Applied Mathematics and Computation 311 (2017) 195–202

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

# SOR-like iteration method for solving absolute value equations☆

Yi-Fen Ke, Chang-Feng Ma*

*School of Mathematics and Computer Science & FJKLMAA, Fujian Normal University, Fuzhou 350117, PR China*

ARTICLE INFO

*MSC:*
65F10
65H10

*Keywords:*
Absolute value equations
SOR-like method
Matrix splitting
Convergence analysis

ABSTRACT

In this paper, we propose an SOR-like iteration method for solving the absolute value equation (AVE), which is obtained by reformulating equivalently the AVE as a two-by-two block nonlinear equation. The convergence results of the proposed iteration method are proved under certain assumptions imposed on the involved parameter. Numerical experiments are given to demonstrate the feasibility, robustness and effectiveness of the SOR-like iteration method.



## 1. Introduction

Consider the absolute value equation (AVE):

$$Ax - |x| = b, \tag{1.1}$$

where $A = (a_{ij}) \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$ and $|x|$ denotes the vector in $\mathbb{R}^n$ with absolute values of components of the vector $x$. The system (1.1) is a special case of the generalized absolute value equation of the following form

$$Ax + B|x| = b, \tag{1.2}$$

where $B \in \mathbb{R}^{n\times n}$, was introduced by Rohn [27] and further studied in [4,12,14–16,18,21,23–26,30].

The AVE (1.1) arises in many areas of scientific computing and engineering applications. For example, linear programs, quadratic programs and bimatrix games can be reduced to a linear complementarity problem (LCP) [7,25]. Based on the modulus method, LCP can be formulated as a system of absolute value equations such as (1.1) [21]. Specially, many modulus-based matrix splitting iteration methods are proposed for solving the solutions of LCP, see for example [2,5,13,31,34,35] and references therein.

In recent years, the problem of solving the AVE has attracted much attention and has been investigated in the literature, see for example [3,6,7,9–12,14–23,25–30,32,33,36] and references therein. A large variety of methods for solving the AVE in (1.1) have been developed. Most of those methods are based on the Newton algorithm as the AVE in (1.1) being a weakly nonlinear equation. For example, in [1], Mangasarian proposed a generalized Newton method for solving the AVE, which generates a sequence formally stated as

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

☆ Supported by Fujian Natural Science Foundation (grant no. 2016J01005) and Strategic Priority Research Program of the Chinese Academy of Sciences (grant no. XDB18010202).

* Corresponding author. Tel. +8613763827962.
*E-mail address:* macf@fjnu.edu.cn (C.-F. Ma).

http://dx.doi.org/10.1016/j.amc.2017.05.035
0096-3003/

where $D(x) := \mathrm{diag}(\mathrm{sgn}(x))$, $x \in \mathbb{R}^n$. In [25], Noor et al. proposed an iteration method for solving the AVE in (1.1) with $A$ being a symmetric matrix.

In this paper, by reformulating the AVE in (1.1) as a two-by-two block nonlinear equation, we propose an SOR-like iteration method for solving it, which is based on a splitting of the two-by-two block coefficient matrix. We prove that the proposed iteration method will converge to the solution of the AVE in (1.1) under suitable choices of the involved parameter. In addition, we also use numerical examples to show that the SOR-like iteration method is feasible and effective in computing.

This paper is organized as follows. In Section 2, we present some notations and preliminaries that will be used throughout the paper. In Section 3, we propose an SOR-like iteration method for solving the AVE in (1.1) and consider the convergence of the proposed iteration method. Experimental results and conclusions are given in Sections 4 and 5, respectively.

## 2. Preliminaries

In this section, we present some notations and auxiliary results.

Let $\mathbb{R}^{n\times n}$ be the set of all $n \times n$ matrices with real entries and $\mathbb{R}^n = \mathbb{R}^{n\times 1}$. The $i$th component of a vector $x \in \mathbb{R}^n$ is denoted by $x_i$ for every $i = 1, \ldots, n$. For $x \in \mathbb{R}^n$, $\mathrm{sgn}(x)$ denotes a vector with components equal to $-1$, 0 or 1 depending on whether the corresponding component of the vector $x$ is negative, zero or positive. Denote $|x|$ the vector with $i$th component equal to $|x_i|$.

The symbol $I$ denotes the $n \times n$ identity matrix. If $x \in \mathbb{R}^n$, then $\mathrm{diag}(x)$ will denote an $n \times n$ diagonal matrix with $(i, i)$th entry equal to $x_i$, $i = 1, \ldots, n$. Let $A = (a_{ij})$ and $B = (b_{ij})$ be two real $m \times n$ matrices, $A \geq B (A > B)$ if $a_{ij} \geq b_{ij} (a_{ij} > b_{ij})$ holds for all $1 \leq i \leq m$ and $1 \leq j \leq n$. A matrix $A \in \mathbb{R}^{m\times n}$ is said to be nonnegative (positive) if its entries satisfy $a_{ij} \geq 0$ $(a_{ij} > 0)$ for all $1 \leq i \leq m$ and $1 \leq j \leq n$. For the matrix $A \in \mathbb{R}^{n\times n}$, $\|A\|$ denotes the spectral norm defined by $\|A\| := \max\{\|Ax\| : x \in \mathbb{R}^n, \ \|x\| = 1\}$, where $\|x\|$ is the 2-norm.

The following proposition was proved in Proposition 4 of [21].

**Proposition 2.1** *[21]. Assume that $A \in \mathbb{R}^{n\times n}$ is invertible. If $\|A^{-1}\| < 1$, then the AVE in (1.1) has a unique solution for any $b \in \mathbb{R}^n$.*

About the nonnegative matrix, we have the following results.

**Lemma 2.1** *[3]. For any vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$, the following results hold:*

(1) $\| |x| - |y| \| \leq \|x - y\|$;
(2) *if* $0 \leq x \leq y$, *then* $\|x\| \leq \|y\|$;
(3) *if* $x \leq y$ *and* $P$ *is a nonnegative matrix, then* $Px \leq Py$.

**Lemma 2.2** *[3]. For any matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times n}$, if $0 \leq A \leq B$, then $\|A\| \leq \|B\|$.*

## 3. SOR-like iteration method

Let $y = |x|$, then the AVE in (1.1) is equivalent to

$$\begin{cases} Ax - y = b, \\ -|x| + y = 0, \end{cases}$$

that is

$$\mathcal{A}\mathbf{z} := \begin{pmatrix} A & -I \\ -D(x) & I \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix} := \mathbf{b}, \tag{3.1}$$

where $D(x) := \mathrm{diag}(\mathrm{sgn}(x))$, $x \in \mathbb{R}^n$.

Note that the Eq. (3.1) is also nonlinear, as the matrix $D(x)$ depends on the variable $x$. It is also quite complicated to solve (3.1) in actual computations.

Let

$$\mathcal{A} = \mathcal{D} - \mathcal{L} - \mathcal{U},$$

where

$$\mathcal{D} = \begin{pmatrix} A & 0 \\ 0 & I \end{pmatrix}, \quad \mathcal{L} = \begin{pmatrix} 0 & 0 \\ D(x) & 0 \end{pmatrix}, \quad \mathcal{U} = \begin{pmatrix} 0 & I \\ 0 & 0 \end{pmatrix},$$

then we can obtain the following fixed point equation

$$(\mathcal{D} - \omega\mathcal{L})\mathbf{z} = [(1-\omega)\mathcal{D} + \omega\mathcal{U}]\mathbf{z} + \omega\mathbf{b},$$

where the parameter $\omega > 0$. That is

$$\begin{pmatrix} A & 0 \\ -\omega D(x) & I \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} (1-\omega)A & \omega I \\ 0 & (1-\omega)I \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \omega b \\ 0 \end{pmatrix}. \tag{3.2}$$

Based on the fixed point equation (3.2), we can establish the following matrix splitting iteration method for solving the AVE in (1.1), called the SOR-like iteration method.

**Method 3.1.** (The SOR-like iteration method for the AVE in (1.1))

Let $A \in \mathbb{R}^{n\times n}$ be a nonsingular matrix and $b \in \mathbb{R}^n$. Given the initial vectors $x^{(0)} \in \mathbb{R}^n$ and $y^{(0)} \in \mathbb{R}^n$, for $k = 0, 1, 2, \ldots$ until the iteration sequence $\{(x^{(k)}, y^{(k)})\}_{k=0}^{+\infty}$ is convergent, compute

$$\begin{cases} x^{(k+1)} = (1-\omega)x^{(k)} + \omega A^{-1}(y^{(k)} + b), \\ y^{(k+1)} = (1-\omega)y^{(k)} + \omega|x^{(k+1)}|. \end{cases} \tag{3.3}$$

Here $\omega$ is a positive constant.

Let $(x^*, y^*)$ be the solution pair of the nonlinear eq (3.1) and $(x^{(k)}, y^{(k)})$ be generated by the iteration method (3.3). Define the iteration errors

$$e_k^x = x^* - x^{(k)}, \quad e_k^y = y^* - y^{(k)}.$$

Now, we are ready to prove the main result of this paper.

**Theorem 3.1.** *Let $A \in \mathbb{R}^{n\times n}$ be a nonsingular matrix and $b \in \mathbb{R}^n$. Denote*

$$\nu = \|A^{-1}\|, \quad \alpha = |1-\omega| \quad \text{and} \quad \beta = \omega^2\nu.$$

*If*

$$0 < \omega < 2 \quad \text{and} \quad \alpha^4 - 3\alpha^2 - 2\beta\alpha - 2\beta^2 + 1 > 0, \tag{3.4}$$

*then the following inequality*

$$|||(e_{k+1}^x, e_{k+1}^y)||| < |||(e_k^x, e_k^y)||| \tag{3.5}$$

*holds for $k = 0, 1, 2, \ldots$. Here, the norm is defined by*

$$|||(e^x, e^y)||| := \sqrt{\|e^x\|^2 + \omega^{-2}\|e^y\|^2}.$$

**Proof.** From (3.2) and (3.3), we have

$$e_{k+1}^x = (1-\omega)e_k^x + \omega A^{-1}e_k^y, \tag{3.6}$$

$$e_{k+1}^y = (1-\omega)e_k^y + \omega(|x^*| - |x^{(k+1)}|). \tag{3.7}$$

From (3.6), we can obtain

$$\|e_{k+1}^x\| \le |1-\omega| \cdot \|e_k^x\| + \omega\nu\, \|e_k^y\|. \tag{3.8}$$

According to (3.7) and Lemma 2.1, we have

$$\begin{aligned} \|e_{k+1}^y\| &\le |1-\omega| \cdot \|e_k^y\| + \omega\| \, |x^*| - |x^{(k+1)}| \, \| \\ &\le |1-\omega| \cdot \|e_k^y\| + \omega\|x^* - x^{(k+1)}\| \\ &= |1-\omega| \cdot \|e_k^y\| + \omega\|e_{k+1}^x\|. \end{aligned} \tag{3.9}$$

Thus, from (3.8) and (3.9), we get

$$\begin{pmatrix} -\omega & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \|e_{k+1}^x\| \\ \|e_{k+1}^y\| \end{pmatrix} \le \begin{pmatrix} 0 & |1-\omega| \\ |1-\omega| & \omega\nu \end{pmatrix} \begin{pmatrix} \|e_k^x\| \\ \|e_k^y\| \end{pmatrix}. \tag{3.10}$$

Let

$$P = \begin{pmatrix} 0 & 1 \\ 1 & w \end{pmatrix} \ge 0.$$

Multiplying (3.10) from left by the nonnegative matrix $P$ and according to Lemma 2.1, we have

$$\begin{pmatrix} \|e_{k+1}^x\| \\ \|e_{k+1}^y\| \end{pmatrix} \le \begin{pmatrix} |1-\omega| & \omega\nu \\ \omega|1-\omega| & |1-\omega| + \omega^2\nu \end{pmatrix} \begin{pmatrix} \|e_k^x\| \\ \|e_k^y\| \end{pmatrix}. \tag{3.11}$$

Denote

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & \omega^{-1} \end{pmatrix} \ge 0.$$

Multiplying (3.11) from left by the nonnegative matrix $Q$, we get

$$\begin{pmatrix} \|e^x_{k+1}\| \\ \omega^{-1}\|e^y_{k+1}\| \end{pmatrix} \leq \begin{pmatrix} |1-\omega| & \omega^2\nu \\ |1-\omega| & |1-\omega|+\omega^2\nu \end{pmatrix} \begin{pmatrix} \|e^x_k\| \\ \omega^{-1}\|e^y_k\| \end{pmatrix}.$$

Hence, we have

$$|||(e^x_{k+1}, e^y_{k+1})||| \leq \|T(\omega)\| \cdot |||(e^x_k, e^y_k)|||,$$

where

$$T(\omega) := \begin{pmatrix} |1-\omega| & \omega^2\nu \\ |1-\omega| & |1-\omega|+\omega^2\nu \end{pmatrix}.$$

Now, we consider the choice of the iteration parameter $\omega$ such that $\|T(\omega)\| < 1$, thus the inequality (3.5) holds. Denote

$$T(\omega) := \begin{pmatrix} \alpha & \beta \\ \alpha & \alpha+\beta \end{pmatrix}.$$

Since

$$T(\omega)^\top T(\omega) = \begin{pmatrix} 2\alpha^2 & \alpha^2+2\alpha\beta \\ \alpha^2+2\alpha\beta & \alpha^2+2\beta^2+2\alpha\beta \end{pmatrix},$$

then we have

$$\mathrm{tr}(T(\omega)^\top T(\omega)) = 3\alpha^2+2\beta^2+2\alpha\beta \tag{3.12}$$

and

$$\det(T(\omega)^\top T(\omega)) = \alpha^4. \tag{3.13}$$

Assume $\lambda$ is an eigenvalue of the matrix $T(\omega)^\top T(\omega)$ with $\lambda \geq 0$, thus $\lambda$ will satisfy

$$\lambda^2 - \mathrm{tr}(T(\omega)^\top T(\omega))\lambda + \det(T(\omega)^\top T(\omega)) = 0.$$

Hence, we have the following relations

$$\lambda_1+\lambda_2 = \mathrm{tr}(T(\omega)^\top T(\omega)), \quad \lambda_1\lambda_2 = \det(T(\omega)^\top T(\omega)). \tag{3.14}$$

If

$$0 < \omega < 2 \quad \text{and} \quad \alpha^4 - 3\alpha^2 - 2\beta\alpha - 2\beta^2 + 1 > 0,$$

from (3.12)–(3.14), we have

$$0 \leq \lambda_1\lambda_2 < 1 \quad \text{and} \quad \lambda_1+\lambda_2 < 1+\lambda_1\lambda_2,$$

that is

$$0 \leq \lambda_1\lambda_2 < 1 \quad \text{and} \quad (\lambda_1-1)(\lambda_2-1) > 0.$$

Then, we can get

$$0 \leq \lambda_1 < 1 \quad \text{and} \quad 0 \leq \lambda_2 < 1.$$

Hence $\|T(\omega)\| < 1$. This completes the proof. □

It is easy to see that if the conditions of Theorem 3.1 are satisfied, then we have

$$0 \leq |||(e^x_k, e^y_k)||| \leq \|T(\omega)\| \cdot |||(e^x_{k-1}, e^y_{k-1})||| \leq \cdots \leq \|T(\omega)\|^k \cdot |||(e^x_0, e^y_0)|||.$$

Since $\|T(\omega)\| < 1$, we have $\lim_{k\to\infty} |||(e^x_k, e^y_k)||| = 0$. Using the definition of the norm $|||\cdot|||$ gives $\lim_{k\to\infty}\|e^x_k\| = 0$ and $\lim_{k\to\infty}\|e^y_k\| = 0$. Therefore, the sequence $\{x^{(k)}\}$ generated by (3.3) will converge to the solution of the AVE in (1.1).

**Corollary 3.1.** *Let $A \in \mathbb{R}^{n\times n}$ be a nonsingular matrix and $b \in \mathbb{R}^n$. Denote $\nu = \|A^{-1}\|$. If*

$$\nu < 1 \quad \text{and} \quad 1-\tau < \omega < \min\left\{1+\tau, \sqrt{\frac{\tau}{\nu}}\right\}, \tag{3.15}$$

*where $\tau = \dfrac{2}{3+\sqrt{5}}$. Then the following inequality*

$$|||(e^x_{k+1}, e^y_{k+1})||| < |||(e^x_k, e^y_k)||| \tag{3.16}$$

*holds for $k = 0, 1, 2, \ldots$.*

**Proof.** From the proof of Theorem 3.1, we have

$$|||(e^x_{k+1}, e^y_{k+1})||| \le \|T(\omega)\| \cdot |||(e^x_k, e^y_k)|||,$$

where

$$T(\omega) := \begin{pmatrix} |1-\omega| & \omega^2\nu \\ |1-\omega| & |1-\omega|+\omega^2\nu \end{pmatrix}.$$

Now, we also consider the choice of the iteration parameter $\omega$ such that $\|T(\omega)\| < 1$, thus the inequality (3.16) holds. Denote

$$\eta = \max\left\{|1-\omega|,\ \omega^2\nu\right\}.$$

Then it holds

$$0 \le T(\omega) \le \begin{pmatrix} \eta & \eta \\ \eta & 2\eta \end{pmatrix} = \eta \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} := \eta S.$$

According to Lemma 2.2, we have

$$\|T(\omega)\| \le \|\eta S\| = \eta\|S\| = \eta \frac{3+\sqrt{5}}{2}.$$

Let $\tau = \dfrac{2}{3+\sqrt{5}}$. Hence, if $\eta < \tau$, then it will hold that $\|T(\omega)\| < 1$. Note that

$$\eta < \tau \Leftrightarrow \begin{cases} |1-\omega| < \tau \\ \omega^2\nu < \tau \end{cases} \Leftrightarrow \begin{cases} 1-\tau < \omega < 1+\tau \\ \omega < \sqrt{\dfrac{\tau}{\nu}} \end{cases}$$

$$\Leftarrow \begin{cases} 1-\tau < \omega < \min\left\{1+\tau, \sqrt{\dfrac{\tau}{\nu}}\right\} \\ 1-\tau < \sqrt{\dfrac{\tau}{\nu}} \end{cases}$$

$$\Leftrightarrow \begin{cases} 1-\tau < \omega < \min\left\{1+\tau, \sqrt{\dfrac{\tau}{\nu}}\right\} \\ \nu < \dfrac{\tau}{(1-\tau)^2} = 1 \end{cases}.$$

This completes the proof. □

If the conditions (3.15) hold, according to Proposition 2.1, we can get that the sequence $\{x^{(k)}\}$ will converge to the unique solution of the AVE in (1.1).

## 4. Numerical experiments

In this section, we use some test problems to examine the effectiveness of the SOR-like iteration method (3.3). All test problems are started from the initial zero vector, are terminated if the current iterations satisfy

$$\text{RES} := \|b + |x^{(k)}| - Ax^{(k)}\| \le 10^{-6}$$

or if the number of the prescribed iteration steps $k_{\max} = 1000$ is exceeded, and are performed under Matlab R2011b on a personal computer with 3.30 GHz central processing unit (Intel(R) Core(TM) i5-4590), 4 GB memory and Windows 7 operating system. In addition, ‘IT’ denotes the number of iteration steps, ‘CPU’ denotes the elapsed CPU time in seconds, and

$$\text{ERR} := \|x^{(k)} - x^*\|,$$

where $x^*$ is the exact solution.

**Example 1.** Consider the AVE in (1.1) with

$$A = \text{tridiag}(-1, 4, -1) \in \mathbb{R}^{n\times n}, \quad x^* = (-1, 1, -1, 1, \dots, -1, 1)^\top \in \mathbb{R}^n,$$

and $b = Ax^* - |x^*|$.

For Example 1, since the matrix $A$ is symmetric positive definite and $C = A - D(x)$ is also symmetric positive definite for any vector $x$, then the generalized Newton method proposed in [1] (denoted by GN) and the iteration method proposed in [25] (denoted by NINA) can be implemented. Thus, we compare the new iteration method with the generalized Newton method and the NINA iteration method. For SOR-like iteration method, the parameter $\omega = 1$.

**Table 1**
Numerical results for Example 1.

| | $n$ | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 |
|---|---|---|---|---|---|---|---|
| | IT | 2 | 2 | 2 | 2 | 2 | 2 |
| GN | CPU | 0.0092 | 0.0459 | 0.0833 | 0.1590 | 0.2322 | 0.3203 |
| | RES | 1.9741e−14 | 2.8002e−14 | 3.4330e−14 | 3.9661e−14 | 4.4356e−14 | 4.8599e−14 |
| | ERR | 4.9364e−15 | 7.0015e−15 | 8.5833e−15 | 9.9159e−15 | 1.1089e−14 | 1.2150e−14 |
| | IT | 18 | 18 | 18 | 18 | 18 | 18 |
| NINA | CPU | 0.3073 | 0.9824 | 2.3326 | 3.9088 | 5.7240 | 7.7787 |
| | RES | 3.4947e−07 | 4.9580e−07 | 6.0786e−07 | 7.0227e−07 | 7.8541e−07 | 8.6055e−07 |
| | ERR | 1.8652e−07 | 2.6468e−07 | 3.2453e−07 | 3.7494e−07 | 4.1934e−07 | 4.5946e−07 |
| | IT | 15 | 16 | 16 | 16 | 16 | 17 |
| SOR-like | CPU | 0.0013 | 0.0020 | 0.0035 | 0.0053 | 0.0065 | 0.0078 |
| | RES | 9.8477e−07 | 6.3197e−07 | 7.7428e−07 | 8.9422e−07 | 9.9988e−07 | 2.0134e−07 |
| | ERR | 4.4031e−07 | 1.0390e−07 | 1.2729e−07 | 1.4701e−07 | 1.6438e−07 | 9.0038e−08 |

**Table 2**
Test problems of Example 2.

| Problem | $n$ | Problem | $n$ |
|---|---|---|---|
| $mesh1e1$ | 48 | $Trefethen\_20b$ | 19 |
| $mesh1em1$ | 48 | $Trefethen\_200b$ | 199 |
| $mesh2e1$ | 306 | $Trefethen\_20000b$ | 19999 |

**Table 3**
Numerical results for Example 2.

| Problem | $\omega$ | IT | CPU | RES | ERR |
|---|---|---|---|---|---|
| $mesh1e1$ | 0.94 | 18 | 0.0015 | 7.8970e−07 | 4.6819e−07 |
| $mesh1em1$ | 0.93 | 18 | 0.0016 | 8.2020e−07 | 4.5596e−07 |
| $mesh2e1$ | 0.94 | 20 | 0.0100 | 9.2042e−07 | 6.7690e−07 |
| $Trefethen\_20b$ | 0.95 | 12 | 0.0007 | 6.6360e−07 | 2.0192e−07 |
| $Trefethen\_200b$ | 0.95 | 12 | 0.0085 | 6.7128e−07 | 2.0587e−07 |
| $Trefethen\_20000b$ | 0.95 | 12 | 166.3732 | 6.7270e−07 | 2.0617e−07 |

The computing results for Example 1 are listed in Table 1. From Table 1, we can see that these iteration methods converge fast to the solution $x^*$ for different dimensions $n$, respectively. However, the elapsed CPU time of the SOR-like iteration method is less than the generalized Newton method and the NINA iteration method.

**Example 2.** Consider the AVE in (1.1), where the matrix $A \in \mathbb{R}^{n\times n}$ comes from six different test problems listed in Table 2. These test matrices $A$ are sparse, symmetry and $\|A^{-1}\| < 1$. Let

$$x^* = (-1, 1, -1, 1, \ldots, -1, 1)^\top \in \mathbb{R}^n \quad \text{and} \quad b = Ax^* - |x^*|.$$

For more about these test problems, it can be seen in [8].

For Example 2, we consider the effect of different parameter $\omega$ for the SOR-like iteration method. In Fig. 1, we present the iteration steps for the SOR iteration method under different parameters $\omega$ for different test problems, respectively. And those parameters satisfy the conditions of Theorem 3.1.

In particular, the computing results under the optimal parameters for Example 2 are listed in Table 3. From Table 3, we can see that the SOR-like iteration method converges fast to the solution $x^*$ for different test problems under the optimal parameters.

## 5. Concluding remarks

We have presented an SOR-like iteration method for solving the AVE in (1.1), which is obtained by reformulating equivalently the AVE as a two-by-two block nonlinear equation. We have proved that the proposed iteration method converges to the solution of the AVE in (1.1) under suitable choices of the involved parameter. Numerical examples have shown that the proposed iteration method is feasible and effective in computing.

However, it is still worth considering to extend the scope of the involved parameter $\omega$ to make sure the convergence of the proposed iteration method. In addition, the choice of the optimal iteration parameter in theory also merits some consideration.

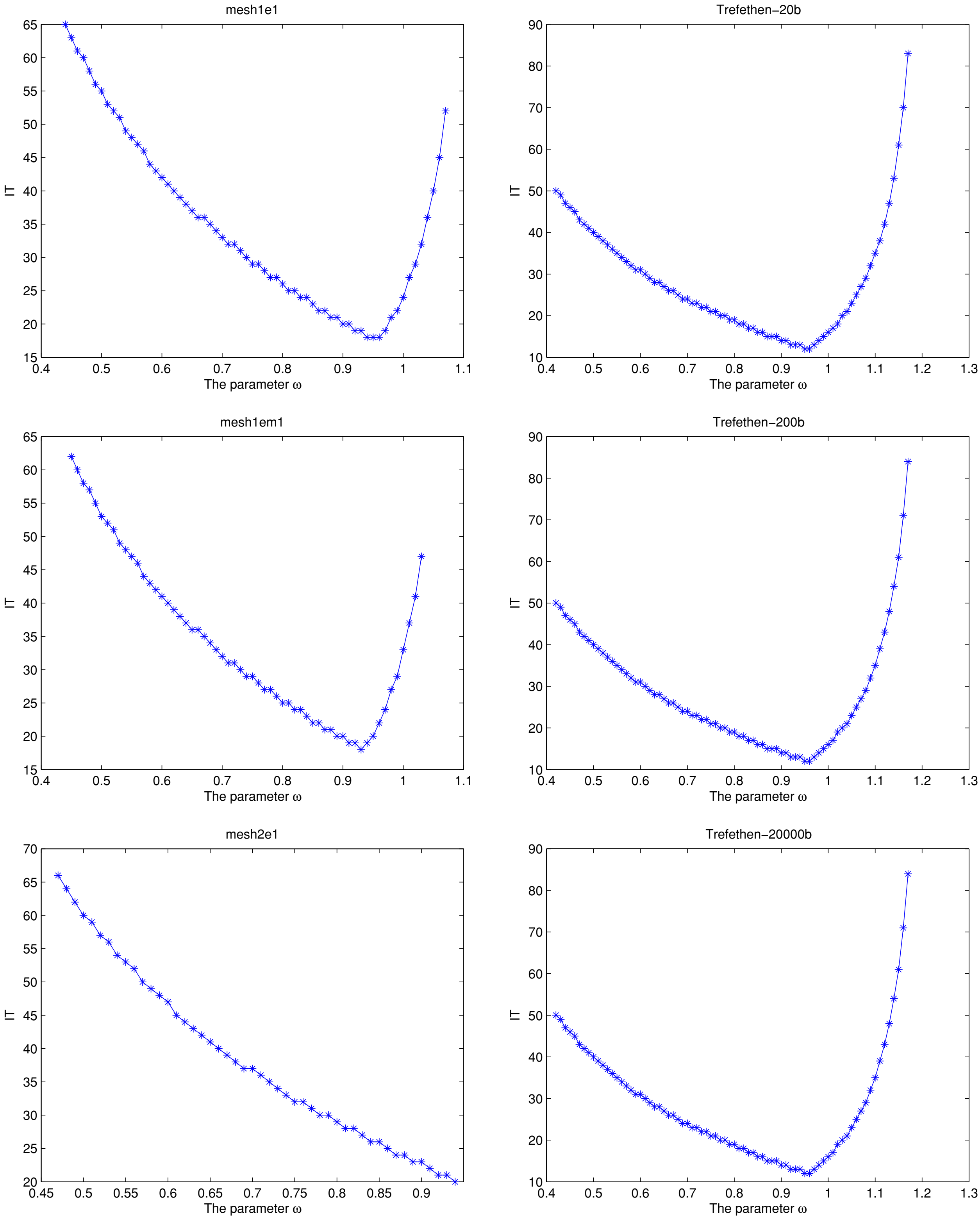


**Fig. 1.** The IT of Example 2 for the SOR-like iteration method.

## Acknowledgments

The authors would like to express their great thankfulness to the referees for the comments and constructive suggestions very much, which are valuable in improving the quality of the original paper.

## References

[1] O.L. Mangasarian, A generalized Newton method for absolute value equations, Optim. Lett. 3 (2009) 101–108.
[2] Z.-Z. Bai, Modulus-based matrix splitting iteration methods for linear complementarity problems, Numer. Linear Algebra Appl. 17 (2010) 917–933.
[3] A. Berman, R. Plemmons, Nonnegative Matrices in the Mathematical Sciences, Academic Press, New York, 1979.
[4] Z.-Z. Bai, X. Yang, On HSS-based iteration methods for weakly nonlinear systems, Appl. Numer. Math. 59 (2009) 2923–2936.
[5] Z.-Z. Bai, L.-L. Zhang, Modulus-based synchronous multisplitting iteration methods for linear complementarity problems, Numer. Linear Algebra Appl. 20 (2013) 425–439.
[6] J.Y.B. Cruz, O.P. Ferreira, L.F. Prudente, On the global convergence of the inexact semi-smooth Newton method for absolute value equation, Comput. Optim. Appl. 65 (2016) 93–108.
[7] L. Caccetta, B. Qu, G.-L. Zhou, A globally and quadratically convergent method for absolute value equations, Comput. Optim. Appl. 48 (2011) 45–58.
[8] T. Davis, University of Florida Sparse Matrix Collection, University of Florida, Gainesville, FL, http://www.cise.ufl.edu/research/sparse/matrices/.
[9] V. Edalatpour, D. Hezari, D. Khojasteh Salkuyeh, A generalization of the Gauss–Seidel iteration method for solving absolute value equations, Appl. Math. Comput. 293 (2017) 156–167.
[10] S.-L. Hu, Z.-H. Huang, A note on absolute value equations, Optim. Lett. 4 (2010) 417–424.
[11] S.-L. Hu, Z.-H. Huang, Q. Zhang, A generalized Newton method for absolute value equations associated with second order cones, J. Comput. Appl. Math. 235 (2011) 1490–1501.
[12] J. Iqbal, M. Arif, Symmetric SOR method for absolute complementarity problems, J. Appl. Math. (2013), doi:10.1155/2013/172060.
[13] Y.-F. Ke, C.-F. Ma, On the convergence analysis of two-step modulus-based matrix splitting iteration method for linear complementarity problems, Appl. Math. Comput. 243 (2014) 413–418.
[14] S. Ketabchi, H. Moosaei, An efficient method for optimal correcting of absolute value equations by minimal changes in the right hand side, Comput. Math. Appl. 64 (2012) 1882–1885.
[15] S. Ketabchi, H. Moosaei, Minimum norm solution to the absolute value equation in the convex case, J. Optim. Theory Appl. 154 (2012b) 1080–1087.
[16] S. Ketabchi, H. Moosaei, S. Fallhi, Optimal error correction of the absolute value equations using a genetic algorithm, Comput. Math. Model. 57 (2013) 2339–2342.
[17] C.-X. Li, A modified generalized Newton method for absolute value equations, J. Optim. Theory Appl. 170 (2016) 1055–1059.
[18] O.L. Mangasarian, Absolute value equation solution via concave minimization, Optim. Lett. 1 (2007) 3–8.
[19] O.L. Mangasarian, A hybrid algorithm for solving the absolute value equation, Optim. Lett. 9 (2015) 1469–1474.
[20] H. Moosaei, S. Ketabchi, M.A. Noor, J. Iqbal, V. Hooshyarbakhsh, Some techniques for solving absolute value equations, Appl. Math. Comput. 268 (2015) 696–705.
[21] O.L. Mangasarian, R.R. Meyer, Absolute value equations, Linear Algebra Appl. 419 (2006) 359–367.
[22] X.-H. Miao, J.-T. Yang, S.-L. Hu, A generalized Newton method for absolute value equations associated with circular cones, Appl. Math. Comput. 269 (2015) 155–168.
[23] M.A. Noor, J. Iqbal, E. Al-Said, Residual iterative method for solving absolute value equations, Abstr. Appl. Anal. (2012), doi:10.1155/2012/406232.
[24] M.A. Noor, J. Iqbal, S. Khattri, E. Al-Said, A new iterative method for solving absolute value equations, Int. J. Phys. Sci. 6 (2011) 1793–1797.
[25] M.A. Noor, J. Iqbal, K.I. Noor, E. Al-Said, On an iterative method for solving absolute value equations, Optim. Lett. 6 (2012) 1027–1033.
[26] M.A. Noor, J. Iqbal, K.I. Noor, E. Al-Said, Generalized AOR method for solving absolute complementarity problems, J. Appl. Math. (2012), doi:10.1155/2012/743861.
[27] J. Rohn, A theorem of the alternatives for the equation $ax + b|x| = b$, Linear Multilinear Algebra 52 (2004) 421–426.
[28] D.K. Salkuyeh, The Picard-HSS iteration method for absolute value equations, Optim. Lett. 8 (2014) 2191–2202.
[29] S.-L. Wu, P. Guo, On the unique solvability of the absolute value equation, J. Optim. Theory Appl. 169 (2016) 705–712.
[30] H. Wang, H. Liu, S. Cao, A verification method for enclosing solutions of absolute value equations, Collect. Math. 64 (2013) 17–18.
[31] L.-L. Zhang, Two-step modulus based matrix splitting iteration for linear complementarity problems, Numer. Algorithms 57 (2011) 83–99.
[32] J.-J. Zhang, The relaxed nonlinear PHSS-like iteration method for absolute value equations, Appl. Math. Comput. 265 (2015) 266–274.
[33] M. Zhang, Z.-H. Huang, Y.-F. Li, The sparsest solution to the system of absolute value equations, J. Oper. Res. Soc. China 3 (2015) 31–51.
[34] N. Zheng, J.-F. Yin, Accelerated modulus-based matrix splitting iteration methods for linear complementarity problem, Numer. Algorithms 64 (2013) 245–262.
[35] N. Zheng, J.-F. Yin, Convergence of accelerated modulus-based matrix splitting iteration methods for linear complementarity problem with an $h_+$-matrix, J. Comput. Appl. Math. 260 (2014) 281–293.
[36] C. Zhang, Q.-J. Wei, Global and finite convergence of a generalized Newton method for absolute value equations, J. Optim. Theory Appl. 143 (2009) 391–403.