Journal of Applied Analysis and Computation
Volume 15, Number 2, April 2025, 896–914
Website:http://www.jaac-online.com
DOI:10.11948/20240198

THE NEWTON-BASED MATRIX SPLITTING ITERATIVE METHOD FOR SOLVING GENERALIZED ABSOLUTE VALUE EQUATION WITH NONLINEAR TERM*

Changfeng Ma[1,†], Yuhong Wu[2] and Yajun Xie[3,†]

Abstract A new Newton-based matrix splitting iterative method is proposed for solving generalized absolute value equation with nonlinear term. We give the global convergence of this method. Further some new convergence conditions are proposed when $A = M - N$ is an H-compatible splitting. Numerical results indicate that the new Newton-based matrix splitting iterative method for solving generalized absolute value equation with nonlinear term is effective.

Keywords Generalized absolute value equation with nonlinear term, Newton method, unique solvability, global convergence, numerical experiments.

MSC(2010) 65H10, 90C30.

1. Introduction

In this paper, we consider the following generalized absolute value equation with nonlinear term (NGAVE):

$$Ax - B|x| + f(x) = b, \tag{1.1}$$

where $A, B \in \mathbb{R}^{n\times n}, b \in \mathbb{R}^n, f : \mathbb{R}^n \mapsto \mathbb{R}^n$ are known, $x \in \mathbb{R}^n$ is unknown. Here, $|x| = (|x_1|, |x_2|, \cdots, |x_n|)^T$. While $f(x) = c \in \mathbb{R}^n$ in (1.1), NGAVE becomes generalized absolute value equation (GAVE)

$$Ax - B|x| = e, \tag{1.2}$$

where $e \in \mathbb{R}^n$. When $f(x) = c \in \mathbb{R}^n, B = I$ in (1.1), NGAVE becomes to absolute value equation (AVE)

$$Ax - |x| = f, \tag{1.3}$$

†The corresponding authors.
[1]School of Data and Computer Science, Xiamen Institute of Technology, Xiamen 361021, China
[2]School of Mathematics and Statistics, Fujian Normal University, Fuzhou 350117, China
[3]School of Big data, Fuzhou University of International Studies and Trade, Fuzhou 350202, China
*The authors were supported by National Natural Science Foundation of China (12371378) and Natural Science Foundation of Fujian Province, China (2024J01980).
Email: macf@fjnu.edu.cn(C. Ma), 921208406@qq.com(Y. Wu), xyj@fzfu.edu.cn(Y. Xie)

in [16] where $f \in \mathbb{R}^n$. Absolute value equation may arise in diverse fields, including complementarity problem, programming problem, and so on, see [1,3,4,11–13,17]. For instance, on the basis of the equivalence of $a \geqslant 0, b \geqslant 0, ab = 0$ and $a+b = |a-b|$, we known that NGAVE (1.1) also can be reformulated as a generalized nonlinear complementarity problems (GNCP):

$$H(x)^T Z(x) = 0, H(x) = Hx + b^{'} + f^{'}(x) \geqslant 0, Z(x) = Zx - b^{'} + f^{'}(x) \geqslant 0, \quad (1.4)$$

where $H = \frac{B^{-1}A+I}{2}, Z = \frac{B^{-1}A-I}{2}, b^{'} = \frac{B^{-1}b}{2}, f^{'}(x) = \frac{B^{-1}f(x)}{2}$.

Recently, in [10], AVE (1.3) is expressed as the nonlinear equation

$$F(x) = Ax - |x| - b = 0, \quad (1.5)$$

and using the Newton iterative method $x^{(k+1)} = x^{(k)} - F^{'}(x^{(k)})^{-1} F(x^{(k)})$, then generalized Newton method (GN)

$$x^{(k+1)} = x^{(k)} - (A - D(x^{(k)}))^{-1}(Ax^{(k)} - |x^{(k)}| - b) \quad (1.6)$$

is obtained, where $F^{'}(x^{(k)})$ denote the Jacobin of F at $x^{(k)}$ and $D(x^{(k)}) = \text{diag}(\text{sign}(x^{(k)}))$. In the calculation, due to the change of matrix $A - D(x^{(k)})$ in the GN method, the computations of the generalized Newton method may be very expensive. To avoid changing the Jacobian, Wang, Cao and Chen utilize $A + \Omega$ as the approximation of $F^{'}(x^{(k)})$ and then get the modified Newton method (MN):

$$x^{(k+1)} = x^{(k)} - (A + \Omega)^{-1}(Ax^{(k)} - |x^{(k)}| - b), \quad (1.7)$$

Ω is positive semi-definite here.

This method does not need to recalculate $F^{'}(x^{(k)})$ at every step, thus reducing the amount of calculation.

But if $A + \Omega$ is ill-conditioned, the MN method may be expensive in practical calculations. Furthermore, in [23], the author proposes a Newton-basedd matrix splitting method

$$x^{(k+1)} = x^{(k)} - (M + \Omega)^{-1}(Ax^{(k)} - |x^{(k)}| - b), \quad (1.8)$$

where $A = M - N$.

Besides, a block matrix splitting method is proposed to solve the absolute value equation in [6]. The AVE (1.3) is equivalent to the block system

$$\begin{pmatrix} A & -I \\ -D(x) & I \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix}, \quad (1.9)$$

where $D(x) = \text{diag}(\text{sign}(x))$. When A is invertible, from (1.9), it is hold that

$$\begin{cases} x^{(k+1)} = A^{-1}(y^{(k)} + b), \\ y^{(k+1)} = (1 - \tau)y^{(k)} + \tau|x^{(k+1)}|. \end{cases} \quad (1.10)$$

Based on the above methods, a new iterative scheme is constructed in this paper by using the characteristics of NGAVE with nonlinear term $f(x)$ to solve NGAVE

(1.1). We give a sufficient condition for the existence of a unique solution to NGAVE and several cases in which the new iterative method converges. Some sufficient conditions for convergence of the method are given when the splitting is H-compatible splitting.

The rest of the paper is organized as follows: In the remainder of this section, some definitions and notations are given, which are suitable for the later discussion. In section 2, sufficient conditions for the existence of the unique solution of NGAVE (1.1) are given. In Section 3, we propose the algorithm for solving NGAVE (1.1). In Section 4, the convergence of this iterative method is discussed. In Section 5, the numerical results of proposed method is reported. In Section 6, we summarize the work done in this paper.

Next, we provide some definitions and notations to conclude this section.

Let $A = (a_{ij}) \in \mathbb{R}^{n\times n}$. A is called as a Z-matrix if $a_{ij} \leqslant 0$ for $i \neq j$. A is a Z-matrix and $A^{-1} \geqslant 0$ then it is called a nonsingular M-matrix. Further, A is called an H-matrix when its comparison matrix $\langle A\rangle = (\langle a\rangle_{ij}) \in \mathbb{R}^{n\times n}$ is a nonsingular M-matrix where $\langle a\rangle_{ii} = |a_{ii}|$ and $\langle a\rangle_{ij} = -|a_{ij}|$ for $i \neq j$. A is an H_+-matrix which is an H-matrix with $\mathrm{diag}(A) > 0$, see [2]. The strictly diagonally dominant (SDD) matrix is defined as $|a_{ii}| > \sum_{j\neq i} |a_{ij}|, i = 1, 2, \cdots, n$. $\rho(A)$ and $\|A\|$ denote the spectral radius and the 2-norm of the matrix A, respectively.

2. Unique solvability of the generalized absolute value equation with nonlinear term

NGAVE (1.1) can be reformulated as a fixed point equation

$$x = A^{-1}(B|x| - f(x) + b) := G(x), \tag{2.1}$$

where A is nonsingular.

First, the following definition and assumption are given

Definition 2.1. [5] There is a constant $L \neq 0$ that satisfy

$$\|F(x) - F(y)\| \leqslant L\|x - y\|,$$

then F is called Lipschitz continuous.

Assumption 2.1. [7] *The nonlinear term $f(x)$ in the NGAVE (1.1) is Lipschitz continuous.*

Definition 2.2. [7] Let $G : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}^n$. If there is $\alpha \in (0, 1)$ that satisfy $\|G(x) - G(y)\| \leqslant \alpha\|x - y\|$ for any $x, y \in \mathcal{D}_0 \subset \mathcal{D}$, G is called a contractive mapping in $\mathcal{D}_0$ and α is called a compression coefficient.

Lemma 2.1. [7] *(Contraction Mapping Principle) Let $G : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}^n$ is a contractive mapping in closed set $\mathcal{D}_0 \subset \mathcal{D}$ and $G(\mathcal{D}_0) \subset \mathcal{D}_0$, then G has a unique fixed point in $\mathcal{D}_0$.*

Lemma 2.2. *For any vectors $x \in \mathbb{R}^n, y \in \mathbb{R}^n$, it holds that $\||x| - |y|\| \leqslant \|x - y\|$.*

Proof. According to the definition of vector norm and inequation $||\alpha| - |\beta|| \leqslant |\alpha - \beta|$, this conclusion can be directly obtained. □

Theorem 2.1. *Under the condition of Assumption 2.1, if* $\|A^{-1}\|(\|B\| + L) < 1$, *then NGAVE (1.1) has a unique solution.*

Proof. According to Assumption 2.1, it can be seen that there is a constant $L \neq 0$ which satisfies $\|f(x) - f(y)\| \leqslant L\|x - y\|$.

For any $x, y \in \mathbb{R}^n$, we have

$$\begin{aligned}\|G(x) - G(y)\| &= \|A^{-1}B|x| - A^{-1}f(x) - A^{-1}B|y| + A^{-1}f(y)\| \\ &= \|A^{-1}B(|x| - |y|) - A^{-1}(f(x) - f(y))\| \\ &\leqslant \|A^{-1}\|(\|B\| + L)\|x - y\|.\end{aligned} \tag{2.2}$$

The last inequality holds according to the Lemma 2.2. Let $\|A^{-1}\|(\|B\| + L) = \alpha_0$. Then by Definition 2.2, it follows that G is a contractive mapping in $\mathbb{R}^n$. According to Lemma 2.1, $G(x)$ has a unique fixed point. Thus NGAVE (1.1) has a unique solution. □

3. Solve generalized absolute value equations with nonlinear term

In this section, we first propose a new iterative method to solve absolute value equation with nonlinear term.

We express the matrix A as $A = M - N$. Using matrix $M + \Omega$ to approximate the Jacobi matrix $F^{'}(x^k)$, we have the following iteration form.

Algorithm 3.1. (A new Newton-based Matrix Splitting Iterative Method)

Step 1. Give initial point $x^{(0)}, y^{(0)} \in \mathbb{R}^n$ and the parameter $\varepsilon > 0$. Assume the split of the matrix A is $A = M - N$. Given $\Omega \in \mathbb{R}^{n\times n}$ which satisfies $M + \Omega$ is invertible. Set $k = 0$.

Step 2. If $\frac{\|Ax^{(k)} - B|x^{(k)}| + f(x^{(k)}) - b\|}{\|b\|} < \varepsilon$, stop;

Step 3. Compute $x^{(k+1)}$ and $y^{(k+1)}$ by

$$\begin{cases} x^{(k+1)} = (M + \Omega)^{-1}\left((N + \Omega)x^{(k)} + B|x^{(k)}| - y^{(k)} + b\right), \\ y^{(k+1)} = \dfrac{1}{\alpha + 1}(f(x^{(k+1)}) + \alpha y^{(k)}), \end{cases} \tag{3.1}$$

where α is a positive real number.

Step 4. Set $k := k + 1$ and go to Step 2.

Algorithm 3.1 produces different iterative forms for different ways of splitting.

(1) When $M = A, N = 0$, Algorithm 3.1 turns into the improvement of the method in [19]

$$\begin{cases} x^{(k+1)} = A^{-1}\left(B\left|x^{(k)}\right| - y^{(k)} + b\right), \\ y^{(k+1)} = \dfrac{1}{\alpha + 1}(f(x^{(k+1)}) + \alpha y^{(k)}), \end{cases} \tag{3.2}$$

which can be called an improved Newton-based Method.

(2) When $M = D, N = L+U$ where $D = \text{diag}(A)$, $-L, -U$ represent the strictly lower-triangular and upper-triangular part of A, respectively, Algorithm 3.1 will be expressed as

$$\begin{cases} x^{(k+1)} = (D+\Omega)^{-1}\left((L+U+\Omega)x^{(k)} + B\left|x^{(k)}\right| - y^{(k)} + b\right), \\ y^{(k+1)} = \dfrac{1}{\alpha+1}(f(x^{(k+1)}) + \alpha y^{(k)}), \end{cases} \tag{3.3}$$

which can be called a new Newton-based Jacobi Method (NNJ).

(3) When $M = D-L, N = U$ where $D = \text{diag}(A)$, $-L, -U$ represent the strictly lower-triangular and upper-triangular part of A, respectively, Algorithm 3.1 will be expressed as

$$\begin{cases} x^{(k+1)} = (D-L+\Omega)^{-1}\left((U+\Omega)x^{(k)} + B\left|x^{(k)}\right| - y^{(k)} + b\right), \\ y^{(k+1)} = \dfrac{1}{\alpha+1}(f(x^{(k+1)}) + \alpha y^{(k)}), \end{cases} \tag{3.4}$$

which can be called a new Newton-based Gauss-Seidel Method (NNGS).

(4) When $M = \frac{1}{\alpha'}D - L, N = (\frac{1}{\alpha'} - 1)D + U$ where $D = \text{diag}(A)$, $-L, -U$ represent the strictly lower-triangular and upper-triangular part of A, respectively, Algorithm 3.1 will be expressed as

$$\begin{cases} x^{(k+1)} = (\dfrac{1}{\alpha'}D - L + \Omega)^{-1}\left(((\dfrac{1}{\alpha'} - 1)D + U + \Omega)x^{(k)} + B\left|x^{(k)}\right| - y^{(k)} + b\right), \\ y^{(k+1)} = \dfrac{1}{\alpha+1}(f(x^{(k+1)}) + \alpha y^{(k)}), \end{cases} \tag{3.5}$$

which can be called a new Newton-based SOR Method (NNSOR).

(5) When $M = \frac{1}{\alpha'}(D - \beta L), N = \frac{1}{\alpha'}((1-\alpha')D + (\alpha' - \beta)L + \alpha' U)$ where $D = \text{diag}(A)$, $-L, -U$ represent the strictly lower-triangular and upper-triangular part of A, respectively, Algorithm 3.1 will be expressed as

$$\begin{cases} x^{(k+1)} = (D - \beta L + \alpha'\Omega)^{-1}\Big(((1-\alpha')D + (\alpha' - \beta)L + \alpha' U + \alpha'\Omega)x^k \\ \qquad\qquad + \alpha'\left(B\left|x^k\right| - y^{(k)} + b\right)\Big), \\ y^{(k+1)} = \dfrac{1}{\alpha+1}(f(x^{(k+1)}) + \alpha y^{(k)}), \end{cases} \tag{3.6}$$

which can be called a new Newton-based AOR Method (NNAOR).

4. Global convergence

In this section, we turn to analyze the convergence properties of Algorithm 3.1 and the following lemmas are required.

Lemma 4.1. [20] *Let λ be any root of the quadratic equation $x^2 - bx + c = 0$ where $b, c \in \mathbb{R}$. Then $|\lambda| < 1$ if and only if $|c| < 1$ and $|b| < 1 + c$.*

Lemma 4.2. [6] *Let $x,\ y \in \mathbb{R}^n$, then $||x| - |y|| \leqslant |x - y|$.*

Let x^* is the solution of NGAVE (1.1), $y^* = f(x^*)$. The iteration errors $e_k^x = x^* - x^{(k)}, e_k^y = y^* - y^{(k)}$ where x^k, y^k is generated by Algorithm 3.1.

Theorem 4.1. *Let* $b \in \mathbb{R}^n, A \in \mathbb{R}^{n\times n}, M+\Omega \in \mathbb{R}^{n\times n}$ *is nonsingular and* $f : \mathbb{R}^n \mapsto \mathbb{R}^n$. *Denote* $\|(M+\Omega)^{-1}\| = \beta, \|N+\Omega\| = \gamma, \|B\| = \delta$. *Under the Assumption 2.1, if* $\beta(\gamma + L + \delta) < 1$, *then the Algorithm 3.1 is convergent.*

Proof. From (1.1), we have

$$Ax^* - B|x^*| + f(x^*) = b. \tag{4.1}$$

Then (4.1) is equivalent to

$$\begin{aligned} &(M+\Omega)x^* = (N+\Omega)x^* + B|x^*| - f(x^*) + b, \\ &i.e.\ x^* = (M+\Omega)^{-1}((N+\Omega)x^* + B|x^*| - y^* + b). \end{aligned} \tag{4.2}$$

And we know

$$\begin{aligned} &(\alpha+1)f(x^*) = f(x^*) + \alpha f(x^*), \\ &i.e.\ f(x^*) = \frac{1}{\alpha+1}(f(x^*) + \alpha y^*). \end{aligned} \tag{4.3}$$

Form (3.1), (4.2), (4.3), we can get

$$\begin{aligned} \|e^x_{k+1}\| &= \|(M+\Omega)^{-1}((N+\Omega)(x^* - x^k) + B(|x^*| - |x^k|) - (y^* - y^{(k)}))\| \\ &= \|(M+\Omega)^{-1}((N+\Omega)e^x_k + B(|x^*| - |x^k|) - e^y_k)\| \\ &\leqslant \|(M+\Omega)^{-1}\|(\|N+\Omega\|\|e^x_k\| + \|B\|\||x^*| - |x^k|\| + \|e^y_k\|) \\ &= \beta(\gamma+\delta)\|e^x_k\| + \beta\|e^y_k\|, \end{aligned} \tag{4.4}$$

and

$$\begin{aligned} \|e^y_{k+1}\| &= \|\frac{1}{\alpha+1}((f(x^*) - f(x^{(k+1)})) + \frac{\alpha}{\alpha+1}(y^* - y^{(k)}))\| \\ &\leqslant \frac{1}{\alpha+1}\|f(x^*) - f(x^{(k+1)})\| + \frac{\alpha}{\alpha+1}\|y^* - y^{(k)}\| \\ &\leqslant \frac{L}{\alpha+1}\|e^x_{k+1}\| + \frac{\alpha}{\alpha+1}\|e^y_k\| \\ &\leqslant \frac{L}{\alpha+1}(\beta(\gamma+\delta)\|e^x_k\| + \beta\|e^y_k\|) + \frac{\alpha}{\alpha+1}\|e^y_k\| \\ &= \frac{L\beta(\gamma+\delta)}{\alpha+1}\|e^x_k\| + \frac{L\beta+\alpha}{\alpha+1}\|e^y_k\|. \end{aligned} \tag{4.5}$$

Further,

$$\begin{aligned} \begin{pmatrix} \|e^x_{k+1}\| \\ \|e^y_{k+1}\| \end{pmatrix} &\leqslant \begin{pmatrix} \beta(\gamma+\delta) & \beta \\ \frac{L\beta(\gamma+\delta)}{\alpha+1} & \frac{L\beta+\alpha}{\alpha+1} \end{pmatrix} \begin{pmatrix} \|e^x_k\| \\ \|e^y_k\| \end{pmatrix} \\ &\leqslant \begin{pmatrix} \beta(\gamma+\delta) & \beta \\ \frac{L\beta(\gamma+\delta)}{\alpha+1} & \frac{L\beta+\alpha}{\alpha+1} \end{pmatrix}^2 \begin{pmatrix} \|e^x_{k-1}\| \\ \|e^y_{k-1}\| \end{pmatrix} \\ &\vdots \end{aligned}$$

$$\leqslant \begin{pmatrix} \beta(\gamma+\delta) & \beta \\ \dfrac{L\beta(\gamma+\delta)}{\alpha+1} & \dfrac{L\beta+\alpha}{\alpha+1} \end{pmatrix}^{k+1} \begin{pmatrix} \|e_0^x\| \\ \|e_0^y\| \end{pmatrix}. \tag{4.6}$$

Let $W = \begin{pmatrix} \beta(\gamma+\delta) & \beta \\ \dfrac{L\beta(\gamma+\delta)}{\alpha+1} & \dfrac{L\beta+\alpha}{\alpha+1} \end{pmatrix}$, we know that when $\rho(W) < 1$, $\lim\limits_{k\to\infty} W^k = 0$. It is shown that $\lim\limits_{k\to\infty} \|e_k^x\| = 0$, $\lim\limits_{k\to\infty} \|e_k^y\| = 0$. In other words, the Algorithm 3.1 converges to the unique solution x^*.

Next, we need to prove $\rho(W) < 1$ under Assumption 2.1.

Let λ be the eigenvalue of the matrix W. Then λ satisfies

$$\lambda^2 - (\beta(\gamma+\delta) + \frac{L\beta+\alpha}{\alpha+1})\lambda + (\frac{\beta(\gamma+\delta)(L\beta+\alpha)}{\alpha+1} - \frac{L\beta^2(\gamma+\delta)}{\alpha+1}) = 0.$$

After simple calculations, we have

$$\lambda^2 - (\beta(\gamma+\delta) + \frac{L\beta+\alpha}{\alpha+1})\lambda + (\frac{\beta\alpha(\gamma+\delta)}{\alpha+1}) = 0. \tag{4.7}$$

From $\beta(\gamma+L+\delta) < 1$, we can get $\beta(\gamma+\delta) < 1-\beta L$. Then $\alpha\beta(\gamma+\delta) < \alpha-\alpha\beta L$. Thus $\beta\alpha(\gamma+\delta)-1-\alpha < \alpha-\alpha\beta L-\alpha-1 = -\alpha\beta L-1 < 0$, i.e. $|\frac{\beta\alpha(\gamma+\delta)}{\alpha+1}| = \frac{\beta\alpha(\gamma+\delta)}{\alpha+1} < 1$. By direct calculations, we get $\beta(\gamma+\delta) + \frac{L\beta+\alpha}{\alpha+1} - \frac{\beta\alpha(\gamma+\delta)}{\alpha+1} = \frac{\beta(\gamma+\delta+L)+\alpha}{\alpha+1} < 1$. Hence $|\beta(\gamma+\delta) + \frac{L\beta+\alpha}{\alpha+1}| = \beta(\gamma+\delta) + \frac{L\beta+\alpha}{\alpha+1} < 1 + \frac{\beta\alpha(\gamma+\delta)}{\alpha+1}$. According to Lemma 4.1, $\rho(W) < 1$. This completes the proof. □

Corollary 4.1. *Set $\Omega = \varepsilon_0 I$. Assume that M, N are symmetric positive definite matrices. Under the Assumption 2.1, if $\lambda_{\max}(N) + L + \delta < \lambda_{\min}(M)$ where $\lambda_{\min}(M), \lambda_{\max}(N)$ are the smallest eigenvalue of matrix M and the largest eigenvalue of matrix N, respectively, then the Algorithm 3.1 is convergent.*

Proof. By simple calculations, we obtain

$$\begin{aligned} \beta(\gamma+L+\delta) &= \|(M+\varepsilon_0 I)^{-1}\|(\|N+\varepsilon_0 I\| + L + \delta) \\ &= \frac{\lambda_{\max}(N) + \varepsilon_0 + L + \delta}{\lambda_{\min}(M) + \varepsilon_0}. \end{aligned} \tag{4.8}$$

If $\lambda_{max}(N) + L + \delta < \lambda_{min}(M)$, then $\beta(\gamma+L+\delta) < 1$. Thus the Algorithm 3.1 is convergent. □

Corollary 4.2. *Under the Assumption 2.1, if $\|M\| < \frac{1}{\|\Omega\|} - (\gamma+L+\delta)$, then the Algorithm 3.1 is convergent.*

Proof. Set $\widehat{A} = \Omega, \widehat{B} = M + \Omega$. Then $\|\widehat{A} - \widehat{B}\| = \|M\| \leqslant \frac{1}{\|\Omega\|} - (\gamma+L+\delta)$. Further, $\|\Omega\|(\frac{1}{\|\Omega\|} - (\gamma+L+\delta)) = 1 - \|\Omega\|(\gamma+L+\delta) < 1$. Based on the perturbation lemma in [15], we get

$$\begin{aligned} \beta &= \|(M+\Omega)^{-1}\| \\ &\leqslant \frac{\|\Omega\|}{1 - \|\Omega\|\|M\|} \end{aligned}$$

$$< \frac{\|\Omega\|}{1-\|\Omega\|(\frac{1}{\|\Omega\|}-(\gamma+L+\delta))}$$
$$= \frac{1}{\gamma+L+\delta}. \tag{4.9}$$

Therefore, $\beta(\gamma+L+\delta)<1$. From Theorem 2.1, Algorithm 3.1 is convergent. □

Lemma 4.3. [21] *Let $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times n}$. If $|A| \leqslant B$, then $\rho(A) \leqslant \rho(B)$.*

Proposition 4.1. *Let $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times n}$. If $|A| \leqslant B$, then $\|A\| \leqslant \|B\|$.*

Proof. From $|A| \leqslant B$, it is following that $|A^TA| \leqslant |A^T||A| \leqslant B^TB$. By the Lemma 4.3, $\rho(A^TA) \leqslant \rho(B^TB)$. Therefore, $\|A\| \leqslant \|B\|$. □

Next, in order to obtain Theorems 4.2 and 4.3, we introduce the definition of H-compatible splitting and some useful lemmas.

Definition 4.1. Let $A = M - N$ with $\det(M) \neq 0$. Then it is named an H-compatible splitting if $\langle A\rangle = \langle M\rangle - |N|$ is a nonsingular M-matrix with $|N| = (|n_{ij}|)$.

Lemma 4.4. [16] *If $A \leqslant B$, where A, B are M-matrix and Z-matrix, respectively, then B is M-matrix.*

Lemma 4.5. [9] *Let $A \in \mathbb{R}^{n\times n}$, then the following properties hold:*

(1) If A is a H-matrix, then $|A^{-1}| \leqslant \langle A\rangle^{-1}$;

(2) Strictly diagonally dominant or irreducible diagonally dominant matrix is H-matrix.

Lemma 4.6. [9] *Let $A, B \in \mathbb{R}^{n\times n}$ are nonsingular M-matrices, and $A \leqslant B$, then $A^{-1} \geqslant B^{-1}$.*

Theorem 4.2. *Let $A = M - N$ is an H-compatible splitting of H_+-matrix A and Ω is a positive diagonal matrix. When $\|(\Omega + \langle M\rangle)^{-1}\| < \frac{1}{\gamma+L+\delta}$, Algorithm 3.1 is convergent.*

Proof. Since $A = M - N$ is an H-compatible splitting of matrix A, we obtain $\langle A\rangle \leqslant \langle M\rangle$. Because A is an H-matrix, $\langle A\rangle$ is a M-matrix. This shows that $\langle M\rangle$ is M-matrix according to Lemma 4.4.

Obviously, $\langle M + \Omega\rangle$ is a nonsingular M-matrix, so $M + \Omega$ is an H-matrix. Based on Lemma 4.5, we can get $|(M + \Omega)^{-1}| \leqslant \langle M + \Omega\rangle^{-1} = (\langle M\rangle + \Omega)^{-1}$. According to Proposition 4.1, $\|(M + \Omega)^{-1}\| \leqslant \|(\langle M\rangle + \Omega)^{-1}\| \leqslant \frac{1}{\gamma+L+\delta}$. Therefore, Algorithm 3.1 is convergent. □

Theorem 4.3. *Let $A = M - N$ is an H-compatible splitting of H-matrix A, M is an H_+-matrix and Ω is a strictly diagonally dominant H_+-matrix. When $\|(\langle\Omega\rangle + \langle M\rangle)^{-1}\| < \frac{1}{\gamma+L+\delta}$, Algorithm 3.1 is convergent.*

Proof. Since $A = M - N$ is an H-compatible splitting of matrix A, it is hold that $\langle A\rangle = \langle M\rangle - |N|$. And M is an H_+-matrix, therefore, $\langle A\rangle \leqslant \langle M\rangle \leqslant diag(M)$. We know that A is an H-matrix, then $\langle A\rangle$ is a M-matrix. From Lemma 4.4, $\langle M\rangle$ is a M-matrix.

According to Assumption, Ω is a strictly diagonally dominant H_+-matrix, then $\langle\Omega\rangle$ is a M-matrix. Therefore, $\langle\Omega\rangle + \langle M\rangle$ is a M-matrix.

$\langle M+\Omega\rangle \geqslant \langle M\rangle + \langle\Omega\rangle$ holds where off-diagonal part is based on inequality $-|m_{ij}+\omega_{ij}| \geqslant -|m_{ij}|-|\omega_{ij}|$ and diagonal part holds naturally. Then $\langle M+\Omega\rangle$ is a M-matrix, i.e., $M+\Omega$ is H_+-matrix. Further, we have $|(M+\Omega)^{-1}| \leqslant \langle M+\Omega\rangle^{-1} \leqslant (\langle M\rangle + \langle\Omega\rangle)^{-1}$, where the second inequality follows from lemma 4.6. From Proposition 4.1, $\|(M+\Omega)^{-1}\| \leqslant \|(\langle M\rangle + \langle\Omega\rangle)^{-1}\| \leqslant \frac{1}{\gamma+L+\delta}$. Therefore, Algorithm 3.1 is convergent. □

5. Numerical results

In this section, we use the following two examples to verify the feasibility of the algorithm we proposed. And we intuitively analyze the effect of the algorithm from the iteration count (indicated as 'IT'), the relative residual error (indicated as 'RES') and the elapsed CPU time (indicated as 'CPU') where RES is defined as RES $= \frac{\|Ax^k-B|x^k|+f(x^k)-b\|}{\|b\|}$. While RES $< 10^{-4}$ or the prescribed iteration count $k_{max} = 500$ is surpassed, all iterations are terminated. The programming language used was MATLAB R2018a.

Example 5.1. Let

$$S = \begin{pmatrix} 4 & -0.5 & 0 & 0 & 0 \\ -1.5 & 4 & -0.5 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & -1.5 & 4 & -0.5 \\ 0 & 0 & 0 & -1.5 & 4 \end{pmatrix} \in \mathbb{R}^{m\times m},$$

$$M_1 = \begin{pmatrix} S & -0.5I_0 & 0 & 0 & 0 \\ -1.5I_0 & S & -0.5I_0 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & -1.5I_0 & S & -0.5I_0 \\ 0 & 0 & 0 & -1.5I_0 & S \end{pmatrix} \in \mathbb{R}^{n\times n},$$

where $I_0 \in \mathbb{R}^{m\times m}$ is an identity matrix. Consider the NGAVE $Ax - B|x| + f(x) = b$ where $A = M + I,\ B = M - I,\ b = Ax^* - B|x^*| + f(x^*),\ f(\cdot) = sin(\cdot)$. Here, $x^* = (-0.6, -0.6, \cdots, -0.6)^T$ and $M = M_1 + \mu I$. $I \in \mathbb{R}^{n\times n}$ is an identity matrices with n dimensions, $n = m^2$.

For Example 5.1, to improve the convergence speed of all the tested methods, the choice of Ω is $\Omega = 1.2M_1$. we take the parameter $\mu = 4, \alpha = 0.5$. The initial iteration points $x^{(0)}, y^{(0)}$ are

$$x^{(0)} = y^{(0)} = (1, 0, 1, 0, \cdots, 1, 0)^T \in \mathbb{R}^n.$$

In the implementation of the algorithm, inverse matrices of $D - L + \Omega,\ \frac{1}{\alpha'}D - L + \Omega,\ \frac{1}{\alpha'}(D - \beta L) + \Omega,$ can be determined by the sparse LU factorization or the sparse Cholesky factorization. And according to Figure 1, we choose the parameter value to be $\alpha' = 0.9, \beta = 0.6$.

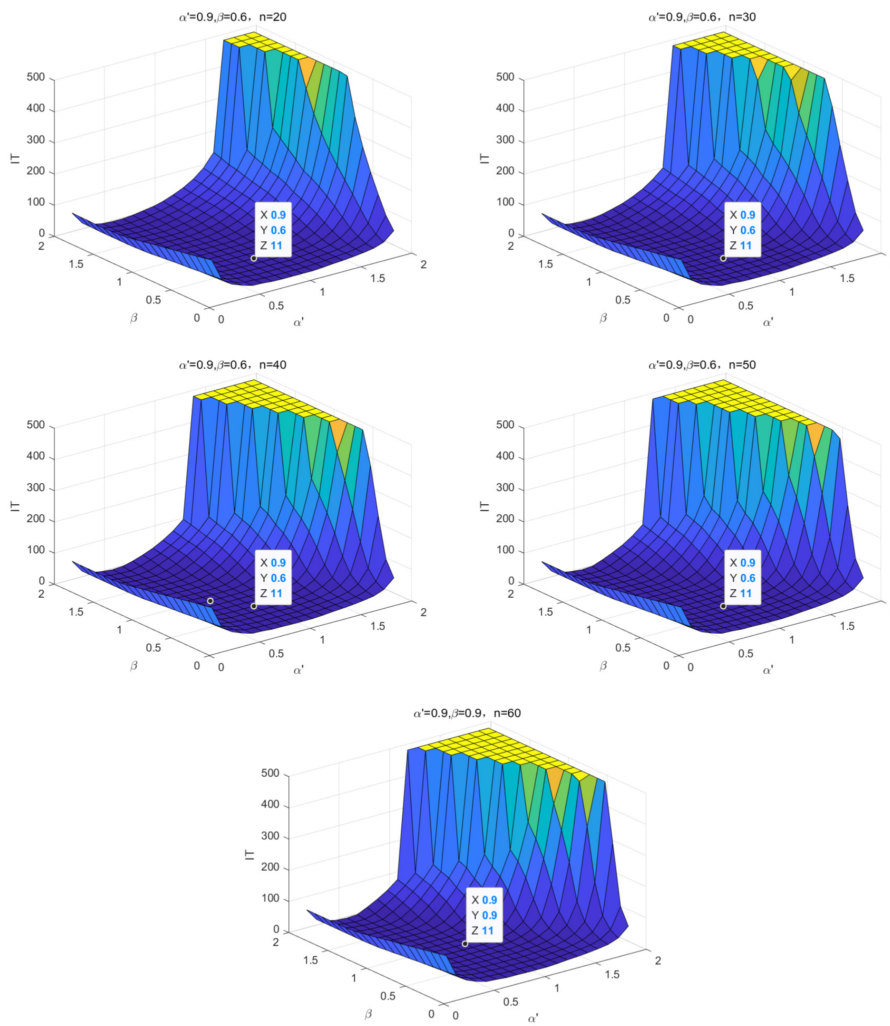

Figure 1. Selection of optimal parameters $\alpha^{'}, \beta$ of the NNAOR method for Example 5.1.

Table 1. Numerical comparisons about the mentioned algorithms for Example 5.1.

Algorithm	n	400	900	1600	2500	3600
	IT	14	14	14	13	13
$GUASS-NEWTON$	RES	9.1499e-07	6.2024e-07	4.6938e-07	8.3176e-07	6.9674e-07
	CPU	0.207728	1.122682	5.436864	15.670527	40.725230
	IT	11	11	11	11	11
NNJ	RES	4.7816e-07	4.7522e-07	4.7350e-07	4.7238e-07	4.7160e-07
	CPU	0.029603	0.108098	0.379684	1.007731	2.621176
	IT	12	12	12	12	12
$NNGS$	RES	8.4979e-07	8.6476e-07	8.5153e-07	8.3644e-07	8.2313e-07
	CPU	0.031724	0.113848	0.391762	0.984423	2.536378
	IT	11	11	11	11	11
$NNSOR$	RES	4.1578e-07	4.1491e-07	4.1450e-07	4.1426e-07	4.1411e-07
$\alpha' = 0.9$	CPU	0.029993	0.105045	0.346909	1.085007	2.780677
	IT	11	11	11	11	11
$NNAOR$	RES	4.5987e-07	4.5975e-07	4.5968e-07	4.5964e-07	4.5961e-07
$(\alpha' = 0.9; \beta = 0.6)$	CPU	0.031198	0.114716	0.421824	1.064167	2.751035

Table 2. Numerical comparisons about the mentioned algorithms for Example 5.2 (q=1, p=2).

Algorithm	n	400	900	1600	2500	3600
	IT	4	4	5	4	4
$GUASS-NEWTON$	RES	3.4774e-05	5.4918e-05	4.0280e-07	8.4068e-05	9.3288e-05
	CPU	0.104606	0.349643	1.872097	4.493567	12.157909
	IT	5	5	5	5	5
NNJ	RES	5.9439e-05	5.1951e-05	5.2414e-05	5.2997e-05	5.1496e-05
	CPU	0.023763	0.061204	0.173787	0.431277	1.222827
	IT	4	4	5	4	4
$NNGS$	RES	3.8730e-05	2.5590e-05	2.2263e-05	2.1313e-05	1.7856e-05
	CPU	0.025634	0.060369	0.151556	0.467179	1.055523
	IT	3	3	3	3	3
$NNSOR$	RES	3.8850e-05	2.3745e-05	1.7794e-05	1.4794e-05	1.2391e-05
$\alpha' = 1.1$	CPU	0.021410	0.059269	0.162472	0.461926	1.020736
	IT	3	3	3	2	2
$NNAOR$	RES	3.5429e-05	2.0486e-05	1.5043e-05	9.4872e-05	8.4616e-05
$(\alpha' = 1.1; \beta = 0.9)$	CPU	0.024318	0.060364	0.147851	0.412287	1.147714

According to the numerical results given in Table 1 and Figure 2, the Guass-Newton method and the four methods presented in this paper can converge to the solution x^* quickly for different problem sizes. Moreover, the performance of the five methods in Example 5.1 is relatively stable. It can be seen intuitively from Table 1 that CPU time of the NNJ method, the NNGS method, the NNSOR method and the NNAOR method is obviously less than the Guass-Newton method. Because the Guass-Newton method needs to calculate the Jacobian matrix of $F(x^{(k)})$ at each step, it brings a huge amount of computation. However, the method proposed in this paper only needs to calculate the inverse of $M + \Omega$ once and correct the nonlinear part $f(x^{(k)})$ in each update, thus the calculation time of the NNJ method, the NNGS method, the NNSOR method and the NNAOR method are much less than that of Guass-Newton method.

Example 5.2. We consider the NGAVE (1.1) such that

$$A = T_x \otimes I_m + I_m \otimes T_y + pI_n,\ B = I_n, \quad \left(n = m^2\right) \tag{5.1}$$

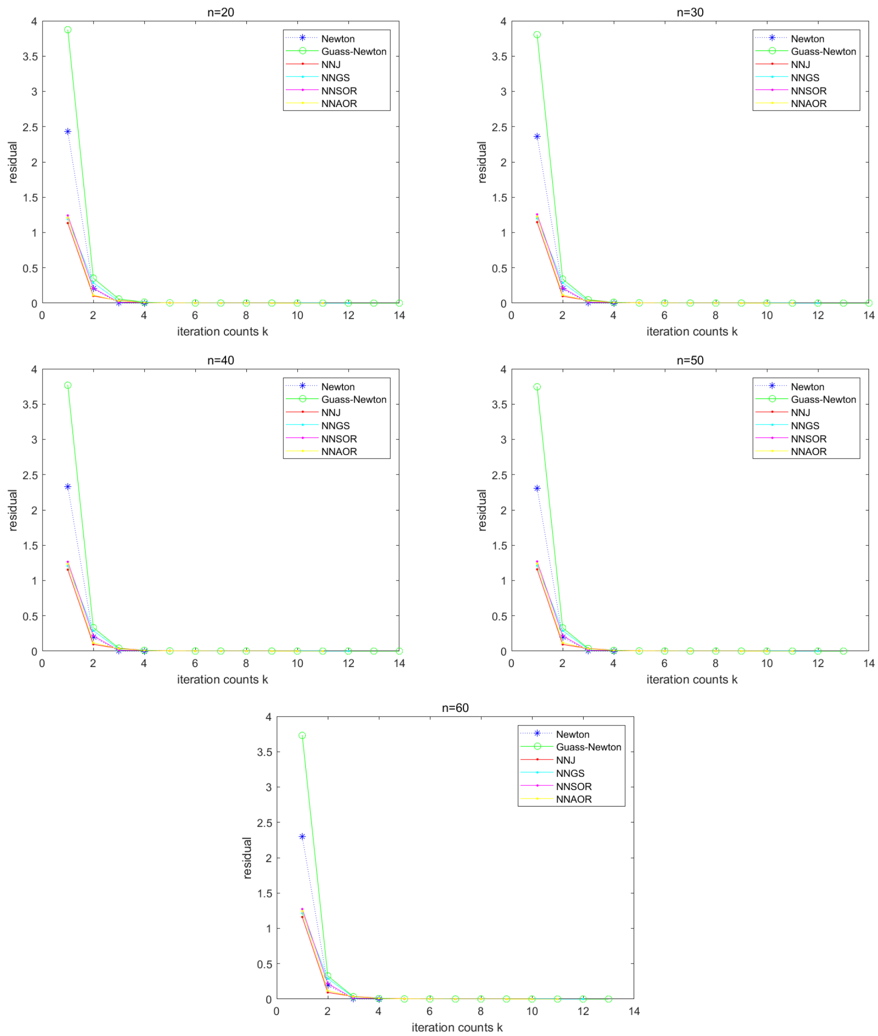

Figure 2. Convergence effect for Example 5.1.

Table 3. Numerical comparisons about the mentioned algorithms for Example 5.2 (q=2, p=4).

Algorithm	n	400	900	1600	2500	3600
	IT	4	4	4	4	4
$GUASS-NEWTON$	RES	2.8344e-05	3.3656e-05	6.7496e-05	6.2878e-05	6.7983e-05
	CPU	0.061057	0.308713	1.111980	3.351973	8.938422
	IT	5	5	5	5	5
NNJ	RES	3.8659e-05	3.2595e-05	3.1079e-05	3.1558e-05	3.0674e-05
	CPU	0.022322	0.059627	0.151547	0.465858	1.131326
	IT	4	3	3	3	3
$NNGS$	RES	3.0823e-05	8.8911e-05	8.9703e-05	9.0955e-05	9.1514e-05
	CPU	0.023348	0.063185	0.149682	0.464332	0.998092
	IT	3	3	3	3	3
$NNSOR$	RES	3.2762e-05	2.0298e-05	1.6252e-05	1.3458e-05	1.1597e-05
$\alpha^{'}=1.1$	CPU	0.032953	0.058465	0.144290	0.463707	1.075843
	IT	3	3	3	2	2
$NNAOR$	RES	3.1594e-05	1.7208e-05	1.2959e-05	8.8250e-05	7.8590e-05
$(\alpha^{'}=1.1;\beta=0.9)$	CPU	0.026301	0.058266	0.170766	0.459618	1.065003

where $T_x = \text{tridiag}(-1-r, 4, -1+r), T_y = \text{tridiag}(-1-r, 0, -1+r), r = (qh)/2$ and $h = 1/(m+1)$ for given real number p and nonnegative constant q. The right-hand side of NGAVE (1.1) is constructed such that $x^* = (1, 2, \cdots, n)^T$ satisfies $Ax^* - |x^*| + f(x^*) = b$ where $f(\cdot) = cos(\cdot)$.

For Example 5.2, the choice of Ω is $\Omega = \delta In, \delta = -0.9$, and $\alpha = 0.5$. We take the parameter $q = 1, p = 2$ and $q = 2, p = 4$, respectively. In this experiment, the initial iteration points $x^{(0)}, y^{(0)}$ are

$$x^{(0)} = y^{(0)} = (0, 0, \cdots, 0, 0)^T \in \mathbb{R}^n. \tag{5.2}$$

And according to Figure 3 and 5, we choose the parameter value to be $\alpha^{'} = 1.1, \beta = 0.9$.

It can be seen from Figure 4 and 6 that all the six methods can effectively and steadily converge to the solution x^* when $q = 1, p = 2$ or $q = 2, p = 4$. The numerical results in Tables 2 and 3 show that the Guass-Newton method and the NNJ method, the NNGS method have little difference in iteration counts and residuals in both cases ($q = 1, p = 2$ and $q = 2, p = 4$). However, the CPU time of the NNJ method, the NNGS method is significantly less than that of the Guass-Newton method. When setting the appropriate values for α' and β, the NNSOR method and the NNAOR method also performed better than the Newton method and the Guass-Newton method in terms of CPU time.

Examples 5.1 and 5.2 show that the new Newton-based matrix splitting iterative method is effective and the convergence effect is superior to the Guass-Newton method. Through direct calculation, the NNJ method, the NNGS method, the NNSOR method and the NNAOR method in Example 5.2 meet the condition proposed in Theorem 4.1 but the condition do not be satisfied in Example 5.1, indicating that the conditions proposed in Theorem 4.1 are sufficient conditions rather than necessary conditions for the new Newton-based matrix splitting iterative method to converge.

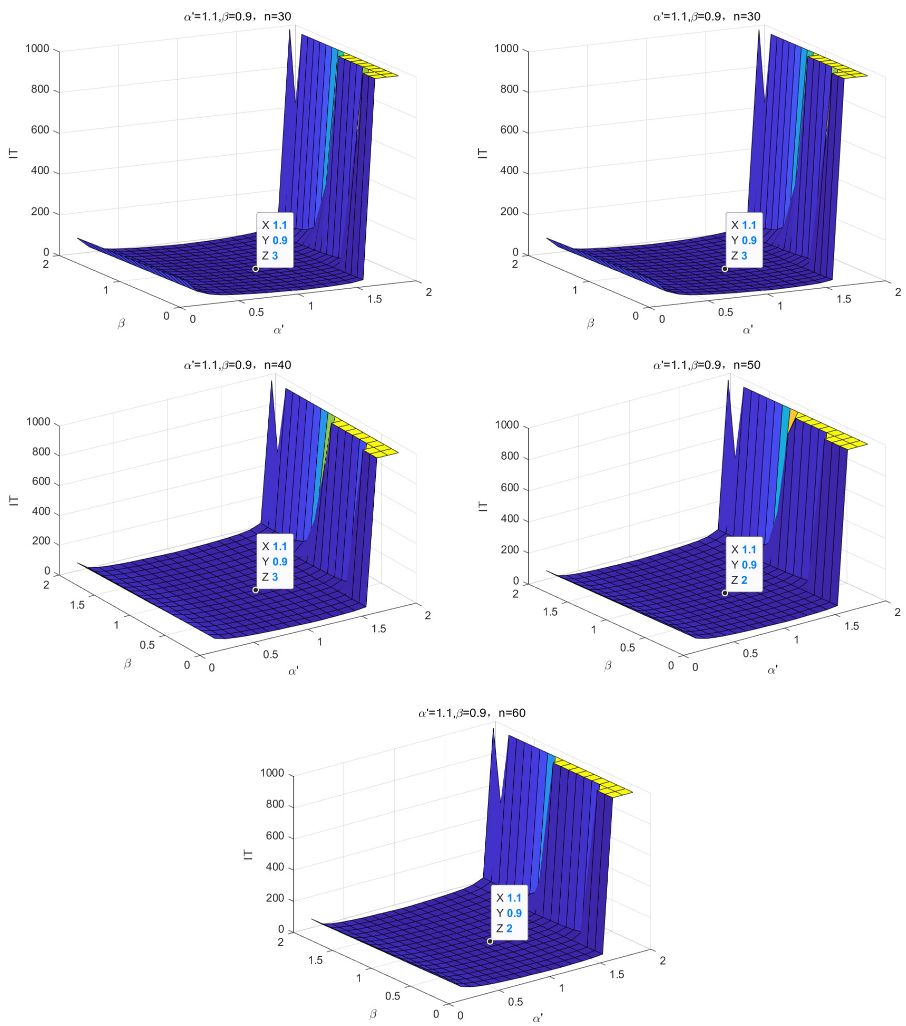

Figure 3. Selection of optimal parameters $\alpha^{'}, \beta$ of the NNAOR method for Example 5.2 (q=1;p=2).

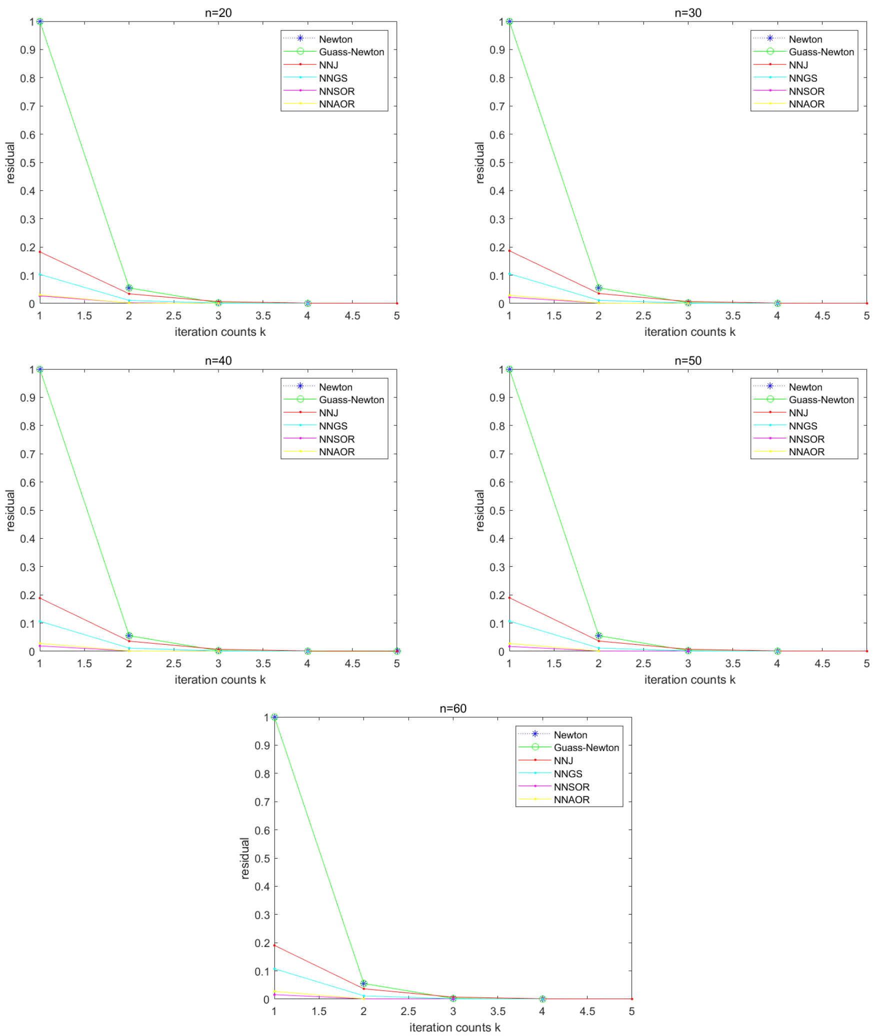

Figure 4. Convergence effect for Example 5.2 (q=1;p=2).

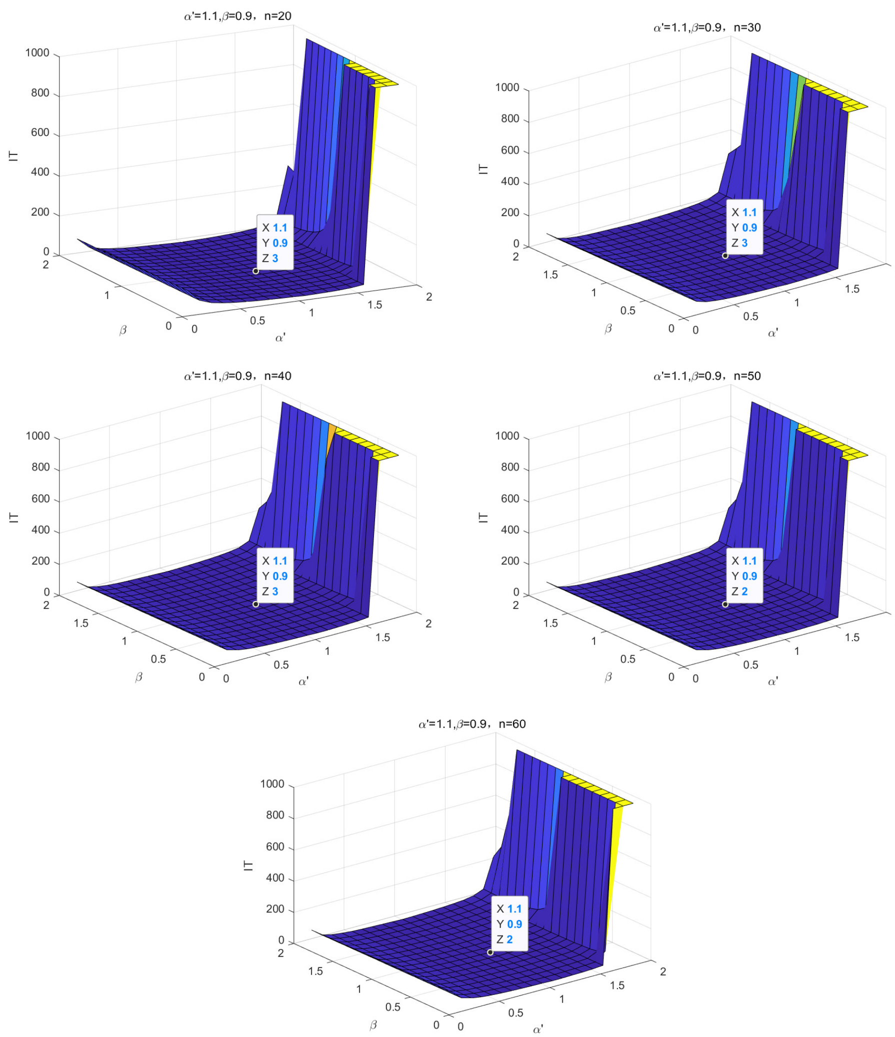

Figure 5. Selection of optimal parameters $\alpha^{'}, \beta$ of the NNAOR method for Example 5.2 (q=2;p=4).

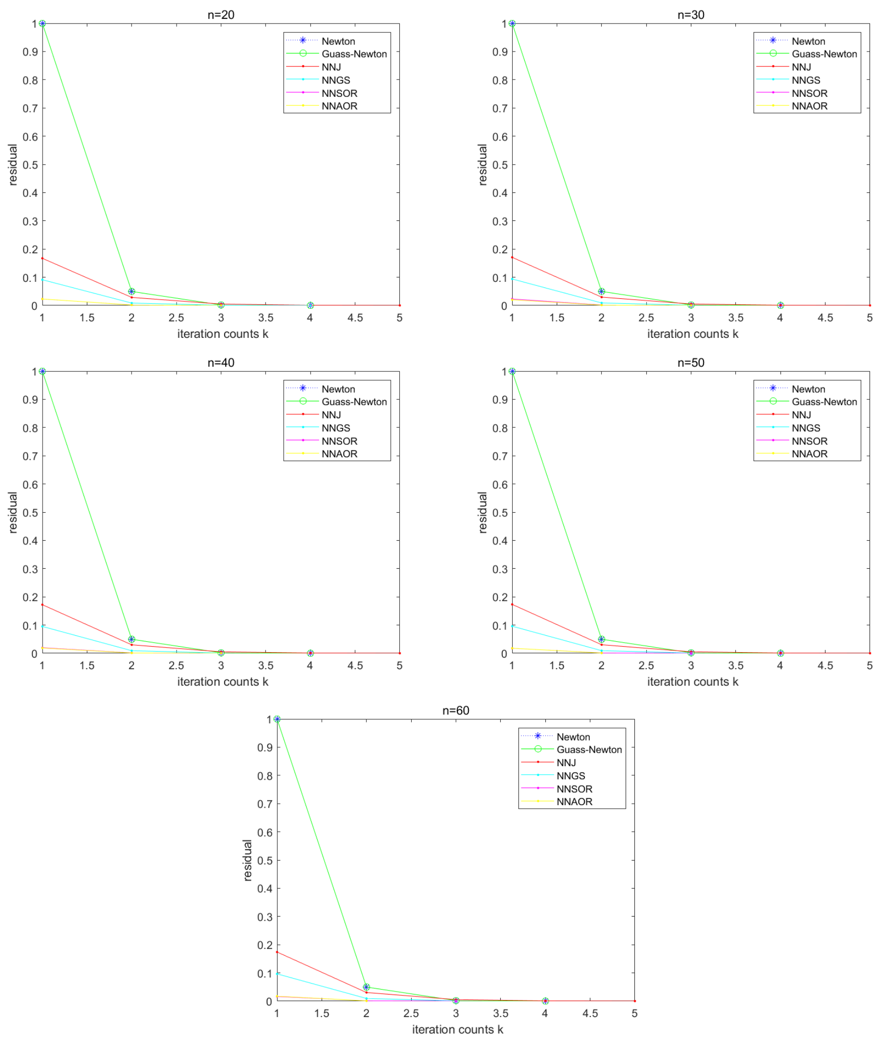

Figure 6. Convergence effect for Example 5.2 (q=2;p=4).

6. Conclusions

In this paper, we come up with the generalized absolute value equation with nonlinear term and a new Newton-based matrix splitting iterative method is proposed for solving generalized absolute value equation with nonlinear term. We give the global convergence of this method and show that some new convergence conditions are proposed for certain splitting or properties of matrix A. Numerical results indicate that the new Newton-based matrix splitting iterative method for solving generalized absolute value equation with nonlinear term is effective. However, as there are three parameters in the NNAOR method, it is very difficult to determine the optimal value for these parameters and it needs further study.

Declarations

Data availability statement. Data will be made available on reasonable request.

Conflict of interest. The author declares that they have no conflict of interest.

Ethical approval. This manuscript does not contain any studies with human participants or animals performed by any of the authors.

References

[1] L. Abdallah, M. Haddou and T. Migot, *Solving absolute value equation using complementarity and smoothing functions*, J. Comput. Appl. Math., 2018, 327, 196–207.

[2] Z. Z. Bai, *On the convergence of the multisplitting methods for the linear complementarity problem*, SIAM J. Matrix Anal. Appl., 1999, 21, 67–78.

[3] R. W. Cottle, J. S. Pang and R. E. Stone, *The Linear Complementarity Problem*, Academic, San Diego, 1992.

[4] B. H. Huang and C. F. Ma, *Convergent conditions of the generalized Newton method for absolute value equation over second order cones*, Appl. Math. Lett., 2019, 92, 1–6.

[5] B. H. Huang and C. F. Ma, *The modulus-based Levenberg-Marquardt method for solving linear complementarity problem*, Numer. Math., 2019, 12, 154–168.

[6] Y. F. Ke, *The new iteration algorithm for absolute value equation*, Appl. Math. Lett., 2020, 99, 105990.

[7] Y. F. Ke, *Iterative Solution of Nonlinear Equations*, Electronic Industry Press, Beijing, 2021.

[8] C. F. Ma, L. H. Jiang and D. S. Wang, *The convergence of a smoothing damped Gauss-Newton method for nonlinear complementarity problem*, Nonlin. Anal. Real World Appl., 2009, 10, 2072–2087.

[9] C. F. Ma, Y. F. Ke, J. Tang and B. G. Chen, *Numerical Linear Algebra and Algorithms*, National Defence Industry Press, Beijing, 2017.

[10] O. L. Mangasarian, *A generalized Newton method for absolute value equations*, Optim. Lett., 2009, 3, 101–108.

[11] O. L. Mangasarian, *Absolute value programming*, Comput. Optim. Appl., 2007, 36, 43–53.

[12] O. L. Mangasarian and R. R. Meyer, *Absolute value equations*, Linear Algebra Appl., 2006, 419, 359–367.

[13] A. Mansoori and M. Erfanian, *A dynamic model to solve the absolute value equations*, J. Comput. Appl. Math., 2018, 333, 28–35.

[14] C. T. Nguyena, B. Saheyab, Y. L. Chang and J. S. Chen, *Unified smoothing functions for absolute value equation associated with second-order cone*, Appl. Numer. Math., 2019, 135, 206–227.

[15] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Sevberal Variables*, Academic Press, 1970.

[16] J. Rohn, *Systems of linear interval equations*, Linear Alge. Appl., 1989, 126, 39–78.

[17] J. Rohn, *A theorem of the alternatives for the equation* $Ax + B|x| = b$, Linear Multilinear Algebra, 2004, 52, 421–426.

[18] J. Rohn, V. Hooshyarbakhsh and R. Farhadsefat, *An iterative method for solving absolute value equations and sufficient conditions for unique solvability*, Optim. Lett., 2014, 8, 35–44.

[19] A. Wang, Y. Cao and J. X. Chen, *Modified Newton-type iteration methods for generalized absolute value equations*, J. Optim. Theo. Appl., 2019, 181, 216–230.

[20] S. L. Wu, T. Z. Huang and X. L. Zhao, *A modified SSOR iterative method for augmented systems*, Comput. Appl. Math., 2009, 228, 424–433.

[21] S. F. Xu, *Matrix Calculation in Cybernetics*, Higher Education Press, Beijing, 2011.

[22] C. C. Zhou, Y. Cao, Q. Q. Shen and Q. Shi, *A modified Newton-based matrix splitting iteration method for generalized absolute value equations*, J. Comput. Appl. Math., 2024, 442, 115747.

[23] H. Y. Zhou, S. L. Wu and C. X. Li, *Newton-basedd matrix splitting method for generalized absolute value equation*, J. Comput. Appl. Math., 2021, 394, 360–375.