) of the random SDM indeed plays the role of $1 - \gamma$. According to the above analysis based on the viewpoint of dynamics, we suggest that the random numbers be chosen from $\theta_k \in (0, 1)$ (abbreviated to RSDM1) rather than the original ad hoc range $\theta_k \in [0, 2]$ (abbreviated to RSDM).

Another index is

$$\frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k} = 2\gamma - 1 + \frac{(\mathbf{r}_k^T \mathbf{r}_k)(\mathbf{r}_k^T \mathbf{A}^2 \mathbf{r}_k)}{(\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k)^2}.$$

Using again the Kantorovich inequality we derive the upper bound of $\frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}$ as in the following:

$$\frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k} \leq 2\gamma - 1 + \frac{(1 + \text{cond}(\mathbf{A}))^2}{4 \text{cond}(\mathbf{A})}. \quad (36)$$

According to the estimate (36), the strictly decreasing $\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}$ may exist merely in the range

$$\text{cond}(\mathbf{A}) < 3 - 4\gamma + \sqrt{(3 - 4\gamma)^2 - 1}.$$

Observe that the range of strictly decreasing $\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}$ for $\gamma \neq 0$ appears to be narrower than that for $\gamma = 0$. It provides one of the reasons why the irregularly jumping behavior of the residual errors for the accelerated algorithm ASDM ($0 < \gamma < 1$) is more violent than that of the classical algorithm SDM ($\gamma = 0$). The corresponding behavior of SDM ($\gamma = 0$) was explored in (Nocedal *et al.*, 2002).

3.3 Bidirectional Methods

In deriving the unidirectional methods we use the Euler scheme to discretize the dynamical system (17). Now we discretize the flow $\dot{\mathbf{x}}$ as the weighted difference of states:

$$\dot{\mathbf{x}}(k\Delta t) = \frac{\mathbf{x}((k+1)\Delta t) - \beta_k \mathbf{x}(k\Delta t)}{\Delta t}$$

so that the dynamical system (17) becomes

$$\mathbf{x}_{k+1} = \beta_k \mathbf{x}_k - \alpha_k \mathbf{r}_k. \quad (37)$$

With $\phi_{k+1} = \phi(\mathbf{x}_{k+1}) = \phi(\beta_k \mathbf{x}_k - \alpha_k \mathbf{r}_k)$ the minimization problem (12) becomes

$$\min_{\beta_k, \alpha_k} \phi_{k+1}. \quad (38)$$

Thus it follows that

$$\alpha_k = \frac{a_2^k d_1^k - a_1^k d_2^k}{a_1^k a_3^k - (a_2^k)^2}, \quad \beta_k = \frac{a_3^k d_1^k - a_2^k d_2^k}{a_1^k a_3^k - (a_2^k)^2}, \quad (39)$$

where $a_1^k := \mathbf{x}_k^T \mathbf{A} \mathbf{x}_k$, $a_2^k := \mathbf{r}_k^T \mathbf{A} \mathbf{x}_k$, $a_3^k := \mathbf{r}_k^T \mathbf{A} \mathbf{r}_k$, $d_1^k := \mathbf{b}^T \mathbf{x}_k$, $d_2^k := \mathbf{b}^T \mathbf{r}_k$. The trajectory (26) of \mathbf{x}_{k+1} generated by Eq. (37) with Eq. (39) will be kept on the invariant manifold if and only if

$$C_{k+1} = C_k + \frac{(a_1^k)^2 a_3^k + d_1^k [2(a_2^k)^2 + a_3^k d_1^k - 2a_2^k d_2^k] + a_1^k [-(a_2^k)^2 - 2a_3^k d_1^k + (d_2^k)^2]}{2[(a_2^k)^2 - a_1^k a_3^k]}. \quad (40)$$

Note that $Q_{k+1} = 1$ implied in Eq. (26). In order to speed up the rate of convergence, we stick to Eq. (40) but replace $Q_{k+1} = 1$ with the accelerator $Q_1 > 1$ and $Q_{k+1} > Q_k$. According to Eq. (40), we select a_1^k , a_2^k , a_3^k , d_1^k , and d_2^k to be the factors of the accelerator Q_{k+1} . Basing on our investigation, we find that with the best accelerator

$$Q_{k+1} = \exp \left[\frac{2\gamma [a_1^k a_3^k - (a_2^k)^2] (a_2^k d_1^k - a_1^k d_2^k)}{[\gamma^2 (a_2^k)^2 - a_1^k a_3^k] d_1^2 + (1 - \gamma^2) [2a_1^k a_2^k d_2^k - (a_1^k)^2 (d_2^k)^2]} \alpha_k \right] \quad (41)$$

containing a parameter $\gamma \in [0, 1)$, the extremal of the optimization problem

$$\min_{\beta_k, \alpha_k} Q_{k+1} \phi_{k+1}, \quad (42)$$

instead of Eq. (38), occurs at

$$\alpha_k = (1 - \gamma) \frac{a_2^k d_1^k - a_1^k d_2^k}{a_1^k d_3^k - (a_2^k)^2}, \quad \beta_k = \frac{\gamma a_2^k (a_1^k d_2^k - a_2^k d_1^k) + a_1^k (a_3^k d_1^k - a_2^k d_2^k)}{a_1^k [a_1^k d_3^k - (a_2^k)^2]}. \quad (43)$$

Following the discrete time dynamics of \mathbf{x}_{k+1} of Eq. (37) with Eq. (43), the \mathbf{x}_{k+1} evolves, generating the trajectory (21) on the invariant manifold if and only if C_{k+1} and Q_{k+1} satisfy Eqs. (40) and (41), respectively.

According to the above derivation, an iterative algorithm is proposed as below.

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

$$\mathbf{r}_k = \mathbf{A}\mathbf{x}_k - \mathbf{b}.$$

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

$$\begin{aligned} \alpha_k &= (1 - \gamma) \frac{(\mathbf{r}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{b}^T \mathbf{x}_k) - (\mathbf{x}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{b}^T \mathbf{r}_k)}{(\mathbf{x}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{r}_k^T \mathbf{A}\mathbf{r}_k) - (\mathbf{r}_k^T \mathbf{A}\mathbf{x}_k)^2}, \\ \beta_k &= \frac{\gamma \mathbf{r}_k^T \mathbf{A}\mathbf{x}_k [(\mathbf{x}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{b}^T \mathbf{r}_k) - (\mathbf{r}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{b}^T \mathbf{x}_k)] + \mathbf{x}_k^T \mathbf{A}\mathbf{x}_k [(\mathbf{r}_k^T \mathbf{A}\mathbf{r}_k)(\mathbf{b}^T \mathbf{x}_k) - (\mathbf{r}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{b}^T \mathbf{r}_k)]}{\mathbf{x}_k^T \mathbf{A}\mathbf{x}_k [(\mathbf{x}_k^T \mathbf{A}\mathbf{x}_k)(\mathbf{r}_k^T \mathbf{A}\mathbf{r}_k) - (\mathbf{r}_k^T \mathbf{A}\mathbf{x}_k)^2]}, \\ \mathbf{x}_{k+1} &= \beta_k \mathbf{x}_k - \alpha_k \mathbf{r}_k, \end{aligned}$$

and go to step (ii).

For convenience the new algorithm is referred to as the accelerated bidirectional method (A2DM), “accelerated” referring to $Q_{k+1} > 1$ and “bi-directional” to $\mathbf{x}_{k+1} \in \text{span}\{\mathbf{x}_k, \mathbf{r}_k\}$. When $\gamma = 0$ the accelerator is turned off, i.e. $Q_{k+1} = 1$, and hence the A2DM reduces to the bidirectional method (2DM). Table 1 summarizes the relations of the SDM, ASDM, 2DM, and A2DM.

4. Further Study by Numerical Experiments

To assess the performance of the newly developed ASDM, 2DM, and A2DM algorithms let us perform numerical experiments on three selected examples of ill-posed problems with condition numbers $\text{cond}(\mathbf{A}) = 1.59 \times 10^{13}$, 4.79×10^{18} , and 1.36×10^{19} in 2, 50, 100 dimensions, respectively. The numerical results are compared with those obtained from the steepest descent method (SDM), the Barzilai-Borwein method (BBM), and the random SDM with the intervals of $\theta_k \in [0, 2]$ (RSDM) and $\theta_k \in (0, 1)$ (RSDM1).

4.1 Example 1

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

$$\begin{bmatrix} 2 & 6 \\ 2 & 6.00001 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 8 \\ 8.00001 \end{bmatrix}. \quad (44)$$

The condition number is $\text{cond}(\mathbf{A}) = 1.59 \times 10^{13}$, where $\mathbf{A} = \mathbf{B}^T \mathbf{B}$ is positive definite and \mathbf{B} denotes the coefficient matrix, which is not symmetric, of Eq. (44). The exact solution is $(x^*, y^*) = (1, 1)$. The seven algorithms of SDM, BBM, RSDM, RSDM1, ASDM, 2DM, and A2DM are applied to find the solutions of Eq. (44), whose results are displayed correspondingly in Fig. 1 to Fig. 7.

To promote a better understanding of these methods and, in particular, the influence of initial guesses, we run the algorithms starting from various initial points $(x_0, y_0) = (1 + \cos \frac{i}{360}, 1 + \sin \frac{i}{360})$, $i = 1, 2, \dots, 360$, and stopped at the 19th step ($k = 19$). The iterations of 19 points (x_k, y_k) , $k = 0, 1, \dots, 19$, per initial guess for the 360 initial guesses were plotted in the (a) parts of the seven figures. In the (b) and (c) parts plotted are the residual errors $\mathbf{r}_{19}^T \mathbf{r}_{19}$ and the solution errors $\sqrt{(x_{19} - x^*)^2 + (y_{19} - y^*)^2}$, respectively, at step $k = 19$ for the 360 initial guesses.

Figure 1(a) shows the 360 \times 20 points of (x_k, y_k) during the 19 iterations by using the SDM for the 360 initial points on the circle $(x - 1)^2 + (y - 1)^2 = 1$, among which for illustration purposes, in particular, 2 initial points (x_0, y_0) and their first iteration points (x_1, y_1) are each marked by a cross and connected by thin lines to show their orbits. It is clearly seen that

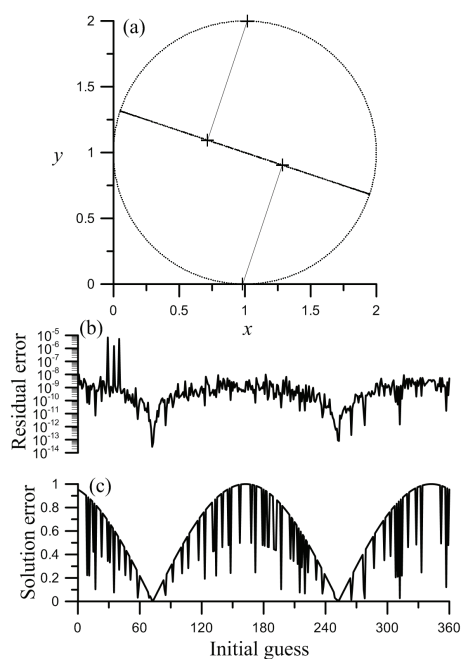


Figure 1. Example 1. Evolution of iterations, residual errors and solution errors for SDM

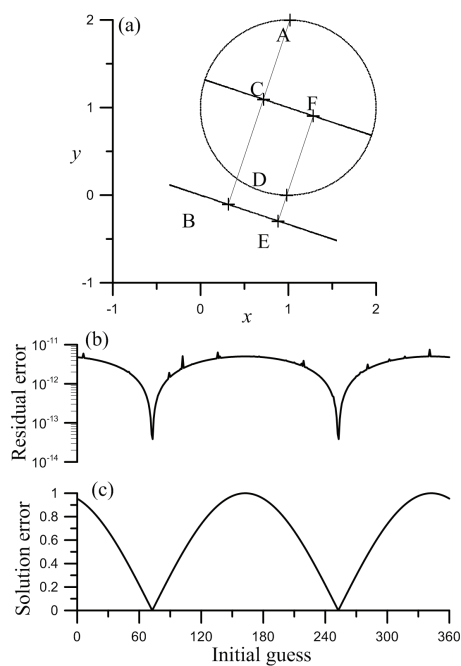


Figure 2. Example 1. Evolution of iterations, residual errors and solution errors for BBM

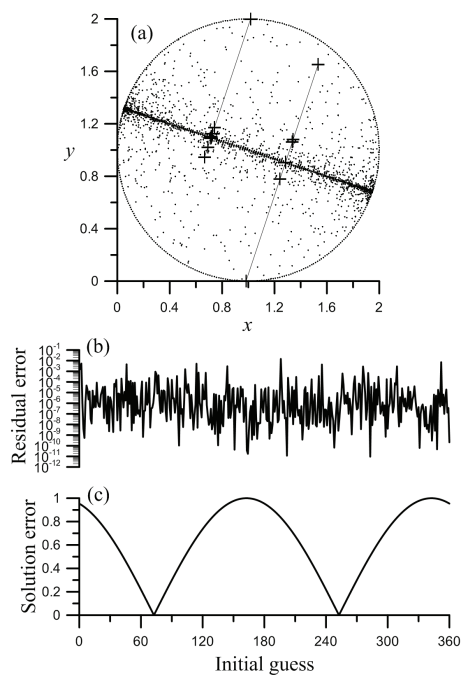


Figure 3. Example 1. Evolution of iterations, residual errors and solution errors for RSDM

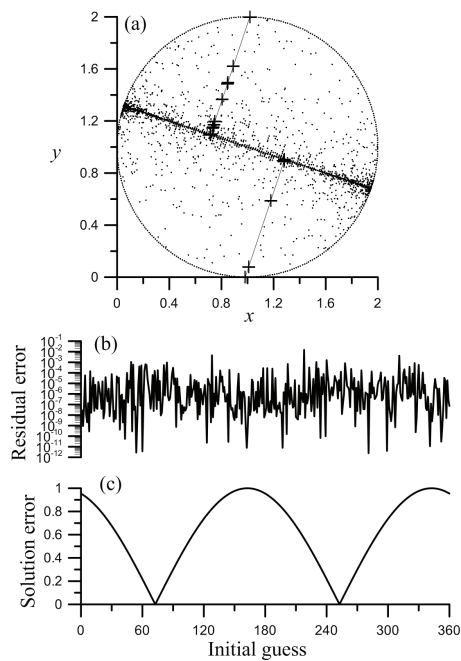


Figure 4. Example 1. Evolution of iterations, residual errors and solution errors for RSDM1

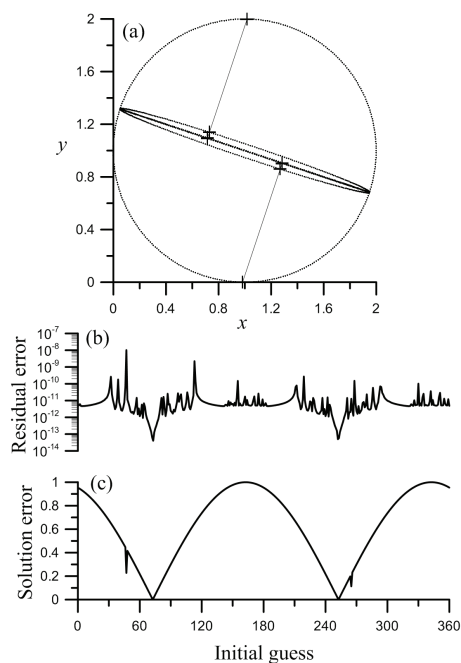


Figure 5. Example 1. Evolution of iterations, residual errors and solution errors for ASDM

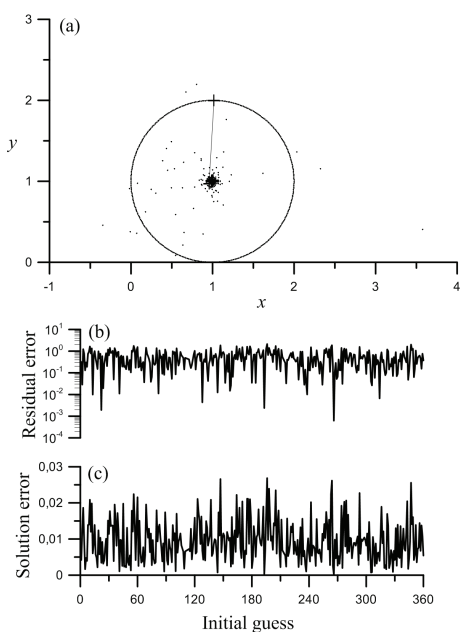


Figure 6. Example 1. Evolution of iterations, residual errors and solution errors for 2DM

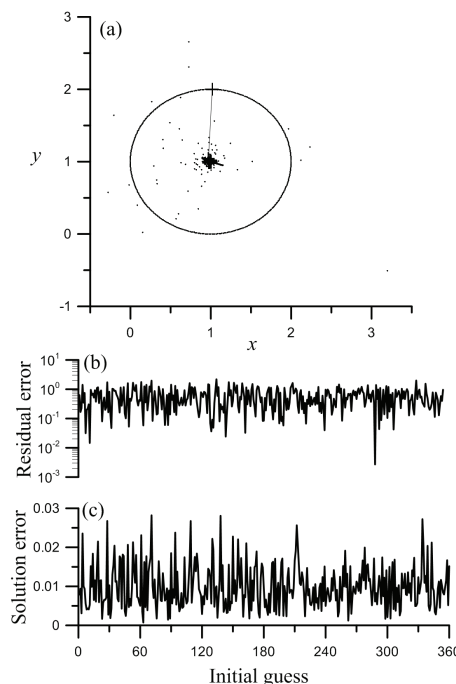


Figure 7. Example 1. Evolution of iterations, residual errors and solution errors for A2DM

all the first iterates (x_1, y_1) are already located nearly on the line $x + 3y = 4$. It is known that the line $x + 3y = 4$ is indeed the major axis of a family of extremely elongated ellipses which are the level curves of ϕ of the ill-posed problem (44). It looks like an extremely flat, narrow, steep-sided valley. In fact, all the first iteration points (x_1, y_1) are distributed nearly on the line $x + 3y = 4$ and all subsequent iteration points (x_k, y_k) , $k = 2, 3, \dots, 19$, are clustered around their own first iteration points. The latter fact is indeed the consequence of the property of the SDM whose gradient directions between two iterations are notoriously perpendicular so that the iterates turn directions right and left and hardly move on the flat valley $x + 3y = 4$.

Figure 1(b) shows that the residual errors of the SDM oscillate irregularly between 10^{-14} and 10^{-5} . The solution errors of the SDM is shown in Fig. 1(c); it is obvious that the initial guesses have structured influence on the solution errors.

By using the BBM, the evolution of iterations is shown in Fig. 2. It is interesting to observe that in Fig. 2(a) the 360 initial points (x_0, y_0) (including points A and D) originally situated on the circle $(x - 1)^2 + (y - 1)^2 = 1$ start to iterate so that the 360 first iteration points (x_1, y_1) are almost all on a line along \overline{BE} , then the 360 second iteration points (x_2, y_2) are almost all on a line along \overline{CF} , and then all the subsequent iterates are clustered around their own second iteration points nearly on the line along \overline{CF} . Two orbits $ABC \dots$ and $DEF \dots$ indicated by thin lines with cross marks are displayed in Fig. 2(a) for illustration purposes. Figures 2(b) and 2(c) show how the residual errors and solution errors change with the initial guesses. Observe that the style of the changing residual errors in Fig. 2(b) is very different from the corresponding one exhibited in the previous figure.

The evolution of iterations using the RSDM is plotted in Fig. 3(a). Since for RSDM the parameters θ_k are random numbers generated from the interval $[0, 2]$, observe that the iterations of (x_k, y_k) starting from the initial points $(1 + \cos \frac{i}{360}, 1 + \sin \frac{i}{360})$, $i < 180$, cross the line $x + 3y = 4$ and reach the lower left region $x + 3y < 4$, while the iterations starting from the initial points $(1 + \cos \frac{i}{360}, 1 + \sin \frac{i}{360})$, $i > 180$, conversely reach the upper right region $x + 3y > 4$. Figure 3(b) shows the residual errors varying between 10^{-10} and 10^{-2} . It seems that the initial guesses do not influence the trend of the residual errors. Figure 3(c) shows how systematically the solution errors are related to the initial guesses.

Different from the above using the RSDM, the results of evolution by using the RSDM1 are shown in Fig. 4(a). Since we choose $\theta_k \in (0, 1)$ for RSDM1, all iterations of (x_k, y_k) starting from the side $x + 3y > 4$ (resp. $x + 3y < 4$) remain inside the region $x + 3y \geq 4$ (resp. $x + 3y \leq 4$). Figure 4(b) shows the residual errors varying between 10^{-10} and 10^{-2} . Figure 4(c) demonstrates almost the same systematical way as Fig. 3(c) in which the solution errors depend on the initial guesses.

The results of the ASDM, 2DM, and A2DM are shown in Figs. 5, 6, and 7, respectively. We use $\gamma = 0.05$ for the accelerated algorithms of ASDM and A2DM. Figure 5(a) shows that all the 360 first iteration points (x_1, y_1) are located nearly on an ellipse inside the circle $(x - 1)^2 + (y - 1)^2 = 1$ and that all the second iteration points (x_2, y_2) are located

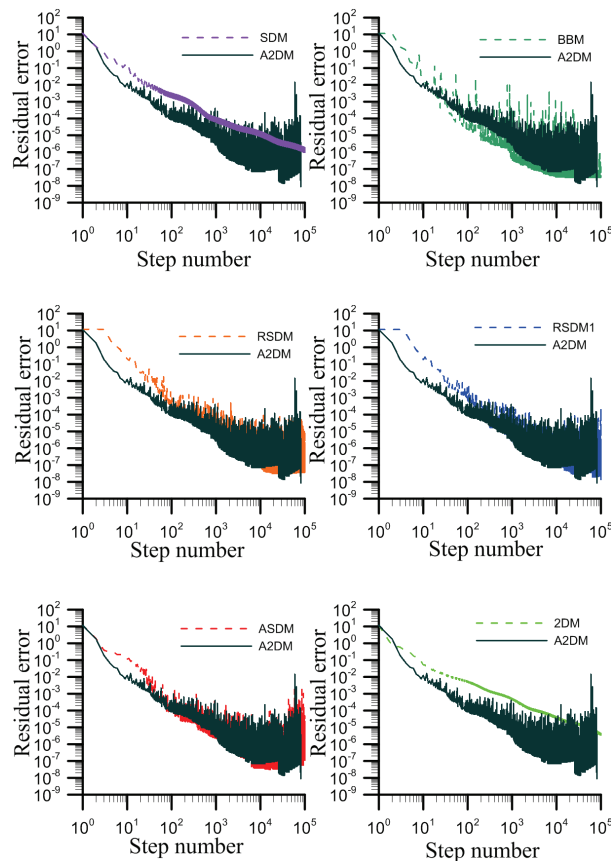


Figure 8. Example 2. Comparison of residual errors for A2DM with the other six methods

nearly on the line $x + 3y = 4$. It can be seen in Fig. 5(b) that the range of the residual errors is $(10^{-14}, 10^{-8})$ and in Fig. 5(c) how the solution errors depend on the initial guesses.

Up to now, we have explored the properties of the unidirectional methods. Now let us consider the bidirectional methods of 2DM and A2DM. In this two-dimensional case, the results of 2DM and A2DM are similar. Figures 6(a) and 7(a) show that almost all points are located very near to the solution point $(x^*, y^*) = (1, 1)$, besides the initial points. As can be seen in Figs. 6(c) and 7(c) the solution errors are smaller than those using the other five (unidirectional) methods. Although, on the other hand, the residual errors shown in Figs. 6(b) and 7(b) are greater than those using the others, this matter does not do harm but instead demonstrates the low reliability of the residual errors due to a large condition number as noted in (Nocedal *et al.*, 2002). Summarizing, for the ill-posed linear system (44), the A2DM and 2DM display their superior performance, with their iterations converging rapidly to the solution point $(x^*, y^*) = (1, 1)$ in less than 19 steps.

4.2 Example 2

In this example we consider a highly ill-posed linear equation (1) with \mathbf{A} being the Hilbert matrix of elements in Eq. (5). The ill-posedness of Eq. (1) increases fast with n .

In order to compare the numerical solutions with exact solutions we suppose that $x_1^* = x_2^* = \dots = x_n^* = 1$, and then by Eq. (5) we have

$$b_i = \sum_{j=1}^n \frac{1}{i+j-1}. \quad (45)$$

We solve this problem for the case with $n = 50$. The resulting linear system of equations is highly ill-posed, the condition number being as large as 4.79×10^{18} . Seven methods are applied to find the solutions of Eq. (1), whose results are displayed in Figs. 8 to 10. We used $\gamma = 0.15$ for A2DM and $\gamma = 0.05$ for ASDM.

The A2DM and 2DM have a privileged set of initial guesses $x_i = c, \forall c \in \mathbb{R}$ which happen to lie in a ray from the origin $x_i = 0$ through the solution point $x_i^* = 1$. If we start from these initial guesses, the A2DM and 2DM converge in only one step. Therefore, we chose another initial guess $x_i = (-1)^i \times 0.5, i = 1, 2, \dots, 50$, and set a stopping criterion $\varepsilon = 10^{-8}$. The

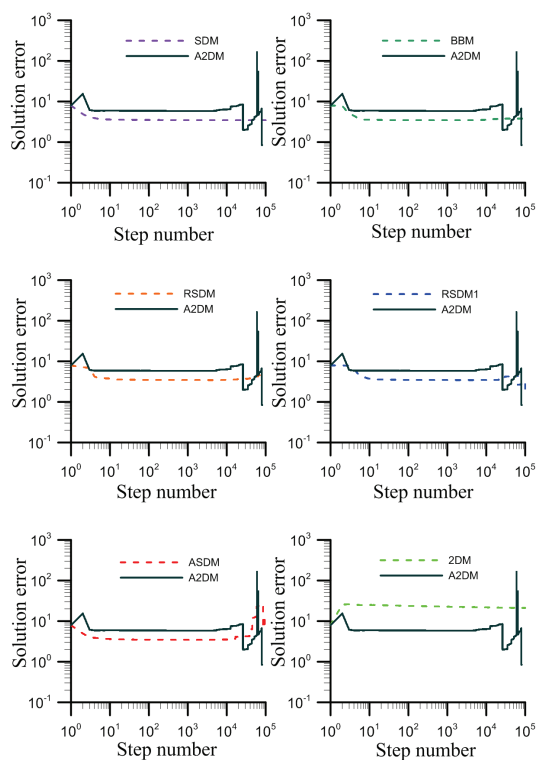


Figure 9. Example 2. Comparison of solution errors for A2DM with the other six methods

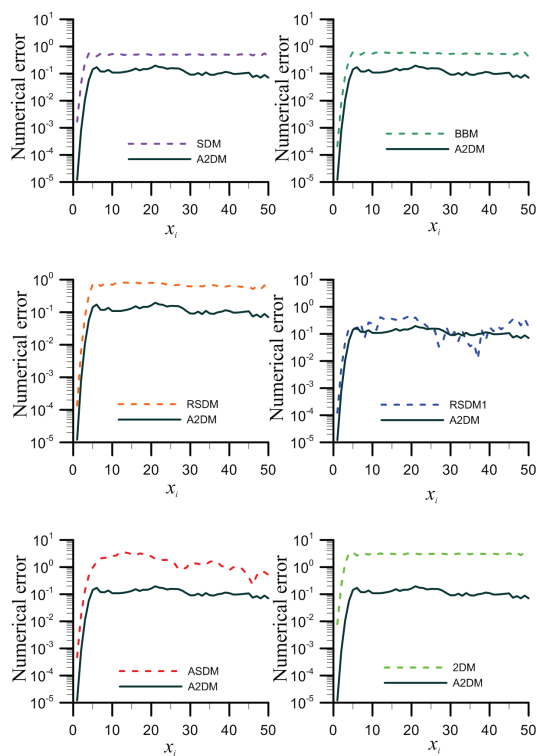


Figure 10. Example 2. Comparison of Numerical errors for A2DM with the other six methods

RSDM1 and the A2DM converge in 99985 iterations and 81660 iterations, respectively, but the others do not converge within 10^5 iterations. Figure 8 shows the residual errors $\|\mathbf{r}_k^T \mathbf{r}_k\|$ versus the step number k , the result of A2DM being compared with those of the other six methods in six plots. It is seen that A2DM has a faster convergence rate than the others. In this 50-dimensional case, 2DM does not work well as it did in the 2-dimensional case (Example 1). Observe that the residual errors of SDM and 2DM are changing smoothly and converging slowly, whereas those of ASDM, BBM, RSDM, RSDM1, and A2DM are changing violently to the contrary.

Figure 9 shows that the solution errors $\|\mathbf{x}_k - \mathbf{x}^*\|$ versus the step number k . The solution errors of all methods display nearly stationary during a relatively long period from step $k = 10^1$ to step $k = 10^4$. In this duration the performance of A2DM is merely better than 2DM. But beyond the 10^4 iterations, the solution errors of A2DM vary to a larger extent and succeed in attaining much lower values. Figure 10 displays the 50 components $|(\mathbf{x}_{N_f} - \mathbf{x}^*)_i|/(\mathbf{x}^*)_i$ of numerical errors at the final converged step N_f ; the distance between the final point and the solution point $x_i^* = 1$ for A2DM is the shortest. Both figures confirm that A2DM renders the most accurate results in the end.

4.3 Example 3

As an application of the new algorithms we consider a polynomial interpolation problem. In other words, given some data points, such as obtained by sampling of a measurement, the aim is to find a polynomial which goes exactly through these points.

Given a set of m data points (u_i, v_i) where no two u_i are the same, one is looking for a polynomial $p(u)$ of degree at most $m - 1$ with the following property:

$$p(u_i) = v_i, \quad i = 1, \dots, m, \quad (46)$$

where $u_i \in [a, b]$, $[a, b]$ being the spatial interval of the problem domain.

The unisolvence theorem states that such a polynomial $p(u)$ exists and is unique, and can be proved by using the Vandermonde matrix. Suppose that the interpolation polynomial is in the form of

$$p(u) = \sum_{i=1}^m x_i u^{i-1}, \quad (47)$$

where $1, u^1, \dots, u^{m-1}$ constitute a monomial basis. The statement that $p(u)$ interpolates the data points means that Eq. (46) must hold.

If we substitute Eq. (47) into Eq. (46), we obtain a system of linear equations in the unknown coefficients x_i ,

$$\begin{bmatrix} 1 & u_1 & u_1^2 & \dots & u_1^{m-2} & u_1^{m-1} \\ 1 & u_2 & u_2^2 & \dots & u_2^{m-2} & u_2^{m-1} \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 1 & u_{m-1} & u_{m-1}^2 & \dots & u_{m-1}^{m-2} & u_{m-1}^{m-1} \\ 1 & u_m & u_m^2 & \dots & u_m^{m-2} & u_m^{m-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{m-1} \\ x_m \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{m-1} \\ v_m \end{bmatrix}. \quad (48)$$

We have to solve the above system for x_i to construct the interpolant $p(u)$.

In order to compare the numerical solutions with exact solutions we suppose that $x_1^* = x_2^* = \dots = x_m^* = 1$. Then we obtain v_1, \dots, v_m according to Eq. (48). In this case we take $m = 100$ and $u_i = -1 + 2i/100$ to be the nodal points. The condition number of this system is $\text{cond}(\mathbf{A}) = 1.36 \times 10^{19}$ where $\mathbf{A} = \mathbf{B}^T \mathbf{B}$ and \mathbf{B} denotes the coefficient matrix.

Seven methods are applied to find the solutions of Eq. (48), whose results are displayed in Figs. 11 to 13. As in the last example, the A2DM and 2DM have a privileged set of initial guesses $x_i = c, \forall c \in \mathbb{R}$ which happen to lie in a ray from the origin $x_i = 0$ through the solution point $x_i^* = 1$. If starting from these initial guesses, the A2DM and 2DM would converge in only one step. Therefore, we start from, instead, a fairly ordinary initial guess $x_i = (-1)^i \times 0.5, i = 1, 2, \dots, 100$, and set a stopping criterion $\epsilon = 10^{-8}$. All the methods do not converge in 10^5 iterations. The comparison of the residual errors for these methods are shown in Fig. 11. The residual errors of A2DM plotted in the six charts with the other six methods exhibit faster convergence rate almost all the way in the whole iterative process. It can also be seen that ASDM performs better than BBM, RSDM, and RSDM1.

It is interesting to examine more closely the behavior of the residual errors exhibited in Fig. 11 of this example as well as Fig. 8 of the previous example. We observe that SDM and 2DM have similar trends of smoother and slower decreasing rates, but the others exhibit more violently oscillating behavior. These common phenomena for the BBM, RSDM, RSDM1, ASDM, and A2DM may be attributed to their steplength modifications by a multiplicative scalar factor, namely $\|\mathbf{r}_k\|^2 \|\Delta \mathbf{r}_{k-1}\|^2 / [\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k (\Delta \mathbf{r}_{k-1})^T \Delta \mathbf{x}_{k-1}]$ in BBM, θ_k in RSDM and RSDM1, and $1 - \gamma$ (and hence Q_k) in ASDM and

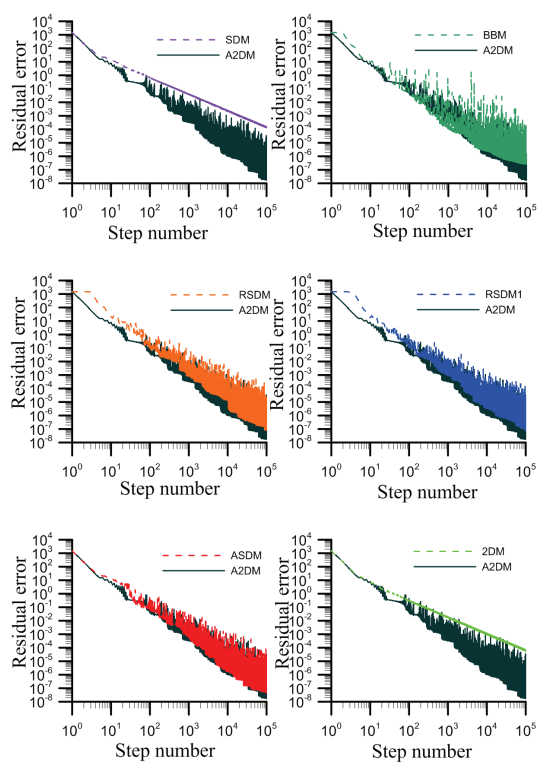


Figure 11. Example 3. Comparison of residual errors for A2DM with the other six methods

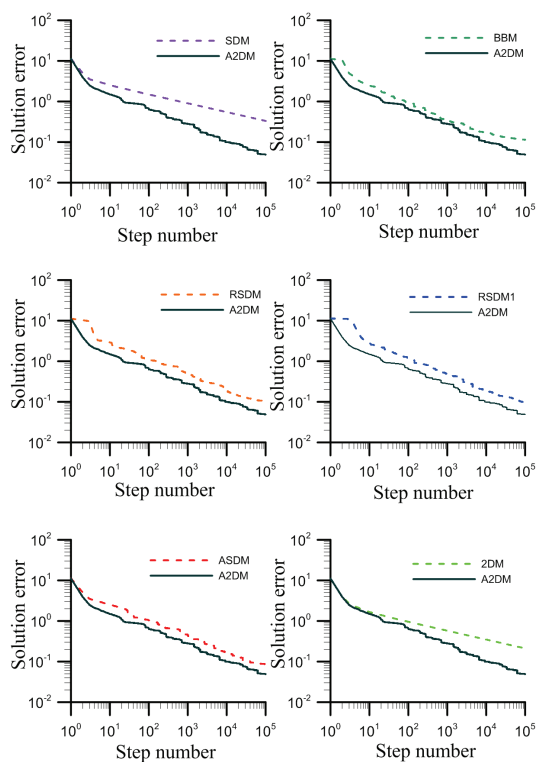


Figure 12. Example 3. Comparison of solution errors for A2DM with the other six methods

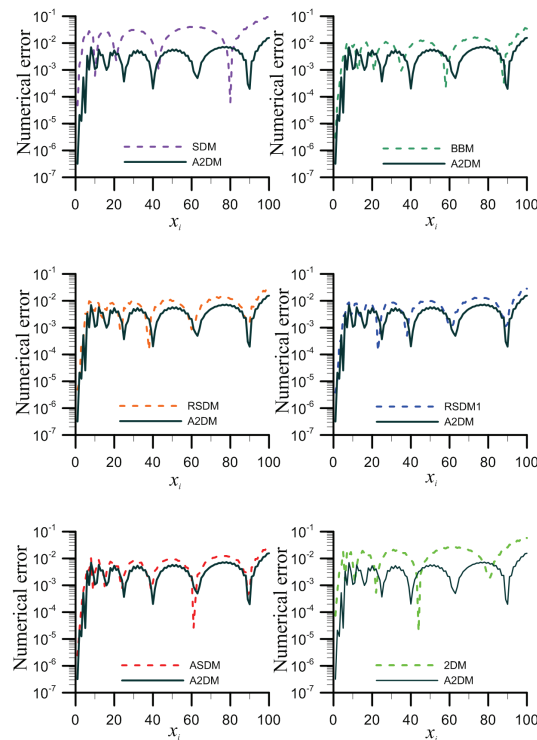


Figure 13. Example 3. Comparison of Numerical errors for A2DM with the other six methods

A2DM. It is observed, in particular, that the accelerator $Q_1 > 1$ and $Q_{k+1} > Q_k$ plays a prominent role of switching from the situation of slow convergence to a new situation that the functional ϕ_{k+1} tends to decrease stepwise in an intermittent and ceaseless manner.

Since we know the exact solution of problem (48), we plot the solution errors $\|\mathbf{x}_k - \mathbf{x}^*\|$ of these seven methods for all steps k in Fig. 12. The result of A2DM is compared with those of the other six methods in the six plots. The solution errors using A2DM converge faster than those using the others nearly at all steps k . Similarly, the numerical errors of A2DM are smaller than those of the others as can be seen in Fig. 13.

5. Concluding Remarks

We have proposed accelerated and/or bidirectional modifications of the classical steepest descent method (SDM), namely the accelerated SDM (ASDM), the bidirectional method (2DM), and the accelerated 2DM (A2DM). The proposed methods are all furnished with dynamics-theoretical and optimization interpretation. By embedding the minimization problem of a quadratic functional into an invariant manifold in the product space of the state space and the fictitious time axis, we have constructed a continuous time dynamical system for the unknown vector \mathbf{x} in the state space. Using different discretization schemes, we have constructed the invariant manifolds of the unidirectional method and the 2DM. Locally they pursue optimal line search and optimal plane search, respectively. The unidirectional method is nothing but the classical SDM. On the basis of dynamics of trajectories we have equipped SDM and 2DM with accelerators $Q_0 > 1$ and $Q_{k+1} > Q_k$ in order to improve the convergence rate and have thus developed the accelerated algorithms of ASDM and A2DM through optimization processes.

The iterative algorithms so developed retain the simplicity of the classical SDM algorithm and are robust and easy to implement. Moreover, both the operations of multiplication and addition are of the same order of N^2 for the SDM, ASDM, 2DM, and A2DM in N -dimensional problems. Comparison of SDM, BBM, RSDM, RSDM1, ASDM, 2DM, and A2DM methods are made and demonstrated in the numerical experiments of three highly ill-posed problems.

In Example 1, we have observed that the initial guesses have not structured influence for the 2DM and A2DM and their iterations converge rapidly to the solution point. For the example of Hilbert matrix, the RSDM1 and A2DM converge within 10^5 iterations under the stopping criterion $\varepsilon = 10^{-8}$ but the A2DM shows the lower value of solution error in the end. Even though all the methods do not converge in 10^5 iterations with the stopping criterion $\varepsilon = 10^{-8}$, the A2DM exhibits faster convergence rate almost all the way in the whole iterative process of Example 3.

The A2DM consequently displays better performance in all cases. The 2DM works well merely in the 2-dimensional case. The ASDM is a second choice besides the A2DM. According to the numerical experiments, the bidirectional methods of 2DM and A2DM (the second column in Table 1) display better efficiency and accuracy than the unidirectional methods of SDM and ASDM (the first column in Table 1), and the accelerated methods of ASDM and A2DM (the second row in Table 1) work better than the original methods (the first row in Table 1).

Table 1. Classification of the proposed modifications

	unidirectional method	bidirectional method
———	SDM	2DM
accelerated	ASDM	A2DM

The merits of A2DM are (i) the point of view of global dynamics (virtual time method \cong continuous time dynamics and iterative algorithm \cong discrete time dynamics) with the $Q_0 > 1$ and $Q_{k+1} > Q_k$ as an accelerator and (ii) both the steplength and direction at each step being determined locally by an optimization in a two dimensional subspace without sophisticated computation.

Acknowledgement

The research is financed by Ministry of Science and Technology of Taiwan (MOST 104-2218-E-002-026-MY3 and MOST 103-2221-E-002-244-MY2).

References

- Akaike, H. (1959). On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. *Ann. Inst. Statist. Math.*, 11, 38-43. <http://dx.doi.org/10.1007/BF01831719>
- Barzilai, J., & Borwein, J. M. (1988). Two point step size gradient methods. *IMA J. Numer. Anal.*, 8, 141-148. <http://dx.doi.org/10.1093/imanum/8.1.141>
- Dai, Y. H., & Liao, L. H. (2002). R -linear convergence of the Barzilai and Borwein gradient method. *IMA J. Numer. Anal.*, 22, 1-10. <http://dx.doi.org/10.1093/imanum/22.1.1>
- Dai, Y. H., & Yuan, Y. (2003). Alternate minimization gradient method. *IMA J. Numer. Anal.*, 23, 377-393. <http://dx.doi.org/10.1093/imanum/23.3.377>
- Dai, Y. H., Yuan, J. Y., & Yuan, Y. (2002). Modified two-point stepsize gradient methods for unconstrained optimization. *Comp. Optim. Appl.*, 22, 103-109. <http://dx.doi.org/10.1023/A:101483841961>
- Fletcher, R. (2005). On the Barzilai-Borwein gradient method. In Qi, L., Teo, K., & Yang, X. (Ed.), *Optimization and Control with Applications*, New York: Springer.
- Friedlander, A., Martinez, J. M., Molina, B., & Raydan, M. (1999). Gradient method with retards and generalizations. *SIAM J. Numer. Anal.*, 36, 275-289. <http://dx.doi.org/10.1137/S003614299427315X>
- Jacoby, S. L. S., Kowalik, J. S., & Pizzo, J. T. (1972). *Iterative Methods for Nonlinear Optimization Problems*. New Jersey: Prentice-Hall.
- Kunisch, K., & Zou, J. (1998). Iterative choices of regularization parameters in linear inverse problems. *Inverse Problems*, 14, 1247-1264. <http://dx.doi.org/10.1088/0266-5611/14/5/010>
- Liu, C. S. (2007a). A highly accurate solver for the mixed-boundary potential problem and singular problem in arbitrary plane domain. *Comp. Model. Eng. Sci.*, 20, 111-122. <http://dx.doi.org/10.3970/cmes.2007.019.145>
- Liu, C. S. (2007b) A modified Trefftz method for two-dimensional Laplace equation considering the domain's characteristic length. *Comp. Model. Eng. Sci.*, 21, 53-66. <http://dx.doi.org/10.3970/cmes.2007.021.053>
- Liu C. S. (2007c). An effectively modified direct Trefftz method for 2D potential problems considering the domain's characteristic length. *Eng. Anal. Bound. Elem.*, 31, 983-993. <http://dx.doi.org/10.1016/j.enganabound.2007.04.006>
- Liu, C. S., & Atluri, S. N. (2009a). 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. *Comp. Model. Eng. Sci.*, 41, 243-261. <http://dx.doi.org/10.3970/cmes.2009.041.243>
- Liu, C. S., & Atluri, S. N. (2009b). A highly accurate technique for interpolations using very high-order polynomials, and its applications to some ill-posed linear problems. *Comp. Model. Eng. Sci.*, 43, 253-276.

<http://dx.doi.org/10.3970/cmcs.2009.043.253>

- Liu, C. S., & Chang, C. W. (2009). Novel methods for solving severely ill-posed linear equations system. *J. Marine Sci. Tech.*, 17, 216-227.
- Liu, C. S., Hong, H. K., & Atluri, S. N. (2010). Novel algorithms based on the conjugate gradient method for inverting ill-conditioned matrices, and a new regularization method to solve ill-posed linear systems. *Comp. Model. Eng. Sci.*, 60, 279-308. <http://dx.doi.org/10.3970/cmcs.2010.060.279>
- Liu, C. S. (2012). Modifications of steepest descent method and conjugate gradient method against noise for ill-posed linear dystems. *Commun. Numer. Anal.* Article ID cna-00115, 24 Pages. <http://dx.doi.org/10.5899/2012/cna-00115>
- Lorentz, E., & Benallal, A. (2005). Gradient constitutive relations: numerical aspects and application to gradient damage. *Comput. Methods Appl. Mech. Engrg.*, 194, 5191-5220. <http://dx.doi.org/10.1016/j.cma.2004.12.016>
- Nocedal, J., Sartenauer, A., & Zhu, C. Y. (2002). On the behavior of the gradient norm in the steepest descent method. *Comput. Optim. Appl.*, 22, 5-35. <http://dx.doi.org/10.1023/A:1014897230089>
- Ostrowski, A. M. (1973). *Solution of equations in Euclidean and Banach spaces*, New York: Academic Press.
- Raydan, M. (1993). On the Barzilai and Borwein choice of steplength for the gradient method. *IMA J. Numer. Anal.*, 13, 321-326. <http://dx.doi.org/10.1093/imanum/13.3.321>
- Raydan, M. (1997). The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. *SIAM J. Optim.*, 7, 26-33. <http://dx.doi.org/10.1137/S1052623494266365>
- Raydan, M., & Svaiter, B. F. (2002). Relaxed steepest descent and Cauchy-Barzilai-Borwein method. *Comput. Optim. Appl.*, 21, 155-167. <http://dx.doi.org/10.1023/A:1013708715892>
- Resmerita, E. (2005). Regularization of ill-posed problems in Banach spaces: convergence rates. *Inverse Problems*, 21, 1303-1314. <http://dx.doi.org/10.1088/0266-5611/21/4/007>
- Stewart, G. (1973). *Introduction to Matrix Computations*, New York: Academic Press.
- Tikhonov, A. N., & Arsenin, V. Y. (1977). *Solutions of Ill-Posed Problems*, New York: John-Wiley & Sons.
- Wang, Y., & Xiao, T. (2001). Fast realization algorithms for determining regularization parameters in linear inverse problems. *Inverse Problems*, 17, 281-291. <http://dx.doi.org/10.1088/0266-5611/17/2/308>
- Xie, J., & Zou, J. (2002). An improved model function method for choosing regularization parameters in linear inverse problems. *Inverse Problems*, 18, 631-643. <http://dx.doi.org/10.1088/0266-5611/18/3/307>
- Yuan, Y. (2006). A new stepsize for the steepest descent method, *J. Comput. Math.*, 24, 149-156.

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).