Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

The Picard-HSS-SOR iteration method for absolute value equations

Lin Zheng[1*]

[*]Correspondence:
hbzhenglin@126.com
[1]School of Statistics and Applied Mathematics, Anhui University of Finance and Economics, Bengbu, 233030, China

Abstract

In this paper, we present the Picard-HSS-SOR iteration method for finding the solution of the absolute value equation (AVE), which is more efficient than the Picard-HSS iteration method for AVE. The convergence results of the Picard-HSS-SOR iteration method are proved under certain assumptions imposed on the involved parameter. Numerical experiments demonstrate that the Picard-HSS-SOR iteration method for solving absolute value equations is feasible and effective.

MSC: 65F10; 65F50; 90C33

Keywords: Absolute value equation; Picard-HSS-SOR iteration method; Convergence analysis

1 Introduction

Let $A \in R^{n\times n}$, $b \in R^n$. We consider the following absolute value equation (AVE):

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

where $|x|$ denotes the vector in R^n with absolute values of component of x. The AVE (1.1) is the special case of generalized system of absolute value equations of the form

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

where $B \in R^{n\times n}$. The system of absolute value equation (1.2) was introduced in [1] and investigated in a more general context in [2].

The importance of the AVE (1.1) arises from the fact that linear programs, bimatrix games and other important problems in optimization all can be reduced to the system of absolute value equations. In recent years, the problem of finding solution of AVE has been attracted much attention and has been studied in the literature [3–18]. For the numerical solution of the AVE (1.1), there exist many efficient numerical methods, such as the SOR-like iteration method [12], the relaxed nonlinear PHSS-like iteration method [15], the Levenberg–Marquardt method [16], the generalized Newton method [17], the Gauss–Seidel iteration method [19] and so on.

Recently, Salkuyeh [18] presented the Picard-HSS iteration method for solving the AVE and established the convergence theory under suitable conditions. Some numerical experiments showed in [18] that the Picard-HSS iteration method is more efficient than the Picard iteration method and generalized Newton method.

In the paper, we present a new version iteration method for finding the solution of the absolute value equation(AVE), which is more efficient than the Picard-HSS iteration method for AVE. The convergence results of the Picard-HSS-SOR iteration method are proved under certain assumptions imposed on the involved parameter. Numerical experiments demonstrate that the Picard-HSS-SOR iteration method for solving absolute value equations is feasible and effective.

This article is arranged as follows. In Sect. 2, we recall the Picard-HSS iteration method and some results that will be used in following analysis. The Picard-HSS-SOR iteration method and its convergence analysis are proposed in Sect. 3. Experimental results and conclusions are given in Sects. 4 and 5, respectively.

2 Preliminaries

Firstly, we present some notations and auxiliary results.

The symbol I_n denotes the $n \times n$ identity matrix. $\|A\|$ denotes the spectral norm defined by $\|A\| := \max\{\|Ax\| : x \in \mathbf{R}^n, \|x\| = 1\}$, where $\|x\|$ is the 2-norm. For $x \in \mathbf{R}^n$, sign(x) denotes a vector with components equal to -1, 0 or 1 depending on whether the corresponding component of x is negative, zero or positive. In addition, diag(sign(x)) is a diagonal matrix whose diagonal elements are sign(x). A matrix $A = (a_{ij}) \in R^{m\times n}$ 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 $1 \leq j \leq n$.

Proposition 2.1 ([2]) *Suppose that $A \in \mathbf{R}^{n\times n}$ is invertible. If $\|A^{-1}\| < 1$, then the AVE in* (1.1) *has a unique solution for any $b \in \mathbf{R}^n$.*

Lemma 2.1 ([20]) *For any vectors $x = (x_1, x_2, \dots, x_n)^T \in \mathbf{R}^n$ and $y = (y_1, y_2, \dots, y_n)^T \in \mathbf{R}^n$, the following results hold*:

(I) $\| |x| - |y| \| \leq \|x - y\|$; (II) *if* $0 \leq x \leq y$, *then* $\|x\| \leq \|y\|$;

(III) *if $x \leq y$ and P is a nonnegative matrix, then $Px \leq Py$, where $x \leq y$ denotes $x_i \leq y_i, 1 \leq i \leq n$.*

Let $A \in \mathbf{R}^{n\times n}$ be a non-Hermitian positive definite matrix. Then the matrix A possesses a Hermitian/skew-Hermitian (*HSS*) *splitting*

$$A = H + S,$$

where

$$H = \frac{1}{2}(A + A^H) \quad \textit{and} \quad S = \frac{1}{2}(A - A^H).$$

Algorithm 2.1 (The Picard-HSS iteration method [18]) *Given an initial guess $x^{(0)} \in \mathbf{R}^n$ and a sequence $\{l_k\}_{k=0}^{\infty}$ of positive integers, compute $x^{(k+1)}$ for $k = 0, 1, 2, \dots$, using the following iteration scheme until $\{x^{(k)}\}$ satisfies the following stopping criterion*:

(a) *Set* $x^{(k,0)} := x^{(k)}$;

(b) *for* $l = 0, 1, \ldots, l_k - 1$, *solve the following linear systems to obtain* $x^{(k,l+1)}$:

$$\begin{cases} (\alpha I + H)x^{(k,l+\frac{1}{2})} = (\alpha I - S)x^{(k,l)} + |x^{(k)}| + b, \\ (\alpha I + S)x^{(k,l+1)} = (\alpha I - H)x^{(k,l+\frac{1}{2})} + |x^{(k)}| + b, \end{cases} \tag{2.1}$$

where α *is a given positive constant*;

(c) *set* $x^{(k+1)} := x^{(k,l_k)}$.

The $(k+1)$*th iterate of the Picard-HSS method can be written as*

$$\begin{aligned} x^{(k+1)} &= T^{l_k}(\alpha)x^{(k)} + \sum_{j=0}^{l_k-1} T^j(\alpha)G(\alpha)\bigl(|x| + b\bigr) \\ &= T^{l_k}(\alpha)x^{(k)} + \bigl(I - T^{l_k}(\alpha)\bigr)A^{-1}\bigl(|x| + b\bigr), \quad k = 0, 1, 2, \ldots, \end{aligned} \tag{2.2}$$

where

$$T(\alpha) = (\alpha I + S)^{-1}(\alpha I - H)(\alpha I + H)^{-1}(\alpha I - S)$$

and

$$G(\alpha) = 2\alpha(\alpha I + S)^{-1}(\alpha I + H)^{-1}.$$

Theorem 2.1 ([18]) *Let* $A \in \mathbf{R}^{n\times n}$ *be a positive definite matrix. If* $\nu = \|A^{-1}\| < 1$, *then the AVE* (1.1) *has a unique solution* x^*, *and for any initial guess* $x^{(0)} \in \mathbf{R}^n$ *and any sequence of positive integers* $l_k, k = 0, 1, 2, \ldots$, *the iteration sequence* $\{x^{(k)}\}$ *generated by the Picard-HSS iteration method converges to* x^* *provided that* $\widetilde{l} = \liminf_{k\to+\infty} l_k \geq N$, *where* N *is a natural number satisfying*

$$\bigl\|T^s(\alpha)\bigr\| < \frac{1-\nu}{1+\nu}, \quad \forall s \geq N. \tag{2.3}$$

3 The Picard-HSS-SOR iteration method

In the section, we will introduce the Picard-HSS-SOR iteration method and prove the convergence of the proposed method.

Recently, Ke et al. presented the SOR-like method for solving (1.1) in [13]. Let $y = |x|$, then the AVE in (1.1) is equivalent to

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

that is,

$$\mathcal{A}\mathbf{z} := \begin{pmatrix} A & -I_n \\ -D(x) & I_n \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix} := \mathbf{b}, \tag{3.2}$$

where $D(x) := \operatorname{diag}(\operatorname{sign}(x)), x \in \mathbf{R}^n$.

Based on Eq. (3.2), we present the Picard-HSS-SOR iteration method for solving AVE (3.1) as follows.

Algorithm 3.1 (The Picard-HSS-SOR iteration method) *Let $A \in \mathbf{R}^{n\times n}$ be a positive definite matrix with $H = \frac{1}{2}(A+A^T)$ and $S = \frac{1}{2}(A-A^T)$ being the Hermitian and skew-Hermitian parts of the matrix A, respectively. Given an initial guess $x^{(0)} \in \mathbf{R}^n$ and $y^{(0)} \in \mathbf{R}^n$. Compute $\{(x^{(k+1)}, y^{(k+1)})\}$ for $k = 0, 1, 2, \ldots$, using the following iteration scheme until $\{(x^{(k)}, y^{(k)})\}$ satisfies the stopping criterion*:

(i) *Set $x^{(k,0)} := x^{(k)}$*;

(ii) *for $l = 0, 1, \ldots, l_k - 1$, solve the following linear systems to obtain $x^{(k,l+1)}$*:

$$\begin{cases} (\alpha I + H)x^{(k,l+\frac{1}{2})} = (\alpha I - S)x^{(k,l)} + y^{(k)} + b, \\ (\alpha I + S)x^{(k,l+1)} = (\alpha I - H)x^{(k,l+\frac{1}{2})} + y^{(k)} + b; \end{cases} \tag{3.3}$$

(iii) *set*

$$\begin{cases} x^{(k+1)} = x^{(k,l_k)}, \\ y^{(k+1)} = (1-\tau)y^{(k)} + \tau|x^{(k+1)}|, \end{cases} \tag{3.4}$$

where $\alpha > 0$ and $0 < \tau < 2$.

Let (x^*, y^*) be the solution pair of the nonlinear equation (3.1) and $(x^{(k)}, y^{(k)})$ be produced by the Algorithm 3.1. Define the iteration errors

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

Next, we will prove the main result of this paper.

Theorem 3.1 *Let $\nu = \|A^{-1}\|$, $\beta = |1-\tau|$ and $\widetilde{l} = \liminf_{k\to+\infty} l_k \geq N$, where N is a natural number satisfying (2.3). If $0 < \tau < 2$ and*

$$4\beta\tau\nu + \left(1+\tau^2\right)\left(1+4\nu^2\right) < 1, \tag{3.5}$$

then the inequality

$$\left\| \begin{pmatrix} e_{k+1}^x \\ e_{k+1}^y \end{pmatrix} \right\| < \left\| \begin{pmatrix} e_k^x \\ e_k^y \end{pmatrix} \right\| \tag{3.6}$$

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

Proof The $(k+1)$th iterate of the Picard-HSS-SOR iteration method can be written as

$$\begin{cases} x^{(k+1)} = T^{l_k}(\alpha)x^{(k)} + (I - T^{l_k}(\alpha))A^{-1}(y^{(k)} + b), \\ y^{(k+1)} = (1-\tau)y^{(k)} + \tau|x^{(k+1)}|, \end{cases} \tag{3.7}$$

where

$$T(\alpha) = (\alpha I + S)^{-1}(\alpha I - H)(\alpha I + H)^{-1}(\alpha I - S).$$

Since (x^*, y^*) is the solution pair of the nonlinear equation (3.1), from (3.7), we can obtain

$$e^x_{k+1} = T^{l_k}(\alpha)e^x_k + \bigl(I - T^{l_k}(\alpha)\bigr)A^{-1}e^y_k, \tag{3.8}$$

$$e^y_{k+1} = (1-\tau)e^y_k + \tau\bigl(|x^*| - |x^{(k+1)}|\bigr). \tag{3.9}$$

From Lemma 2.1 and (3.9), we can obtain

$$\begin{aligned} \|e^y_{k+1}\| &\le |1-\tau| \cdot \|e^y_k\| + \tau \bigl\| |x^*| - |x^{(k+1)}| \bigr\| \\ &\le |1-\tau| \cdot \|e^y_k\| + \tau \|x^* - x^{(k+1)}\| \\ &= \beta \cdot \|e^y_k\| + \tau \|e^x_{k+1}\|. \end{aligned} \tag{3.10}$$

From Theorem 2.1 and (3.8), we have

$$\begin{aligned} \|e^x_{k+1}\| &\le \|T^{l_k}(\alpha)\| \|e^x_k\| + \bigl(1 + \|T^{l_k}(\alpha)\|\bigr) \|A^{-1}\| \|e^y_k\| \\ &\le \|e^x_k\| + 2\nu \|e^y_k\|. \end{aligned} \tag{3.11}$$

Therefore, from (3.10) and (3.11), we have

$$\begin{pmatrix} 1 & 0 \\ -\tau & 1 \end{pmatrix} \begin{pmatrix} \|e^x_{k+1}\| \\ \|e^y_{k+1}\| \end{pmatrix} \le \begin{pmatrix} 1 & 2\nu \\ 0 & \beta \end{pmatrix} \begin{pmatrix} \|e^x_k\| \\ \|e^y_k\| \end{pmatrix}. \tag{3.12}$$

Let $P = \left(\begin{smallmatrix} 1 & 0 \\ \tau & 1 \end{smallmatrix}\right)$. In this case we have P is nonnegative, i.e. $P \ge 0$.

According to Lemma 2.1, multiplying (3.12) from left by the nonnegative matrix P, we can obtain

$$\begin{pmatrix} \|e^x_{k+1}\| \\ \|e^y_{k+1}\| \end{pmatrix} \le \begin{pmatrix} 1 & 2\nu \\ \tau & \beta + 2\tau\nu \end{pmatrix} \begin{pmatrix} \|e^x_k\| \\ \|e^y_k\| \end{pmatrix}. \tag{3.13}$$

Let

$$W = \begin{pmatrix} 1 & 2\nu \\ \tau & \beta + 2\tau\nu \end{pmatrix},$$

thus, we get

$$\begin{pmatrix} \|e^x_{k+1}\| \\ \|e^y_{k+1}\| \end{pmatrix} \le \|W\| \begin{pmatrix} \|e^x_k\| \\ \|e^y_k\| \end{pmatrix}. \tag{3.14}$$

Next, we will consider the choice of the parameter τ such that $\|W\| < 1$, therefore the inequality (3.6) holds. Since

$$W^T W = \begin{pmatrix} 1 + \tau^2 & \tau\beta + 2\nu(1+\tau^2) \\ \tau\beta + 2\nu(1+\tau^2) & 4\nu^2 + (\beta + 2\tau\nu)^2 \end{pmatrix},$$

we can obtain

$$\det\bigl(W^T W\bigr) = \beta^2 \tag{3.15}$$

and

$$\operatorname{tr}\bigl(W^T W\bigr) = \beta^2 + 4\tau\nu\beta + \bigl(1+\tau^2\bigr)\bigl(1+4\nu^2\bigr). \tag{3.16}$$

Suppose λ is an eigenvalue of the matrix $W^T W$ with $\lambda \geq 0$. Therefore λ will satisfy

$$\lambda^2 - \operatorname{tr}\bigl(W^T W\bigr)\lambda + \det\bigl(W^T W\bigr) = 0. \tag{3.17}$$

Thus, we can obtain the following relations:

$$\lambda_1 + \lambda_2 = \operatorname{tr}\bigl(W^T W\bigr), \qquad \lambda_1\lambda_2 = \det\bigl(W^T W\bigr),$$

where λ_1 and λ_2 are eigenvalues of the matrix $W^T W$.

If $0 < \tau < 2$, we have $\det(W^T W) < 1$, that is, $0 \leq \lambda_1\lambda_2 < 1$.

From (3.5), we have $\lambda_1 + \lambda_2 < 1 + \lambda_1\lambda_2$, that is, $(\lambda_1 - 1)(\lambda_2 - 1) > 0$. Hence, we can obtain

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

Therefore $\|W\| < 1$. The proof is completed. □

4 Numerical results

To illustrate the implementation and efficiency of the Picard-HSS-SOR iteration method, we test the following test problems. All test problems are performed by MATLAB R2019a on a personal computer with 2.4 GHz central processing unit (Intel (R) Core (TM) i5-3210M), 8GB memory. We use a null vector as initial guess and all the experiments are terminated if the current iterations satisfy

$$\frac{\|b + |x^{(k)}| - Ax^{(k)}\|_2}{\|b\|_2} \leq 10^{-6},$$

or if the number of the prescribed iteration steps $k_{\max} = 500$ is exceeded. In addition, the stopping criterion for the inner iterations is set to be

$$\frac{\|b^{(k)} - As^{(k,l_k)}\|_2}{\|b^{(k)}\|_2} \leq 0.01,$$

where $b^{(k)} = |x^{(k)}| + b - Ax^{(k,l_k)}$, $s^{(k,l_k)} = x^{(k,l_k)} - x^{(k,l_k-1)}$, l_k is the number of inner iteration steps and a maximum number of iterations 10.

Next,we consider the two-dimensional convection-diffusion equation

$$\begin{cases} -(u_{xx} + u_{yy}) + q(u_x + u_y) + pu = f(x,y), & (x,y) \in \Omega, \\ u(x,y) = 0, & (x,y) \in \partial\Omega, \end{cases}$$

where $\Omega = (0,1) \times (0,1)$, $\partial\Omega$ is its boundary, q is a positive constant used to measure the magnitude of the diffusive term and p is a real number. We apply the five-point finite difference scheme to the diffusive terms and the central difference scheme to the convective terms. Let $h = 1/(m+1)$ and $\mathrm{Re} = (qh)/2$ denote the equidistant step size and the mesh

Table 1 Numerical results for the test problem with different values of m and q ($p = 0$)

	Methods		$m = 10$	$m = 20$	$m = 30$	$m = 40$
$q = 0$	PHSS	IT	10	12	14	15
		CPU	0.0458	0.9178	7.9636	37.1752
		RES	2.1e-5	4.9e-5	5.6e-5	1.7e-4
	PHSSR	IT	2	3	3	4
		CPU	0.0090	0.2213	2.1105	9.4696
		RES	4.7e-5	1.6e-5	6.1e-5	1.4e-4
$q = 1$	PHSS	IT	9	12	14	15
		CPU	0.03959	0.9171	8.2287	38.8795
		RES	2.5e-5	3.7e-5	4.7e-5	1.2e-4
	PHSSR	IT	2	3	4	4
		CPU	0.0092	0.2236	2.3239	10.4736
		RES	4.5e-5	3.0e-5	9.8e-5	1.6e-4
$q = 10$	PHSS	IT	6	11	13	15
		CPU	0.0283	0.8427	7.4282	45.1799
		RES	3.6e-5	3.7e-5	4.6e-5	6.5e-5
	PHSSR	IT	3	3	4	4
		CPU	0.0137	0.2481	2.3239	9.3871
		RES	6.0e-6	7.3e-5	9.8e-5	1.1e-4
$q = 100$	PHSS	IT	19	8	9	11
		CPU	0.1015	0.6448	5.2356	31.6856
		RES	8.7e-4	1.9e-4	2.0e-4	7.1e-5
	PHSSR	IT	3	3	6	4
		CPU	0.0139	0.2743	3.7824	9.3845
		RES	2.0e-4	9.4e-5	3.0e-5	7.1e-5

Reynolds number, respectively. Then we get a system of linear equations $Bx = d$, where B is a matrix of order $n = m^2$ of the form

$$B = T_x \otimes I_m + I_m \otimes T_y + pI_n,$$

with

$$T_x = \operatorname{tridiag}(t_2, t_1, t_3)_{m\times m} \quad \text{and} \quad T_y = \operatorname{tridiag}(t_2, 0, t_3)_{m\times m},$$

where $t_1 = 4$, $t_2 = -1 - \mathrm{Re}$, $t_3 = -1 + \mathrm{Re}$, I_m and I_n are the identity matrices of order m and n, respectively, $\otimes$ means the Kronecker product.

For our numerical experiments, we set $A = B + \frac{1}{2}(L - L^T)$, where L is the strictly lower part of B, and the right hand side vector b of the AVE(1.1) is taken in such a way that the vector $x = (x_1, x_2, \dots, x_n)^T$ with $x_k = (-1)^k$, $k = 1, 2, \dots, n$, being the exact solution. It is easy to see that the matrix A is non-symmetric positive definite.

The computation of the optimal parameter is often problem-dependent and generally difficult to determine. The optimal parameter α and τ employed in each method is experimentally determined such that it results in the least number of iterations.

In Tables 1 and 2, we present the numerical results with respect to the Picard-HSS (PHSS) and the Picard-HSS-SOR (PHSSR) iterations. We give the elapsed CPU time in seconds for the convergence (denoted CPU), the norm of absolute residual vectors (denoted RES), and the number of iteration steps (denoted IT).

From Tables 1 and 2, we can see that the Picard-HSS-SOR (PHSSR) iteration method takes fewer iterations and CPU times than the Picard-HSS iteration method. It means the PHSSR iteration method for solving absolute value equations is feasible and effective.

Table 2 Numerical results for the test problem with different values of m and q ($p = 0.5$)

	Methods		$m = 10$	$m = 20$	$m = 30$	$m = 40$
$q = 0$	PHSS	IT	9	11	13	14
		CPU	0.0384	0.8241	9.1203	31.8901
		RES	4.5e-5	6.2e-5	5.9e-5	1.2e-4
	PHSSR	IT	2	5	2	2
		CPU	0.0092	0.3994	1.2641	6.6407
		RES	1.6e-7	1.5e-4	1.5e-5	6.6e-5
$q = 1$	PHSS	IT	9	11	13	14
		CPU	0.0386	0.8310	8.8706	34.4268
		RES	3.6e-5	5.3e-5	5.2e-5	1.2e-4
	PHSSR	IT	2	2	2	2
		CPU	0.0092	0.1765	1.2702	5.3334
		RES	4.2e-7	4.0e-6	2.2e-5	1.8e-4
$q = 10$	PHSS	IT	9	11	12	13
		CPU	0.0389	0.8306	8.1935	30.2050
		RES	1.5e-5	1.5e-5	8.9e-5	1.4e-4
	PHSSR	IT	2	2	2	2
		CPU	0.0093	0.1546	1.2707	4.8570
		RES	2.1e-5	3.2e-5	3.9e-5	4.5e-5
$q = 100$	PHSS	IT	6	8	9	10
		CPU	0.0252	0.6039	5.2415	24.6826
		RES	1.8e-4	1.1e-4	1.0e-4	2.4e-04
	PHSSR	IT	2	2	2	2
		CPU	0.0093	0.1526	1.2538	5.3998
		RES	3.9e-5	1.2e-5	9.4e-5	1.1e-4

5 Conclusions

In this paper, the Picard-HSS-SOR iteration method is presented to solve the absolute value equation, which is more efficient than the Picard-HSS iteration method. We proved the convergence results of the Picard-HSS-SOR iteration method under certain assumptions. Finally, numerical experiments were also implemented so as to check the effective of the proposed method.

Funding
The work are supported by The Natural Science Foundation of Education Bureau of Anhui Province (KJ2020A0017, KJ2017A432).

Availability of data and materials
Not applicable.

Competing interests
The author declares that there is no conflict of interests. The author declares that there is no competing interests.

Authors' contributions
The author carried out the results, and read and approved the current version of the manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 24 September 2020 Accepted: 1 December 2020 Published online: 09 December 2020

References

1. Rohn, J.: A theorem of the alternatives for the equation $Ax + B|x| = b$. Linear Multilinear Algebra **52**, 421–426 (2004)
2. Mangasarian, O.L., Meyer, R.R.: Absolute value equations. Linear Algebra Appl. **419**, 359–367 (2006)
3. Rohn, J.: An algorithm for computing all solutions of an absolute value equation. Optim. Lett. **6**, 851–856 (2012)
4. Rohn, J., Hooshyarbakhsh, V., Farhadsefat, R.: An iterative method for solving absolute value equations and sufficient conditions for unique solvability. Optim. Lett. **8**, 35–44 (2014)
5. Noor, M.A., Iqbal, J., Noor, K.I.: On an iterative method for solving absolute value equations. Optim. Lett. **6**, 1027–1033 (2012)

6. Zainali, N., Lotfi, T.: On developing a stable and quadratic convergent method for solving absolute value equation. J. Comput. Appl. Math. **330**, 742–747 (2018)
7. Mangasarian, O.L.: Sufficient conditions for the unsolvability and solvability of the absolute value equation. Optim. Lett. **11**, 1–7 (2017)
8. Wu, S.L., Li, C.X.: The unique solution of the absolute value equations. Appl. Math. Lett. **76**, 195–200 (2018)
9. Haghani, F.K.: On generalized Traub's method for absolute value equations. J. Optim. Theory Appl. **166**, 619–625 (2015)
10. Li, C.X.: A modified generalized Newton method for absolute value equations. J. Optim. Theory Appl. **170**, 1055–1059 (2016)
11. Tang, J.Y., Zhou, J.C.: A quadratically convergent descent method for the absolute value equation $Ax + B|x| = b$. Oper. Res. Lett. **47**, 229–234 (2019)
12. Guo, P., Wu, S.L., Li, C.X.: On the SOR-like iteration method for solving absolute value equations. Appl. Math. Lett. **97**, 107–113 (2019)
13. Ke, Y.F., Ma, C.F.: SOR-like iteration method for solving absolute value equations. Appl. Math. Comput. **311**, 195–202 (2017)
14. Ke, Y.F.: The new iteration algorithm for absolute value equation. Appl. Math. Lett. **99**, 105990 (2020)
15. Zhang, J.J.: The relaxed nonlinear PHSS-like iteration method for absolute value equations. Appl. Math. Comput. **265**, 266–274 (2015)
16. Iqbal, J., Iqbal, A., Arif, M.: Levenberg–Marquardt method for solving systems of absolute value equations. J. Comput. Appl. Math. **282**, 134–138 (2015)
17. Mangasarian, O.L.: A generalized Newton method for absolute value equations. Optim. Lett. **3**, 101–108 (2009)
18. Salkuyeh, D.K.: The Picard-HSS iteration method for absolute value equations. Optim. Lett. **8**, 2191–2202 (2014)
19. Edalatpour, V., Hezari, D., Salkuyeh, D.K.: A generalization of the Gauss–Seidel iteration method for solving absolute value equations. Appl. Math. Comput. **293**, 156–167 (2017)
20. Berman, A., Plemmons, R.: Nonnegative Matrices in the Mathematical Sciences. Academic Press, New York (1979)