# SOR-**Like Method for a New Generalized Absolute Value Equation**

## Shuan Yang[1*] and Shi-Liang Wu[1**]

[1] *School of Mathematics, Yunnan Normal University, Kunming, 650500 PR China*
Received June 8, 2022; in final form, July 2, 2022; accepted August 20, 2022

**Abstract**—In this paper, we extend the SOR-like iteration method for a new generalized absolute value equation and obtain its convergence properties. What is more, the optimal parameter of the SOR-like iteration is obtained. The result of numerical experiments shows that the proposed method is reliable and feasible.

## 1. INTRODUCTION

In this paper, we focus on the numerical solution of the new generalized absolute value equation (NGAVE)

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

where $A, B \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, and $|x|$ denotes the vector that consists of the absolute values of components of the vector $x$. Obviously, it is not difficult to find when $B = I$ and the NGAVE (1.1) reduces to the standard absolute value equation (AVE), see [1]:

$$Ax - |x| = b. \tag{1.2}$$

The AVE (1.2) has many applications in the field of optimization, such as linear complementarity problem, linear programming, complex quadratic programming, and binary matrix game. By now, the existence and uniqueness of the solution of the AVE have been comprehensively studied. For example, Mangasarian [1] proved that the AVE (1.2) has a unique solution if the smallest singular value of $A$ exceeds 1. In [2], it was established that the AVE (1.2) has a unique solution if and only if $A + (I - 2D)$ is nonsingular, where $D = \text{diag}(d_i)$, $d_i \in [-1, 1]$. Wu [3] gave some necessary and sufficient conditions for the unique solvability of the NGAVE (1.1). For solving the AVE (1.2), many algorithms are proposed, such as the Newton method and its other versions in [4] and [5], the NINE-method (Noor, Iqbal, Noor) in [6], and the SOR-like iteration method in [7].

To our knowledge, there does not exist a numberical method for solving the NGAVE (1.1) so far, and this is our motivation to write this paper. In this paper, we will try to extend the SOR-like iterative method to solve the NGAVE (1.1) and further to discuss the convergence condition for the SOR-like iterative method. Finally, the efficiency of the proposed method is verified by numerical experiments.

## 2. PRELIMINARIES

In this section, we give some symbol descriptions and lemmas to help the discussion later.
We introduce the following notation:

- $I$ is an identity matrix.

- $\rho(\,\cdot\,)$ is the spectral radius.

[*]E-mail: yangynnu@163.com
[**]E-mail: slwuynnu@126.com

- $\| \cdot \|$ is the 2-norm.

- $\sigma(\cdot)$ is a singular value.

- $\mathrm{diag}(x)$ is the diagonal matrix whose diagonal elements are components of the vector $x$.

**Lemma 2.1** [3]. *Assume that $A$ and $B$ are nonsingular and $\sigma_{\min}(AB^{-1}) > 1$. Then the NGAVE* (1.1) *has a unique solution for any $b \in \mathbb{R}^n$, where $\sigma_{\min}$ denotes the smallest singular value.*

**Lemma 2.2** [8]. *For the real quadratic equation $x^2 - bx + c = 0$, where $b, c \in \mathbb{R}$, both absolute values of the roots are less than one if and only if $|c| < 1$ and $|b| < 1 + c$.*

## 3. SOR-LIKE ITERATION METHOD

First, let $y = |Bx|$. Then we find that the NGAVE (1.1) is equivalent to

$$\begin{cases} Ax - y = b, \\ -|Bx| + y = 0. \end{cases} \tag{3.1}$$

We can rewrite (3.1) as

$$\overline{A}z := \begin{bmatrix} A & -I \\ -D(x) & I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix} := \overline{b}, \tag{3.2}$$

where $D(x) := \mathrm{diag}(\mathrm{sgn}(Bx))B$.

Let $\overline{A} = D - L - U$, where

$$D = \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix}, \qquad L = \begin{bmatrix} 0 & 0 \\ D(x) & 0 \end{bmatrix}, \qquad U = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}. \tag{3.3}$$

Thus, we obtain

$$(D - \omega L)z = [(1 - \omega)D + \omega U]z + \omega\overline{b}, \tag{3.4}$$

where $\omega > 0$. Equation (3.4) can be expanded as follows:

$$\begin{bmatrix} A & 0 \\ -\omega D(x) & I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} (1 - \omega)A & \omega I \\ 0 & (1 - \omega)I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \omega b \\ 0 \end{bmatrix}. \tag{3.5}$$

Based on (3.5), we can easily obtain the SOR-like method, which is described below:

$$\begin{cases} x^{k+1} = (1 - \omega)x^k + \omega A^{-1}(y^k + b), \\ y^{k+1} = (1 - \omega)y^k + w|Bx^{k+1}|, \end{cases} \tag{3.6}$$

where $k = 0, 1, \ldots, n$.

To study the convergence property of (3.6), let

$$M_\omega = (D - \omega L)^{-1}[(1 - \omega)D + \omega U].$$

Now we note that the eigenvalue of $M_\omega$ must be less than 1, and we need to find the values of the parameter $\omega$ satisfying this condition. To solve this problem, we need some lemmas.

**Lemma 3.1.** *Let $AB^{-1} \in \mathbb{R}^{n \times n}$ satisfy the assumptions of Lemma* 2.1. *If $\mu$ is an eigenvalue of the matrix $D(x)A^{-1}$, then $|\mu| < 1$.*

**Proof.** From Lemma 2.1, we know that $\sigma_{\min}(AB^{-1}) > 1$ is equivalent to $\|BA^{-1}\| < 1$. Further, we have

$$|\mu| \le \rho(D(x)A^{-1}) \le \|D(x)A^{-1}\| \le \|\mathrm{diag}(\mathrm{sgn}(Bx))\| \, \|BA^{-1}\| < 1.$$

This completes the proof. $\qquad\square$

**Lemma 3.2.** *Let $AB^{-1} \in \mathbb{R}^{n \times n}$ satisfy the assumptions of Lemma 2.1. If $\lambda$ is an eigenvalue of the matrix $M_\omega$, then $\lambda \neq 1$.*

**Proof.** Assume that a vector

$$\begin{bmatrix} x \\ y \end{bmatrix}$$

is an eigenvector of the matrix $M_\omega$ with the corresponding eigenvalue $\lambda = 1$. Then

$$M_\omega \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}, \tag{3.7}$$

that is,

$$\begin{cases} y = Ax, \\ y = D(x)x. \end{cases} \tag{3.8}$$

Deforming (3.8), we obtain

$$(I - D(x)A^{-1})y = 0. \tag{3.9}$$

From Lemma 3.1 it is clear that the matrix $I - D(x)A^{-1}$ is nonsingular. So we have $y = 0$ and $x = A^{-1}y = 0$. This contradicts the fact that

$$\begin{bmatrix} x \\ y \end{bmatrix}$$

is an eigenvector of the matrix $M_\omega$. Hence $\lambda \neq 1$. $\square$

**Lemma 3.3.** *Let $A \in \mathbb{R}^{n \times n}$ be nonsingular, and let $\omega > 0$. Assume that $\lambda$ is an eigenvalue of the matrix $M_\omega$ and $\lambda \neq 1 - \omega$. If $\mu$ satisfies the condition*

$$(\lambda + \omega - 1)^2 = \lambda \omega^2 \mu, \tag{3.10}$$

*then $\mu$ is an eigenvalue of $D(x)A^{-1}$. This conclusion can also be reversed: if $\mu$ is an eigenvalue of $D(x)A^{-1}$, then $\lambda$ is an eigenvalue of the matrix $M_\omega$ provided that* (3.10) *holds.*

**Proof.** From Lemma 3.2, assume that

$$\begin{bmatrix} x \\ y \end{bmatrix}$$

is an eigenvector of the matrix $M_\omega$ with the corresponding eigenvalue $\lambda$. Then

$$M_\omega \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} x \\ y \end{bmatrix}; \tag{3.11}$$

that is,

$$\begin{cases} \omega y = (\lambda + \omega - 1)Ax, \\ (\lambda + \omega - 1)y = \lambda \omega D(x)x. \end{cases} \tag{3.12}$$

Let $t = Ax$. Then from (3.12) we obtain

$$\frac{(\lambda + \omega - 1)^2}{\lambda \omega^2} t = D(x)A^{-1}t, \tag{3.13}$$

and so $\mu = (\lambda + \omega - 1)^2/(\lambda \omega^2)$ is an eigenvalue of the matrix $D(x)A^{-1}$.

We can use a similar method to prove the second assertion. $\square$

**Theorem 3.1.** *Let* $AB^{-1} \in \mathbb{R}^{n \times n}$ *satisfy the assumptions of Lemma* 2.1. *Assume that* $\mu$ *is an eigenvalue of the matrix* $D(x)A^{-1}$. *If* $\mu > 0$, *then the* SOR-*like iteration method converges for* $0 < \omega < 2$.

**Proof.** First, from (3.10) we obtain

$$\lambda^2 + \lambda(-\omega^2\mu + 2\omega - 2) + (\omega - 1)^2 = 0. \tag{3.14}$$

By Lemma 2.2, $|\lambda| < 1$ if and only if

$$\begin{cases} |\omega - 1|^2 < 1, \\ |-\omega^2\mu + 2\omega - 2| < 1 + (\omega - 1)^2. \end{cases} \tag{3.15}$$

Equation (3.15) is equivalent to the following form:

$$\begin{cases} 0 < \omega < 2, \\ -\omega^2 < -\omega^2\mu < (\omega - 2)^2. \end{cases} \tag{3.16}$$

Obviously, from (3.16) we find $0 < \omega < 2$. $\hspace{1cm}$ $\square$

To minimize the number of iterations of the SOR-like method, it is necessary to find the optimal relaxation parameters of the iterative equation. To this end, we have the following theorem.

**Theorem 3.2.** *Let* $AB^{-1} \in \mathbb{R}^{n \times n}$ *satisfy the assumptions of Lemma* 2.1, *and let* $\rho = \rho(D(x)A^{-1})$. *Assume also that* $\mu$ *is any eigenvalue of the matrix* $D(x)A^{-1}$ *and* $\mu$ *is positive. Then the parameter* $\omega_0 = 2/(1 + \sqrt{1 - \rho})$ *is optimal.*
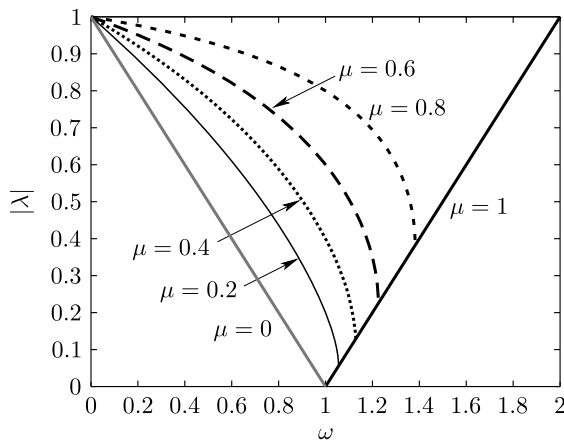
**Proof.** From (3.10) we obtain

$$\lambda = \frac{1}{2}\left[\omega^2\mu - 2\omega + 2 \pm \sqrt{(\omega^2\mu - 2\omega + 2)^2 - 4(\omega - 1)^2}\right]. \tag{3.17}$$

By calculating the modulus of $\lambda$, having in mind that the range of $\omega$ is $(0, 2)$, we have

$$|\lambda| = \begin{cases} |1 - \omega|, & \dfrac{2}{1 + \sqrt{1 - \mu}} \leq \omega < 2, \\ \dfrac{1}{2}\left[|\omega^2\mu - 2\omega + 2| + \sqrt{(\omega^2\mu - 2\omega + 2)^2 - 4(\omega - 1)^2}\right], & 0 < \omega < \dfrac{2}{1 + \sqrt{1 - \mu}}. \end{cases} \tag{3.18}$$

The graph of the function (3.18) is shown in Fig. 1.



**Fig. 1.** The value of $|\lambda|$ against $\omega$ for different values of $\mu$.

In Fig. 1, $|\lambda|$ denotes the vertical axis, and $\omega$ denotes the horizontal axis. Different values of $\mu$ represent different segmented function images, the larger $\mu$ is, the higher the position of segmented function will be. It is clear that the $|\lambda|$ is maximal when $\mu = \mu_{\max} = \rho$. Hence we have

$$\rho(M_\omega) = \begin{cases} |1 - \omega|, & \dfrac{2}{1 + \sqrt{1 - \rho}} \le \omega < 2, \\[2ex] \dfrac{1}{2}\Big[|\omega^2\rho - 2\omega + 2| + \sqrt{(\omega^2\rho - 2\omega + 2)^2 - 4(\omega - 1)^2}\Big], & 0 < \omega < \dfrac{2}{1 + \sqrt{1 - \rho}}. \end{cases} \tag{3.19}$$

For a given $\rho$, the $\rho(M_\omega)$ is a function of $\omega$, and we visualize (3.19) in Fig. 2.
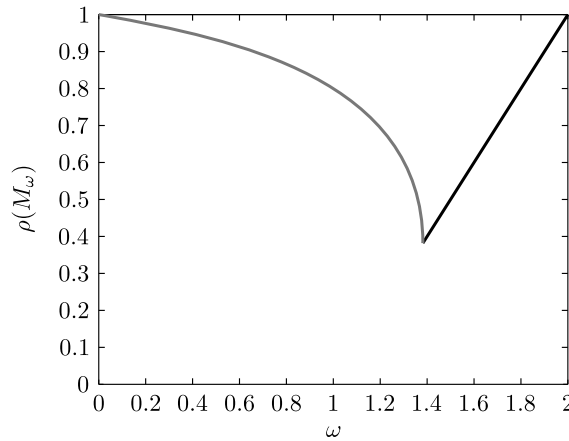


**Fig. 2.** $\rho(M_\omega)$ against $\omega$.

Obviously, $\rho_{\min}(M_\omega)$ is at the intersection of two curves, corresponding to $\omega = 2/(1 + \sqrt{1 - \rho})$.     □

## 4. NUMERICAL EXPERIMENTS

In this section, we test the efficiency of the optimal parameter by some numerical experiments. In these experiments, the initial vector is the zero vector, $\rho = \rho(AB^{-1})$, the error is expressed by $RES$, and it satisfies

$$RES = \|Ax^k + |Bx^k| - d\| < 10^{-6}.$$

The SOR-like iterative process will be terminated only for $RES \le 10^{-6}$ or after 500 iterations. All the tests are executed in MATLAB 2020b.

**Example 4.1** [9]**.** Let $A, B \in \mathbb{R}^{n \times n}$ and $d \in \mathbb{R}^{n \times 1}$, where

$$A = \begin{bmatrix} 8 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 8 & 1 & \cdots & 0 & 0 \\ 0 & 1 & 8 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 8 & 1 \\ 0 & 0 & 0 & \cdots & 1 & 8 \end{bmatrix}_{7 \times 7}, \quad B = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & 0 & \cdots & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & \cdots & 0 & 0 \\ 0 & \frac{1}{4} & \frac{1}{2} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & 0 & \cdots & \frac{1}{4} & \frac{1}{2} \end{bmatrix}_{7 \times 7}, \quad b = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$

We can calculate $\rho = 0.0977$ and $\omega_0 = 1.0257$ easily, and the solution of the corresponding equation is

$$(0.1223 \quad 0.1100 \quad 0.1112 \quad 0.1111 \quad 0.1112 \quad 0.1100 \quad 0.1223)^T.$$

Then we draw the corresponding plot as shown below.

The optimal parameter can be seen in Fig. 3, between 1.1 and 1.2; this is consistent with our theoretical optimal parameters $\omega_0$.
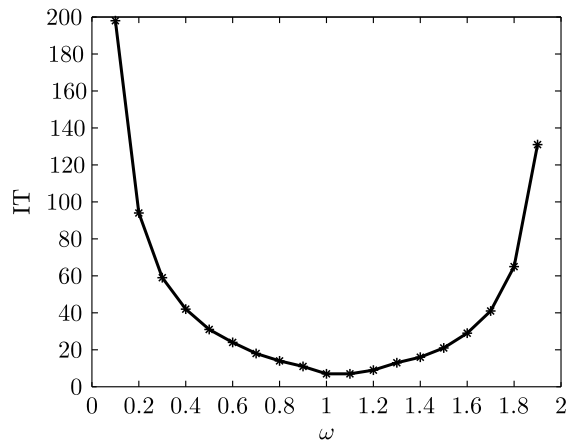
**Fig. 3.** IT and $\omega$ of Example 4.1.

**Example 4.2.** Let $A, B \in \mathbb{R}^{n \times n}$ and $d \in \mathbb{R}^{n \times 1}$, where

$$
A = \begin{bmatrix}
4 & -1 & 0 & \cdots & 0 & 0 \\
-1 & 4 & -1 & \cdots & 0 & 0 \\
0 & -1 & 4 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 4 & -1 \\
0 & 0 & 0 & \cdots & -1 & 4
\end{bmatrix}_{n \times n}, \quad
B = \begin{bmatrix}
\frac{1}{2} & \frac{1}{4} & 0 & \cdots & 0 & 0 \\
\frac{1}{4} & \frac{1}{2} & \frac{1}{4} & \cdots & 0 & 0 \\
0 & \frac{1}{4} & \frac{1}{2} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \frac{1}{2} & \frac{1}{4} \\
0 & 0 & 0 & \cdots & \frac{1}{4} & \frac{1}{2}
\end{bmatrix}_{n \times n}, \quad
b = \begin{bmatrix}
1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1
\end{bmatrix}_{n \times 1}.
$$

**Table 1.** Numerical results in Example 4.2

|          |            | 1000       | 2000       | 3000       | 4000       | 5000       |
|----------|------------|------------|------------|------------|------------|------------|
|          | $n$        |            |            |            |            |            |
| SOR-like | $\omega_0$ | 1.0263     | 1.0263     | 1.0263     | 1.0263     | 1.0263     |
|          | IT         | 8          | 8          | 8          | 8          | 8          |
|          | CPU(s)     | 0.0012     | 0.0019     | 0.0033     | 0.0051     | 0.0063     |
|          | RES        | 2.7262e-07 | 3.8641e-07 | 4.7361e-07 | 5.4708e-07 | 6.1179e-07 |
| GN       | IT         | 2          | 2          | 2          | 2          | 2          |
|          | CPU(s)     | 0.0092     | 0.0459     | 0.0833     | 0.1590     | 0.2322     |
|          | RES        | 1.9741e-14 | 2.8002e-14 | 3.4330e-14 | 3.9661e-14 | 4.4356e-14 |
| NINE     | IT         | 18         | 18         | 18         | 18         | 18         |
|          | CPU(s)     | 0.3073     | 0.9824     | 2.3326     | 3.9088     | 5.7240     |
|          | RES        | 3.4947e-07 | 4.9580e-07 | 6.0786e-07 | 7.0227e-07 | 7.8541e-07 |

In Table 1, we display results for different sizes of $n$. In this table, 'IT' denotes the number of iteration steps, and 'CPU' denotes the elapsed CPU time in seconds. From Table 1, we see that the iteration steps increase very slowly for different dimensions. The accuracy of the GN method is an order of magnitude higher than other methods and the number of iterations is the least. The NINA method

is not an advantage compared with other methods. Nevertheless, the SOR-like method can rapidly converge to the results under the specific parameter. In the elapsed CPU time, the SOR-like method is advantageous over the GN and NINA methods.

## CONCLUSIONS

In this paper, we mainly discuss the SOR-like iterative method for solving the new generalized absolute value equation and further consider the convergence conditions. For the problem of the optimal parameter, we give the formula to calculate it. Finally, numerical experiments confirm our theory, showing that our algorithm is efficient and feasible.

## ACKNOWLEDGMENTS

## FUNDING

## REFERENCES

1. O. L. Mangasarian and R. R. Meyer, "Absolute value equations," Linear Algebra Appl. **419** (2−3), 359−367 (2006).
2. S.-L. Wu and C.-X. Li, "The unique solution of the absolute value equations," Appl. Math. Lett. **76**, 195−200 (2018).
3. S.-L. Wu, "The unique solution of a class of the new generalized absolute value equation," Appl. Math. Lett. **116**, 107029 (2021).
4. O. L. Mangasarian, "A generalized Newton method for absolute value equations," Optim. Lett. **3** (1), 101−108 (2009).
5. H.-Y. Zhou, S.-L. Wu, and C.-X. Li, "Newton-based matrix splitting method for generalized absolute value equation," J. Comput. Appl. Math. **394**, 113578 (2021).
6. M. Noor, J. Iqbal, and K. Noor, "On an iterative method for solving absolute value equations," Optim. Lett. **6** (5), 1027−1033 (2012).
7. P. Guo, S.-L. Wu, and C.-X. Li, "On the SOR-like iteration method for solving absolute value equations," Appl. Math. Lett. **97**, 107−113 (2019).
8. D. M. Young, *Iterative Solution of Large Linear Systems* (Academic Press, New York, 2014).
9. Y.-F. Ke, "The new iteration algorithm for absolute value equation," Appl. Math. Lett. **99**, 105990 (2020).