Research Article

A Three-Step Iterative Method for Solving Absolute Value Equations

Jing-Mei Feng ^{1,2} and San-Yang Liu ¹

¹School of Mathematics and Statistics, Xidian University, Xi'an 710071, China

²School of Engineering and Architecture, Shaanxi Radio and TV University, Xi'an 710119, China

Correspondence should be addressed to San-Yang Liu; liusanyang@126.com

Received 12 May 2020; Accepted 17 June 2020; Published 25 July 2020

Academic Editor: Nan-Jing Huang

Copyright © 2020 Jing-Mei Feng and San-Yang Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we transform the problem of solving the absolute value equations (AVEs) $Ax - |x| = b$ with singular values of A greater than 1 into the problem of finding the root of the system of nonlinear equation and propose a three-step algorithm for solving the system of nonlinear equation. The proposed method has the global linear convergence and the local quadratic convergence. Numerical examples show that this algorithm has high accuracy and fast convergence speed for solving the system of nonlinear equations.

1. Introduction

In the last few decades, the system of absolute value equations has been recognized as a kind of NP-hard and nondifferentiable problem, which can be equivalent to many mathematical problems, such as generalized linear complementarity problem, bilinear programming problem [1], knapsack feasibility problem [2], traveling agent problem [3], etc. . . However, compared with the above problems, it has the characteristics of simple structure and easy solution, which attracts much attention. This paper mainly designed a three-step iterative algorithm to solve the absolute value equations (AVEs) of the following type:

$$Ax - |x| = b, \quad (1)$$

where $A \in R^{n \times n}$, $b \in R^n$, and $|\cdot|$ indicates the absolute value. In [4], some theoretical conclusions about the solutions of AVEs (1) are given, and the sufficient conditions for the existence of unique solutions, nonnegative solutions, 2^n solutions, and no solutions are given for the first time. Next, many scholars began to design effective algorithms to solve equations with different types of solutions. Mangasarian [1] proposed a generalized Newton method for solving AVEs (1) when it has a unique solution; then

100 AVEs (1) of 1000 dimensions generated randomly are used to test effectiveness of this method, and the accuracy is up to 10^{-6} . On this basis, an improved generalized Newton algorithm is proposed by adding a dynamic step size [5], and the accuracy of the solution is of 10^{-9} . Feng and Liu [6] presented a two-step iterative algorithm which decreases the probability of the occurrence of poor solutions in the first iteration, thus improving the global convergence of the whole algorithm. A new method for solving the system of absolute value equations is proposed [7] by the preconditioning technique and successive over-relaxation. It was inspired by the Picard-Hermitian and skew-Hermitian splitting iteration method to solve the system nonlinear equations. Salkuyeh [8] presented the Picard-HSS iteration method for solving the system of absolute value equations, and there are also two CSCS-based iteration methods [9] and other iterative algorithms [10–12]. In addition, some iterative algorithms for solving the system of nonlinear equations are also interesting, for example, many three-step higher-order iterative algorithms are designed to solve the system nonlinear equations [13], Cordero et al. proposed two iterative algorithms with fourth-order and fifth-order convergences to solve the system nonlinear equations in [14].

Considering the nondifferentiation of absolute value equation, taking the advantages of the above algorithms and overcoming their disadvantages, a three-step iterative algorithm is designed for effectively solving AVEs (1) in this paper. Firstly, we give a three-step iterative formula, which absorbs the advantages of both the generalized classic Newton method and higher-order iterative algorithm; it has the advantages of small error and fast iteration speed. Then, we can prove that this method can linearly converge to the global optimal solution of AVEs (1) and possess locally quadratic convergence. Finally, some numerical experiments and comparison with both the classic generalized Newton method [1] and two-step iterative algorithm [6] show that our method has high accuracy and fast convergence speed.

The rest of this article is arranged as follows. Section 2 gives the preliminary basic knowledge about AVEs (1). We give a three-step iterative algorithm and prove the global convergence and local quadratic convergence of the proposed method in Section 3. In Section 4, we present some numerical experiments. Some concluding remarks to end the paper in Section 5.

2. Preliminaries

We are now describing some symbols and backgrounds. Let e and I be a unit vector and a unit matrix, respectively. $\|x\|$ is the two-norm $(x^T x)^{1/2}$, and $\text{sign}(x)$ is a vector with components equal to 1, 0, or -1 depending on whether the corresponding component of x is positive, zero, or negative. In addition, $\text{diag}(\text{sign}(x))$ is a diagonal matrix whose diagonal elements are the components of $\text{sign}(x)$. The diagonal matrix $D(x)$ represents a generalized Jacobian matrix of $|x|$:

$$D(x) = \partial|x| = \text{diag}(\text{sign}(x)). \quad (2)$$

We define a nonlinear function $f(x)$ as follows:

$$f(x) = Ax - |x| - b. \quad (3)$$

A generalized Jacobian of $f(x)$ at x is

$$\partial f(x) = A - D(x), \quad (4)$$

where $D(x) = \text{diag}(\text{sign}(x))$.

For solving AVEs (1), a generalized Newton iteration algorithm was presented in [1] and the iteration formula is as follows:

$$x^{k+1} = (A - D(x^k))^{-1} b, \quad k = 0, 1, 2, \dots \quad (5)$$

Effectiveness of this method was tested by the 100 randomly generated 1,000-dimensional AVEs (1), when the singular values of A are not less than 1, and the accuracy is up to 10^{-6} .

A two-step iterative method for solving AVEs (1) is presented in [6] and the iterative formula is

$$\begin{aligned} y^k &= x^k + (A - D(x^k))^{-1} f(x^k), \\ x^{k+1} &= x^k - (A - D(x^k))^{-1} (f(y^k) - f(x^k)), \end{aligned} \quad (6)$$

$k = 0, 1, 2, \dots$

Example in [1] is used to verify the effectiveness of this algorithm. Compared with generalized Newton method, the convergence speed is faster and the solution accuracy is improved.

For solving the system of nonlinear equations $g(x) = 0$, where $g: D \subset \mathbb{R} \rightarrow \mathbb{R}$, many methods are given in [13–15], these methods also give us a lot of good ideas. Taking some of the advantages of the above approach, considering the particularity of the system of absolute value equations, this paper designed the following three-step iterative algorithm for AVEs (1).

3. Three-Step Iterative Algorithm

In this section, a three-step iterative algorithm is presented in the following form:

$$\begin{aligned} y^k &= x^k - (A - D(x^k))^{-1} f(x^k), \\ z^k &= y^k - \frac{(A - D(x^k))^{-1} + (A - D(y^k))^{-1}}{2} f(y^k), \\ x^{k+1} &= z^k - (A - D(z^k))^{-1} f(z^k), \end{aligned} \quad (7)$$

$$k = 0, 1, 2, \dots$$

Based on this iterative formula (7), we design a new algorithm for solving AVEs (1).

The algorithm steps and the specific process are shown as follows:

- Step 1. Randomly generated an initial vector $x^k \in \mathbb{R}^n$ to AVEs (1), set $k = 0$;
- Step 2. Compute x^{k+1} by the (7);
- Step 3. If $\|Ax^{k+1} - |x^{k+1}| - b\| = 0$, stop, otherwise go to step 4;
- Step 4. Set $k := k + 1$, go to Step 2.

The global linear convergence and the local quadratic convergence of the Algorithm 1 are given below.

Lemma 1 (see [4]). *The minimum singular value of the matrix $A \in \mathbb{R}^{n \times n}$ is greater than 1 if and only if all the eigenvalues of $A^T A$ are greater than 1.*

Lemma 2 (see [4]). *If the minimum singular value of $A \in \mathbb{R}^{n \times n}$ exceeds 1 for the algorithm (6), then for any diagonal matrix D whose diagonal elements $D_{ii} = +1, 0, -1, i = 1, 2, \dots, n$, $(A - D)^{-1}$ exists.*

Lemma 3 (see [1]). *Lipschitz continuity of the absolute value. Let the vectors $x, y \in \mathbb{R}^n$, then*

$$\||x| - |y|\| \leq 2 \|x - y\|. \quad (8)$$

Theorem 1. *If the minimum singular value of the matrix A is greater than 1 and $D(x^{k+1}) = D(z^k)$ for some k for the three-step algorithm (6), then x^{k+1} is a solution to the AVEs (1).*

(1) **Initialization:**
An initial vector:
 $x^k \in R^n, k = 0$.
(2) **while** $\|Ax^k - |x^k| - b\| \geq \varepsilon$ **do**
(3) By (7), calculate x^{k+1} .
(4) By (3), calculate $f(x^{k+1})$.
(5) $k = k + 1$.
(6) **end while**

ALGORITHM 1: Three-step iterative algorithm.

Proof. Let's first simplify $x^{k+1} = z^k - (A - D(z^k))^{-1} f(z^k)$ in formula (7),

$$\begin{aligned}
 x^{k+1} &= z^k - (A - D(z^k))^{-1} f(z^k) \\
 &= z^k - (A - D(z^k))^{-1} (Az^k - |z^k| - b) \\
 &= z^k - (A - D(z^k))^{-1} (Az^k - D(z^k)z^k - b) \\
 &= z^k - (A - D(z^k))^{-1} ((A - D(z^k))z^k - b) \\
 &= z^k - (A - D(z^k))^{-1} (A - D(z^k))z^k + (A - D(z^k))^{-1} b \\
 &= z^k - z^k + (A - D(z^k))^{-1} b \\
 &= (A - D(z^k))^{-1} b.
 \end{aligned} \tag{9}$$

Then, $(A - D(z^k))x^{k+1} = b$, so,

$$\begin{aligned}
 0 &= (A - D(z^k))x^{k+1} - b = (A - D(z^{k+1}))x^{k+1} - b \\
 &= Ax^{k+1} - D(x^{k+1})x^{k+1} - b = Ax^{k+1} - |x^{k+1}| - b.
 \end{aligned} \tag{10}$$

Hence, x^{k+1} is the solution to the AVEs (1).

Theorem 2 (global linear Convergence). *If $\exists K_0$, make $D(x^k) = D(y^k), k > K_0$, and $\|(A - D)^{-1}\| < 1/3$ for any diagonal matrix D whose diagonal elements are ± 1 or 0 , then the three-step iterative algorithm (6) from any initial point x^0 converges linearly to a solution $\bar{x}$ for any solvable AVEs (1).*

Proof. Suppose that $\bar{x}$ is a solution of the AVE (1), noting that $D(\bar{x})\bar{x} = |\bar{x}|$ and $D(x^k)x^k = |x^k|$, for simplicity, let $\bar{D} = D(\bar{x}), D^k = D(x^k)$.

The proof of Theorem 1 shows that the first one $y^k = x^k - (A - D(x^k))^{-1} f(x^k)$ in formula (7) can be reduced to $y^k = (A - D(x^k))^{-1} b = (A - D^k)^{-1} b$, that is, $(A - D^k)y^k = b$.

Because $(A - \bar{D})\bar{x} = b$, let us take $(A - \bar{D})\bar{x} = b$ minus $(A - D^k)y^k = b$,

$$\begin{aligned}
 A(y^k - \bar{x}) &= D^k y^k - \bar{D}\bar{x} \\
 &= D^k(y^k - x^k + x^k) - \bar{D}\bar{x} \\
 &= D^k(y^k - x^k) + D^k x^k - \bar{D}\bar{x} \\
 &= |x^k| - |\bar{x}| + D^k(y^k - x^k) \\
 &= |x^k| - |\bar{x}| + D^k(y^k - \bar{x} + \bar{x} - x^k) \\
 &= |x^k| - |\bar{x}| + D^k(y^k - x^k) + D^k(\bar{x} - x^k).
 \end{aligned} \tag{11}$$

So,

$$(A - D^k)(y^k - \bar{x}) = |x^k| - |\bar{x}| + D^k(\bar{x} - x^k). \tag{12}$$

Therefore,

$$(y^k - \bar{x}) = (A - D^k)^{-1} (|x^k| - |\bar{x}| + D^k(\bar{x} - x^k)). \tag{13}$$

According to Lemma 3 and given conditions,

$$\begin{aligned}
 \|y^k - \bar{x}\| &= \|(A - D^k)^{-1} (|x^k| - |\bar{x}| + D^k(\bar{x} - x^k))\| \\
 &\leq \|(A - D^k)^{-1}\| (\| |x^k| - |\bar{x}| \| + \|D^k(\bar{x} - x^k)\|) \\
 &\leq \|(A - D^k)^{-1}\| (2\|x^k - \bar{x}\| + \|\bar{x} - x^k\|) \\
 &\leq \|(A - D^k)^{-1}\| (3\|x^k - \bar{x}\|) \\
 &< \|x^k - \bar{x}\|.
 \end{aligned} \tag{14}$$

Because $k > K_0$, $D(x^k) = D(y^k)$, the second one

$$z^k = y^k - \frac{(A - D(x^k))^{-1} + (A - D(y^k))^{-1}}{2} f(y^k), \tag{15}$$

in formula (7) can be simplified to the following form:

$$z^k = y^k - (A - D(y^k))^{-1} f(y^k) = (A - D(y^k))^{-1} b, \tag{16}$$

In the same way,

$$\|z^k - \bar{x}\| < \|y^k - \bar{x}\|. \tag{17}$$

Now let us simplify the third one $x^{k+1} = z^k - (A - D(z^k))^{-1} f(z^k)$ in formula (7) to

$$x^{k+1} = (A - D(z^k))b. \tag{18}$$

Also get,

$$\|x^{k+1} - \bar{x}\| < \|z^k - \bar{x}\|. \tag{19}$$

Hence,

$$\|x^{k+1} - \bar{x}\| < \|z^k - \bar{x}\| < \|y^k - \bar{x}\| < \|x^k - \bar{x}\|. \tag{20}$$

So the $\|x^{k+1} - \bar{x}\|$ converges to zero, the sequence iterated by formula (7) linearly converges to a solution $\bar{x}$. $\square$

Theorem 3 (locally quadratic convergence). *For any diagonal matrix D with diagonal elements being ± 1 or 0 , if $(A - D)$ is nonsingular, then the sequence $\{x^{k+1}\}$ from three-step iterative algorithm (6) converges to $\bar{x}$ and $\|x^{k+1} - \bar{x}\| = O(\|x^k - \bar{x}\|^2)$.*

Proof (see [16]). We omit it here.

4. Numerical Examples

In this section, six AVEs (1) examples are worked out to illustrate the feasibility and effectiveness of the proposed methods. All the numerical computations are performed

TABLE 1: The comparison of three-SIA, two-SIA, and GNM for Example 1.

Algorithm	X_1	X_2	X_3	$R_1; R_2$
GNM	$x_1 = -1.040901213171577$ $x_2 = 1.002079722703640$	$x_1 = -1.0000000000000000$ $x_2 = 1.0000000000000000$	$x_1 = -1$ $x_2 = 1$	2.081802426343145; 0
Two-SIA	$x_1 = -1.002171578270887$ $x_2 = 0.991745740048690$	$x_1 = -1$ $x_2 = 1$	$x_1 = -1$ $x_2 = 1$	1.267518301536992; 0
Three-SIA	$x_1 = -1$ $x_2 = 1$	$x_1 = -1$ $x_2 = 1$	$x_1 = -1$ $x_2 = 1$	0; 0

in MATLAB R2017a. We compare the proposed method (Three-SIA) with the two-step iterative algorithm [6] (Two-SIA) and the generalized Newton's method [1] (GNM).

Example 1

$$A = \begin{pmatrix} 50 & 2 \\ 3 & 60 \end{pmatrix}, \quad (21)$$

$$b = \begin{pmatrix} -49 \\ -56 \end{pmatrix},$$

The initial value is $x_0 = \text{rand}(2, 1)$, and the exact solution to this problem is $x_1 = -1, x_2 = 1$.

Example 2

$$A = \begin{pmatrix} 4.3244 & 1.1302 & 0.9364 \\ 1.1302 & 4.6114 & 0.8643 \\ 0.9364 & 0.8643 & 3.8834 \end{pmatrix}, \quad (22)$$

$$b = \begin{pmatrix} 44.4703 \\ 61.5432 \\ 79.1937 \end{pmatrix},$$

The initial value is $x_0 = \text{rand}(3, 1)$, and the exact solution to this problem is $x_1 = 3.2697, x_2 = 10.4486, x_3 = 23.2716$.

Example 3

$$A = \begin{pmatrix} 10 & 1 & 2 & 0 \\ 1 & 11 & 3 & 1 \\ 0 & 2 & 12 & 1 \\ 1 & 7 & 0 & 13 \end{pmatrix}, \quad (23)$$

$$b = \begin{pmatrix} 12 \\ 15 \\ 14 \\ 20 \end{pmatrix},$$

The initial value is $x_0 = \text{rand}(4, 1)$, and the exact solution to this problem is $x_1 = x_2 = x_3 = x_4 = 1$.

Example 4

$$\begin{aligned} n &= 5; \\ \text{rand}('state', 0); \\ R &= 10 * \text{rand}(n, n); \\ A &= R' * R + n * \text{eye}(n); \\ b &= (A - \text{eye}(n, n)) * \text{ones}(n, 1). \end{aligned} \quad (24)$$

The initial value is $x_0 = \text{rand}(5, 1)$, and the exact solution to this problem is $x_1 = x_2 = x_3 = x_4 = x_5 = 1$.

Example 5

$$\begin{aligned} n &= 1000; \\ B &= 0.5 * \text{ones}(n, n); \\ C &= (4 * n - 0.5) * \text{eye}(n); \\ D &= (n - 0.5) * \text{ones}(1, n - 1); \\ E &= \text{diag}(D, 1) + (\text{diag}(D, 1))'; \\ A &= B + C + E; \\ b &= (A - \text{eye}(n)) * \text{ones}(n, 1). \end{aligned} \quad (25)$$

Example 6

$$\begin{aligned} n &= 1000; \\ A &= -10 + \text{rand}(n, n) * 20; \\ x &= -2 + \text{rand}(n, 1) * 4; \\ b &= A * x - \text{abs}(x). \end{aligned} \quad (26)$$

For the Examples 1–4, the numerical solutions and errors ($Ax - |x| - b$) generated by the three algorithms in the first three iterations are given. Let $X_i, i = 1, 2, 3, \dots$ represent the numerical solution after the i -th iteration, and $R_i, i = 1, 2, 3, \dots$ represent the error generated by the i -th iteration.

In Tables 1–4, from the numerical solutions and errors obtained from the three algorithms in each iteration, it is clear that the Three-SIA method has a fast convergence speed and high solution accuracy, especially in Examples 1 and 3.

The convergence curves of the three algorithms for solving the high-dimensional system of absolute value

TABLE 2: The comparison of three-SIA, two-SIA, and GNM for Example 2.

Algorithm	X_1	X_2	X_3	$R_1; R_2; R_3$
GNM	$x_1 = 3.269694330310371$	$x_1 = 3.269694330310371$	$x_1 = 3.269694330310371$	$0.1588 \times 10^{-13};$
	$x_2 = 10.448630719160590$	$x_2 = 10.448630719160590$	$x_2 = 10.448630719160590$	$0.1588 \times 10^{-13};$
	$x_3 = 23.271556738061619$	$x_3 = 23.271556738061619$	$x_3 = 23.271556738061619$	0.1588×10^{-13}
Two-SIA	$x_1 = 3.269694330310371$	$x_1 = 3.269694330310371$	$x_1 = 3.269694330310371$	$0.3256 \times 10^{-13};$
	$x_2 = 10.448630719160588$	$x_2 = 10.448630719160588$	$x_2 = 10.448630719160588$	$0.3256 \times 10^{-13};$
	$x_3 = 23.271556738061616$	$x_3 = 23.271556738061616$	$x_3 = 23.271556738061616$	0.3256×10^{-13}
Three-SIA	$x_1 = 3.269694330310369$	$x_1 = 3.269694330310369$	$x_1 = 3.269694330310369$	$0.7105 \times 10^{-14};$
	$x_2 = 10.448630719160592$	$x_2 = 10.448630719160592$	$x_2 = 10.448630719160592$	$0.7105 \times 10^{-14};$
	$x_3 = 23.271556738061626$	$x_3 = 23.271556738061626$	$x_3 = 23.271556738061626$	0.7105×10^{-14}

TABLE 3: The comparison of three-SIA, two-SIA, and GNM for Example 3.

Algorithm	X_1	X_2	X_3	$R_1; R_2; R_3; R_4$
GNM	$x_1 = 1.0000000000000000$	$x_1 = 1.0000000000000000$	$x_1 = 1.0000000000000000$	$0.5329 \times 10^{-14};$
	$x_2 = 1.0000000000000000$	$x_2 = 1.0000000000000000$	$x_2 = 1.0000000000000000$	$0.5329 \times 10^{-14};$
	$x_3 = 1.0000000000000000$	$x_3 = 1.0000000000000000$	$x_3 = 1.0000000000000000$	$0.5329 \times 10^{-14};$
	$x_4 = 1.0000000000000000$	$x_4 = 1.0000000000000000$	$x_4 = 1.0000000000000000$	0.5329×10^{-14}
Two-SIA	$x_1 = 1.0000000000000000$	$x_1 = 1.0000000000000000$	$x_1 = 1.0000000000000000$	$0.6153 \times 10^{-14};$
	$x_2 = 1.0000000000000000$	$x_2 = 1.0000000000000000$	$x_2 = 1.0000000000000000$	$0.5329 \times 10^{-14};$
	$x_3 = 1.0000000000000000$	$x_3 = 1.0000000000000000$	$x_3 = 1.0000000000000000$	$0.4351 \times 10^{-14};$
	$x_4 = 1.0000000000000000$	$x_4 = 1.0000000000000000$	$x_4 = 1.0000000000000000$	0.1776×10^{-14}
Three-SIA	$x_1 = 1$	$x_1 = 1$	$x_1 = 1$	0;
	$x_2 = 1$	$x_2 = 1$	$x_2 = 1$	0;
	$x_3 = 1$	$x_3 = 1$	$x_3 = 1$	0;
	$x_4 = 1$	$x_4 = 1$	$x_4 = 1$	0

TABLE 4: The comparison of three-SIA, two-SIA, and GNM for Example 4.

Algorithm	X_1	X_2	X_3	$R_1; R_2; R_3; R_4; R_5$
GNM	$x_1 = 1.0000000000000002$	$x_1 = 1.0000000000000002$	$x_1 = 1.0000000000000002$	$0.6126 \times 10^{-11};$
	$x_2 = 1.0000000000000007$	$x_2 = 1.0000000000000007$	$x_2 = 1.0000000000000007$	$0.6126 \times 10^{-11};$
	$x_3 = 1.0000000000000004$	$x_3 = 1.0000000000000004$	$x_3 = 1.0000000000000004$	$0.6126 \times 10^{-11};$
	$x_4 = 0.9999999999999995$	$x_4 = 0.9999999999999995$	$x_4 = 0.9999999999999995$	$0.6126 \times 10^{-11};$
	$x_5 = 1.0000000000000021$	$x_5 = 1.0000000000000021$	$x_5 = 1.0000000000000021$	0.6126×10^{-11}
Two-SIA	$x_1 = 1.0000000000000001$	$x_1 = 1.0000000000000012$	$x_1 = 1.0000000000000001$	$0.4781 \times 10^{-11};$
	$x_2 = 1.0000000000000006$	$x_2 = 1.0000000000000006$	$x_2 = 1.0000000000000006$	$0.4510 \times 10^{-11};$
	$x_3 = 1.0000000000000003$	$x_3 = 1.0000000000000003$	$x_3 = 1.0000000000000003$	$0.4118 \times 10^{-11};$
	$x_4 = 0.9999999999999994$	$x_4 = 0.9999999999999994$	$x_4 = 0.9999999999999994$	$0.3136 \times 10^{-11};$
	$x_5 = 1.0000000000000021$	$x_5 = 1.0000000000000020$	$x_5 = 1.0000000000000020$	0.0873×10^{-11}
Three-SIA	$x_1 = 0.9999999999999998$	$x_1 = 1.0000000000000012$	$x_1 = 1.0000000000000001$	$0.1607 \times 10^{-12};$
	$x_2 = 1.0000000000000000$	$x_2 = 1.0000000000000000$	$x_2 = 1.0000000000000000$	$0.1607 \times 10^{-12};$
	$x_3 = 1.0000000000000003$	$x_3 = 1.0000000000000003$	$x_3 = 1.0000000000000003$	$0.1607 \times 10^{-12};$
	$x_4 = 1.0000000000000001$	$x_4 = 1.0000000000000001$	$x_4 = 1.0000000000000001$	$0.1607 \times 10^{-12};$
	$x_5 = 0.9999999999999997$	$x_5 = 0.9999999999999997$	$x_5 = 0.9999999999999997$	0.1607×10^{-12}

equations Examples 4 and 5 are shown in Figures 1 and 2. It can be seen from the figures that the Three-SIA method is the best in terms of convergence speed and accuracy. So the Algorithm 1 is superior in convergence and the quality of solution for solving AVEs (1). And this algorithm takes a

short time to calculate the low-dimensional absolute value equations, about 2 seconds. When calculating high-dimensional absolute value equations, such as 1000-dimensional equations, it takes about 10 seconds. Overall, the efficiency is higher than other algorithms.

FIGURE 1: The convergence curve of three-SIA, two-SIA, and GNM for Example 5.

FIGURE 2: The convergence curve of three-SIA, two-SIA, and GNM for Example 6.

5. Conclusions

In this paper, we proposed a three-step iterative algorithm for solving nondifferentiable and NP-hard AVEs (1), when the minimum singular value of A is greater than 1. Compared Two-SIA and GNM, our Three-SIA method has faster convergence speed fast and the solution accuracy is high. Next, we will consider solving the system of absolute value equations with smooth high-order algorithm.

Data Availability

Data from previous studies were used to support this study and are cited at relevant places within the text as references.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The research was supported by the National Natural Science Foundation of China (Grant no. 61877046), the Second Batch of Young Outstanding Talents Support Plan of Shaanxi Universities.

References

- [1] O. L. Mangasarian, "A generalized Newton method for absolute value equations," *Optimization Letters*, vol. 3, no. 1, pp. 101–108, 2009.
- [2] O. L. Mangasarian, "Knapsack feasibility as an absolute value equation solvable by successive linear programming," *Optimization Letters*, vol. 3, no. 2, pp. 161–170, 2009.
- [3] K. Moizumi and G. Cybenko, "The traveling agent problem," *Mathematics of Control, Signals, and Systems*, vol. 14, no. 3, pp. 213–232, 2001.
- [4] O. L. Mangasarian and R. R. Meyer, "Absolute value equations," *Linear Algebra and Its Applications*, vol. 419, no. 2-3, pp. 359–367, 2006.
- [5] J.-M. Feng and S.-Y. Liu, "An improved generalized Newton method for absolute value equations," *SpringerPlus*, vol. 5, pp. 1–10, 2016.
- [6] J.-M. Feng and S.-Y. Liu, "A new two-step iterative method for solving absolute value equations," *Journal of Inequalities and Applications*, vol. 2019, p. 1, 2019.
- [7] C.-X. LI, "A preconditioned AOR iterative method for the absolute value equations," *International Journal of Computational Methods*, vol. 14, no. 2, Article ID 1750016, 2016.
- [8] D. K. Salkuyeh, "The Picard-HSS iteration method for absolute value equations," *Optimization Letters*, vol. 8, no. 8, pp. 2191–2202, 2014.
- [9] X.-M. Gu, T.-Z. Huang, H.-B. Li, S.-F. Wang, and L. Li, "Two CSCS-based iteration methods for solving absolute value equations," *Journal of Applied Analysis and Computation*, vol. 7, no. 4, pp. 1336–1356, 2017.
- [10] F. K. Haghani, "On generalized traub's method for absolute value equations," *Journal of Optimization Theory and Applications*, vol. 166, no. 2, pp. 619–625, 2015.
- [11] J. Rohn, V. Hooshyarbakhsh, and R. Farhadsefat, "An iterative method for solving absolute value equations and sufficient conditions for unique solvability," *Optimization Letters*, vol. 8, no. 1, pp. 35–44, 2014.
- [12] M. A. Noor, J. Iqbal, K. I. Noor, and E. Al-Said, "On an iterative method for solving absolute value equations," *Optimization Letters*, vol. 6, no. 5, pp. 1027–1033, 2012.
- [13] S. Singh and D. K. Gupta, "Iterative methods of higher order for nonlinear equations," *Vietnam Journal of Mathematics*, vol. 44, no. 2, pp. 387–398, 2016.
- [14] A. Cordero, E. Martínez, and J. R. Torregrosa, "Iterative methods of order four and five for systems of nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 231, no. 2, pp. 541–551, 2009.
- [15] J.-J. Yu, P. Jiang, and Z. Liu, "The two kind of iterative methods with fifth-order for solving of nonlinear equations," *Mathematica Numerica Sinica*, vol. 39, no. 2, pp. 151–166, 2017, in Chinese.
- [16] L. Qi and J. Sun, "A nonsmooth version of Newton's method," *Mathematical Programming*, vol. 58, no. 1–3, pp. 353–367, 1993.