

A NON-MONOTONE SMOOTHING NEWTON ALGORITHM FOR SOLVING THE SYSTEM OF GENERALIZED ABSOLUTE VALUE EQUATIONS*

Cairong Chen

*School of Mathematics and Statistics, Key Laboratory of Analytical Mathematics and Applications
(Ministry of Education), Fujian Provincial Key Laboratory of Statistics and Artificial Intelligence,
Fujian Normal University, Fuzhou 350117, P.R. China*

Email: cairongchen@fjnu.edu.cn

Dongmei Yu¹⁾

*Institute for Optimization and Decision Analytics, College of Science, Liaoning Technical University,
Fuxin 123000, P.R. China*

Email: yudongmei1113@163.com

Deren Han

*LMIB of the Ministry of Education, School of Mathematical Sciences, Beihang University,
Beijing 100191, P.R. China*

Email: handr@buaa.edu.cn

Changfeng Ma

*School of Mathematics and Statistics, Key Laboratory of Analytical Mathematics and Applications
(Ministry of Education), Fujian Normal University, Fuzhou 350117, P.R. China*

Email: macf@fjnu.edu.cn

Abstract

The system of generalized absolute value equations (GAVE) has attracted more and more attention in the optimization community. In this paper, by introducing a smoothing function, we develop a smoothing Newton algorithm with non-monotone line search to solve the GAVE. We show that the non-monotone algorithm is globally and locally quadratically convergent under a weaker assumption than those given in most existing algorithms for solving the GAVE. Numerical results are given to demonstrate the viability and efficiency of the approach.

Mathematics subject classification: 65F10, 65H10, 90C30.

Key words: Generalized absolute value equations, Smoothing function, Smoothing Newton algorithm, Non-monotone line search, Global and local quadratic convergence.

1. Introduction

The system of generalized absolute value equations (GAVE) is to find a vector $x \in \mathbb{R}^n$ such that

$$Ax + B|x| - b = 0, \quad (1.1)$$

where $A \in \mathbb{R}^{n \times n}$ and $0 \neq B \in \mathbb{R}^{n \times n}$ are two known matrices, $b \in \mathbb{R}^n$ is a known vector, and $|x|$ denotes the componentwise absolute value of $x \in \mathbb{R}^n$. In the literature, GAVE also occurs in the form of $Ax - B|x| = b$. In this paper, we do not make a distinction between it and (1.1) and

* Received April 8, 2022 / Revised version received August 1, 2022 / Accepted November 28, 2022 /

Published online February 26, 2024 /

¹⁾ Corresponding author

put it down to GAVE (1.1). To the best of our knowledge, GAVE (1.1) was first introduced by Rohn in [34] and further investigated in [12, 15, 21, 27, 29, 30] and references therein. Obviously, when $B = -I$ with I being the identity matrix, GAVE (1.1) becomes the system of absolute value equations (AVE)

$$Ax - |x| - b = 0, \quad (1.2)$$

which is the subject of numerous research works, see, e.g. [4, 11, 22, 25, 48, 49] and references therein.

GAVE (1.1) and AVE (1.2) have attracted considerable attention in the field of optimization for almost twenty years, and the primary reason is that they are closely related to the linear complementarity problem (LCP) [25, 30] and the horizontal LCP (HLCP) [27], which encompass many mathematical programming problems and have many practical applications [7, 28]. In addition, GAVE (1.1) and AVE (1.2) are also bound up with the system of linear interval equations [33].

Due to the combinatorial character introduced by the absolute value operator, solving GAVE (1.1) is generally NP-hard [21, Proposition 2]. Moreover, if GAVE (1.1) is solvable, checking whether it has a unique solution or multiple solutions is NP-complete [30, Proposition 2.1]. Recently, GAVE (1.1) and AVE (1.2) have been extensively investigated in the literature, and the main research effort can be summarized to the following two aspects.

On the theoretical side, one of the main branches is to investigate conditions for existence, non-existence and uniqueness of solutions of GAVE (1.1) or AVE (1.2), see, e.g. [12, 13, 25, 27, 30, 34, 43–45] and references therein. Specially, the following necessary and sufficient conditions that ensure the existence and uniqueness of solution of GAVE (1.1) can be found in [27, 45] (see Section 2 for the definition of the column \mathcal{W} -property).

Theorem 1.1 ([27, Theorem 1]). *The following statements are equivalent:*

- (i) *GAVE (1.1) has a unique solution for any $b \in \mathbb{R}^n$.*
- (ii) *$\{A + B, A - B\}$ has the column \mathcal{W} -property.*
- (iii) *For arbitrary nonnegative diagonal matrices $D_1, D_2 \in \mathbb{R}^{n \times n}$ with $D_1 + D_2 > 0$,*

$$\det[(A + B)D_1 + (A - B)D_2] \neq 0.$$

- (iv) *$A + B$ is nonsingular and $\{I, (A + B)^{-1}(A - B)\}$ has the column \mathcal{W} -property.*

Theorem 1.2 ([45, Theorem 3.2]). *GAVE (1.1) has a unique solution for any $b \in \mathbb{R}^n$ if and only if matrix $A + BD$ is nonsingular for any diagonal matrix $D = \text{diag}(d_i)$ with $d_i \in [-1, 1]$.*

It is easy to conclude that Theorems 1.1 and 1.2 imply that $\{A + B, A - B\}$ has the column \mathcal{W} -property if and only if matrix $A + BD$ is nonsingular for any diagonal matrix $D = \text{diag}(d_i)$ with $d_i \in [-1, 1]$ (see Lemma 2.3 for more details). Throughout the theoretical analysis of this paper, we let the following assumption hold.

Assumption 1.1. *Let matrices A and B satisfy $\{A + B, A - B\}$ has the column \mathcal{W} -property.*

On the numerical side, there are various algorithms for solving AVE (1.2) or GAVE (1.1). For example, Mangasarian proposed concave minimization method [22], generalized Newton method [23], and successive linear programming method [24] for solving AVE (1.2). Zamani

and Hladík [48] proposed a new concave minimization algorithm for AVE (1.2), which solves a deficiency of the method proposed in [22]. Zainali and Lotfi [47] modified the generalized Newton method and developed a stable and quadratic convergent method for AVE (1.2). Cruz *et al.* [3] proposed an inexact semi-smooth Newton method for AVE (1.2). Shahsavari and Ketabchi [36] proposed two types of proximal algorithms to solve AVE (1.2). Haghani [11] introduced generalized Traub's method for AVE (1.2). Ke and Ma [19] proposed an SOR-like iteration method for AVE (1.2). Caccetta *et al.* [4] proposed a smoothing Newton method for AVE (1.2). Saheya *et al.* [35] summarized several systematic ways of constructing smoothing functions and proposed a unified neural network model for solving AVE (1.2). Zhang and Wei [49] proposed a generalized Newton method which combines the semismooth and the smoothing Newton steps for AVE (1.2). Lian *et al.* [20] further considered the generalized Newton method for GAVE (1.1) and presented some weaker convergent conditions compared to the results in [15, 23]. Wang *et al.* [42] proposed modified Newton-type iteration methods for GAVE (1.1). Zhou *et al.* [51] established Newton-based matrix splitting methods for GAVE (1.1). Jiang and Zhang [17] proposed a smoothing-type algorithm for GAVE (1.1). Tang and Zhou [40] proposed a quadratically convergent descent method for GAVE (1.1). Hu *et al.* [15] proposed a generalized Newton method for absolute value equations associated with second order cones (SOCAVE), which is an extension of GAVE (1.1). For more numerical algorithms, one can refer to [1, 5, 8, 10, 18, 26, 46] and references therein.

By looking into the mathematical format of GAVE (1.1), non-differentiability is caused by the absolute value operator. Smoothing algorithms have been successfully applied to solve GAVE (1.1) [17, 40]. However, monotone line search techniques were used in the methods proposed in [17, 40]. Recently, great attention has been paid to smoothing algorithms with non-monotone line search, see, e.g. [16, 38, 39, 52] and references therein. Non-monotone line search schemes can improve the likelihood of finding a global optimum and improve convergence speed in cases where a monotone line search scheme is forced to creep along the bottom of a narrow curved valley [50]. It is therefore interesting to develop non-monotone smoothing algorithms for solving GAVE (1.1). This motivates us to develop a non-monotone smoothing Newton algorithm for solving GAVE (1.1). Our work here is inspired by recent studies on weighted complementarity problem [39, 41].

The rest of this paper is organized as follows. In Section 2, we provide some concepts and results used throughout the paper. In Section 3, we develop a non-monotone smoothing Newton algorithm for solving GAVE (1.1), while Section 4 is devoted to discussing the convergence. Numerical experiments are given in Section 5. Finally, Section 6 concludes this paper.

Notation. $\mathbb{R}^{m \times m}$ is the set of all $m \times m$ real matrices, $\mathbb{R}^n = \mathbb{R}^{n \times 1}$, and $\mathbb{R} = \mathbb{R}^1$. \mathbb{R}_+ and \mathbb{R}_{++} denote the nonnegative and positive real number, respectively. I_n (or simply I if its dimension is clear from the context) is the $n \times n$ identity matrix. The superscript “ \cdot^\top ” takes transpose. For $X \in \mathbb{R}^{m \times n}$, $X_{i,j}$ refers to its (i, j) -th entry, $|X|$ is in $\mathbb{R}^{m \times n}$ with its (i, j) -th entry $|X_{i,j}|$. Inequality $X \leq Y$ means $X_{i,j} \leq Y_{i,j}$ for all (i, j) , and similarly for $X < Y$. We use $t \downarrow 0$ to denote the case that a positive scalar t tends to 0. We use $\alpha = \mathcal{O}(\beta)$ to mean α/β is bounded uniformly as $\beta \rightarrow 0$. For any $a \in \mathbb{R}$, we define

$$\text{sgn}(a) := \begin{cases} 1, & \text{if } a > 0, \\ 0, & \text{if } a = 0, \\ -1, & \text{if } a < 0. \end{cases}$$

We denote the diagonal matrix whose i -th diagonal element is x_i by $\text{diag}(x_i)$ and define $D(x) := \text{diag}(\text{sgn}(x_i))$. The symbol $\|\cdot\|$ stands for the 2-norm. For a matrix $P \in \mathbb{R}^{m \times n}$, we use $\sigma_{\min}(P)$ and $\sigma_{\max}(P)$ to denote the smallest singular value and the largest singular value, respectively. For a differentiable mapping $G : \mathbb{V} \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$, we denote $G'(x)$ by the Jacobian of G at $x \in \mathbb{V}$ and $\nabla G(x) = G'(x)^\top$ denotes the gradient of G at x . For $x \in \mathbb{R}^n$, we also denote x by $\text{vec}(x_i)$. $\det(X)$ denotes the determinant of the matrix $X \in \mathbb{R}^{n \times n}$.

2. Preliminaries

In this section, we collect some basic notions as well as corresponding assertions, which are useful in this paper.

Definition 2.1 ([37]). Let $\mathcal{M} := \{M, N\}$ be a set of matrices $M, N \in \mathbb{R}^{n \times n}$. A matrix $R \in \mathbb{R}^{n \times n}$ is called the column representative of \mathcal{M} if

$$R_{\cdot j} \in \{M_{\cdot j}, N_{\cdot j}\}, \quad j = 1, 2, \dots, n,$$

where $R_{\cdot j}, M_{\cdot j}$ and $N_{\cdot j}$ denote the j -th column of R, M and N , respectively. \mathcal{M} is said to have the column \mathcal{W} -property if the determinants of all column representative matrices of \mathcal{M} are all positive or all negative.

Definition 2.2 ([33]). An interval matrix A^I is defined by

$$A^I := [\underline{A}, \bar{A}] = \{X : \underline{A} \leq X \leq \bar{A}\}.$$

A square interval matrix A^I is called regular if each $X \in A^I$ is nonsingular.

Definition 2.3 ([9]). The classic (one-sided) directional derivative of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ at x in the direction y is defined by

$$f'(x; y) = \lim_{t \downarrow 0} \frac{f(x + ty) - f(x)}{t},$$

provided that the limit exists. Accordingly,

$$F'(x; y) = [F'_1(x; y), \dots, F'_m(x; y)]^\top$$

denotes the directional derivative for the vector-valued function $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$.

Definition 2.4 ([9]). A vector-valued function $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is said to be Lipschitz continuous on a set $S \subset \mathbb{R}^n$ if there is a constant $L > 0$ such that

$$\|F(x) - F(y)\| \leq L\|x - y\|, \quad x, y \in S.$$

Moreover, F is called locally Lipschitz continuous on \mathbb{R}^n if it is Lipschitz continuous on all compact subsets $S \subset \mathbb{R}^n$.

If $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is locally Lipschitz continuous, by Rademacher's Theorem, F is differentiable almost everywhere [31]. Let D_F be the set where F is differentiable, then the generalized Jacobian of F at x in the sense of Clarke [6] is

$$\partial F(x) = \text{co} \left\{ \lim_{x^{(k)} \in D_F, x^{(k)} \rightarrow x} \nabla F(x^{(k)}) \right\},$$

where "co" denotes the convex hull.

Definition 2.5 ([32]). *A locally Lipschitz continuous vector-valued function $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is called semismooth at x if $\lim_{V \in \partial F(x+td), d' \rightarrow d, t \downarrow 0} \{Vd'\}$ exists for any $d \in \mathbb{R}^n$.*

Lemma 2.1 ([32]). *Let $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$, then the directional derivative $F'(x; d)$ exists for any $d \in \mathbb{R}^n$ if F is semismooth at x .*

Lemma 2.2 ([32]). *Suppose that $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is semismooth at x . Then it is called strongly semismooth at x if*

$$Vd - F'(x; d) = \mathcal{O}(\|d\|^2)$$

for any $V \in \partial F(x + d)$ and $d \rightarrow 0$.

It is known that, if Assumption 1.1 holds, GAVE (1.1) has a unique solution for any $b \in \mathbb{R}^n$ [27]. In addition, we have the following lemma, which is needed in the subsequent discussion.

Lemma 2.3. *Assumption 1.1 holds if and only if matrix $A+BD$ is nonsingular for any diagonal matrix $D = \text{diag}(d_i)$ with $d_i \in [-1, 1]$, $i = 1, 2, \dots, n$.*

Proof. The result can be straightly derived from Theorems 1.1 and 1.2. Indeed, it follows from Theorem 1.1 that Assumption 1.1 holds if and only if

$$\det[(A + B)\bar{D} + (A - B)\tilde{D}] \neq 0$$

for any nonnegative diagonal matrices $\bar{D}, \tilde{D} \in \mathbb{R}^{n \times n}$ with $\bar{D} + \tilde{D} > 0$, that is,

$$\det[A + B(\bar{D} - \tilde{D})(\bar{D} + \tilde{D})^{-1}] \neq 0 \quad (2.1)$$

for any nonnegative diagonal matrices $\bar{D}, \tilde{D} \in \mathbb{R}^{n \times n}$ with $\bar{D} + \tilde{D} > 0$.

Let

$$\mathcal{D}_1 := \{D \in \mathbb{R}^{n \times n} : D = \text{diag}(d_i), d_i \in [-1, 1], i = 1, 2, \dots, n\},$$

$$\mathcal{D}_2 := \{D \in \mathbb{R}^{n \times n} : D = (\bar{D} - \tilde{D})(\bar{D} + \tilde{D})^{-1}, \bar{D} = \text{diag}(\bar{d}_i) \geq 0,$$

$$\tilde{D} = \text{diag}(\tilde{d}_i) \geq 0, \bar{D} + \tilde{D} > 0\}.$$

Then, on one hand, for any $D \in \mathcal{D}_2$, we have

$$|D_{i,i}| = \frac{|\bar{d}_i - \tilde{d}_i|}{|\bar{d}_i + \tilde{d}_i|} \leq 1.$$

Thus, $\mathcal{D}_2 \subseteq \mathcal{D}_1$. On the other hand, for any $D = \text{diag}(d_i) \in \mathcal{D}_1$, $d_i \in [-1, 1]$ can be expressed by $d_i = (\bar{d}_i - \tilde{d}_i)/(\bar{d}_i + \tilde{d}_i)$ with

$$\begin{cases} \bar{d}_i > 0 \quad \text{and} \quad \tilde{d}_i = 0, & \text{if } d_i = 1, \\ \bar{d}_i = 0 \quad \text{and} \quad \tilde{d}_i > 0, & \text{if } d_i = -1, \\ \bar{d}_i = \frac{(1 + d_i)\tilde{d}_i}{1 - d_i} \quad \text{and} \quad \tilde{d}_i > 0, & \text{if } d_i \in (-1, 1). \end{cases}$$

Hence, $\mathcal{D}_1 \subseteq \mathcal{D}_2$. It follows from the above discussion that $\mathcal{D}_1 = \mathcal{D}_2$. Then (2.1) is equivalent to

$$\det(A + BD) \neq 0$$

for any $D = \text{diag}(d_i)$ with $d_i \in [-1, 1]$, $i = 1, 2, \dots, n$. This completes the proof. \square

Remark 2.1. For symmetric matrices A and B , under the assumption that $\sigma_{\min}(A) > \sigma_{\max}(B)$, the authors in [2, Lemma 1] proved the nonsingularity of $A + BD$ for any diagonal matrix D whose elements are equal to 1, 0 or -1 . We should mentioned that the symmetries of the matrices A and B can be relaxed there and our result here is more general than theirs.

Remark 2.2. In [17], the authors used the assumption that $\sigma_{\min}(A) > \sigma_{\max}(B)$, while in [40], the authors used the assumption that the interval matrix $[A - |B|, A + |B|]$ is regular. The interval matrix $[A - |B|, A + |B|]$ is regular is weaker than $\sigma_{\min}(A) > \sigma_{\max}(B)$ and examples can be found in [49, Examples 2.1, 2.3]. In addition, it is easy to prove that $[A - |B|, A + |B|]$ is regular implies that Assumption 1.1 holds, but the reverse is not true. For instance, let

$$A = \begin{bmatrix} 1001 & -496 \\ -994 & 501 \end{bmatrix}, \quad B = \begin{bmatrix} 999 & -494 \\ -995 & 499 \end{bmatrix},$$

then $\{A + B, A - B\}$ has the column \mathcal{W} -property [27] while $[A - |B|, A + |B|]$ is not regular. Indeed, there exists a singular matrix

$$\begin{bmatrix} 2 & -2 \\ -2 & 2 \end{bmatrix} \in [A - |B|, A + |B|].$$

In conclusion, our Assumption 1.1 here is weaker than the assumptions used in [17, 40].

3. The Algorithm

In this section, we develop a non-monotone smoothing Newton algorithm for solving GAVE (1.1). To this end, we first consider an equivalent reformulation of GAVE (1.1) by introducing a smoothing function for the absolute value operator.

3.1. A smoothing function for $|x|$ with $x \in \mathbb{R}$

In this subsection, we consider a smoothing function for $|x|$ with $x \in \mathbb{R}$ and discuss some of its properties, which lay the foundation of the next subsection.

Since $|x|$ is not differentiable at $x = 0$, in order to overcome the hurdle in analysis and application, researchers construct numerous smoothing functions for it [35]. In this paper, we adopt the following smoothing function $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$, defined by

$$\phi(\mu, x) = \sqrt{\mu^2 + x^2} - \mu, \tag{3.1}$$

which can be derived from the perspective of the convex conjugate [35].

In the following, we give some properties related to the smoothing function (3.1).

Proposition 3.1. *Let ϕ be defined by (3.1), then we have*

(i) $\phi(0, x) = |x|$.

(ii) ϕ is continuously differentiable on $\mathbb{R}^2 \setminus \{(0, 0)\}$, and when $(\mu, x) \neq (0, 0)$, we have

$$\frac{\partial \phi}{\partial \mu} = \frac{\mu}{\sqrt{\mu^2 + x^2}} - 1, \quad \frac{\partial \phi}{\partial x} = \frac{x}{\sqrt{\mu^2 + x^2}}.$$

(iii) ϕ is a convex function on \mathbb{R}^2 , i.e.

$$\phi(\alpha(\bar{\mu}, \bar{x}) + (1 - \alpha)(\tilde{\mu}, \tilde{x})) \leq \alpha\phi(\bar{\mu}, \bar{x}) + (1 - \alpha)\phi(\tilde{\mu}, \tilde{x})$$

for all $(\bar{\mu}, \bar{x}), (\tilde{\mu}, \tilde{x}) \in \mathbb{R}^2$ and $\alpha \in [0, 1]$.

(iv) ϕ is Lipschitz continuous on \mathbb{R}^2 .

(v) ϕ is strongly semismooth on \mathbb{R}^2 .

Proof. The proofs of (i) and (ii) are trivial. Now we turn to the result (iii). For any $(\bar{\mu}, \bar{x}), (\tilde{\mu}, \tilde{x}) \in \mathbb{R}^2$ and $\alpha \in [0, 1]$, we have

$$\begin{aligned} & \phi(\alpha(\bar{\mu}, \bar{x}) + (1 - \alpha)(\tilde{\mu}, \tilde{x})) - \alpha\phi(\bar{\mu}, \bar{x}) - (1 - \alpha)\phi(\tilde{\mu}, \tilde{x}) \\ &= \sqrt{[\alpha\bar{\mu} + (1 - \alpha)\tilde{\mu}]^2 + [\alpha\bar{x} + (1 - \alpha)\tilde{x}]^2} - \alpha\bar{\mu} - (1 - \alpha)\tilde{\mu} \\ & \quad - \alpha\sqrt{\bar{\mu}^2 + \bar{x}^2} + \alpha\bar{\mu} - (1 - \alpha)\sqrt{\tilde{\mu}^2 + \tilde{x}^2} + (1 - \alpha)\tilde{\mu} \\ &= \sqrt{[\alpha\bar{\mu} + (1 - \alpha)\tilde{\mu}]^2 + [\alpha\bar{x} + (1 - \alpha)\tilde{x}]^2} - \alpha\sqrt{\bar{\mu}^2 + \bar{x}^2} \\ & \quad - (1 - \alpha)\sqrt{\tilde{\mu}^2 + \tilde{x}^2}. \end{aligned} \tag{3.2}$$

On one hand,

$$\begin{aligned} & \left(\sqrt{[\alpha\bar{\mu} + (1 - \alpha)\tilde{\mu}]^2 + [\alpha\bar{x} + (1 - \alpha)\tilde{x}]^2} \right)^2 \\ &= \alpha^2(\bar{\mu}^2 + \bar{x}^2) + (1 - \alpha)^2(\tilde{\mu}^2 + \tilde{x}^2) + 2\alpha(1 - \alpha)(\bar{\mu}\tilde{\mu} + \bar{x}\tilde{x}). \end{aligned} \tag{3.3}$$

On the other hand,

$$\begin{aligned} & \left[\alpha\sqrt{\bar{\mu}^2 + \bar{x}^2} + (1 - \alpha)\sqrt{\tilde{\mu}^2 + \tilde{x}^2} \right]^2 \\ &= \alpha^2(\bar{\mu}^2 + \bar{x}^2) + (1 - \alpha)^2(\tilde{\mu}^2 + \tilde{x}^2) + 2\alpha(1 - \alpha)\sqrt{(\bar{\mu}^2 + \bar{x}^2)(\tilde{\mu}^2 + \tilde{x}^2)} \\ &\geq \alpha^2(\bar{\mu}^2 + \bar{x}^2) + (1 - \alpha)^2(\tilde{\mu}^2 + \tilde{x}^2) + 2\alpha(1 - \alpha)|\bar{\mu}\tilde{\mu} + \bar{x}\tilde{x}|. \end{aligned} \tag{3.4}$$

Then the result (iii) follows from (3.2)-(3.4).

Consider the result (iv). For any $(\bar{\mu}, \bar{x}), (\tilde{\mu}, \tilde{x}) \in \mathbb{R}^2$, we have

$$\begin{aligned} |\phi(\bar{\mu}, \bar{x}) - \phi(\tilde{\mu}, \tilde{x})| &= \left| \|\bar{\mu}, \bar{x}\| - \|\tilde{\mu}, \tilde{x}\| + \tilde{\mu} \right| \\ &\leq \left| \|\bar{\mu}, \bar{x}\| - \|\tilde{\mu}, \tilde{x}\| \right| + |\bar{\mu} - \tilde{\mu}| \\ &\leq \|\bar{\mu} - \tilde{\mu}, \bar{x} - \tilde{x}\| + \|\bar{\mu} - \tilde{\mu}, \bar{x} - \tilde{x}\| \\ &= 2\|\bar{\mu} - \tilde{\mu}, \bar{x} - \tilde{x}\|. \end{aligned}$$

Hence, ϕ is Lipschitz continuous with Lipschitz constant 2.

Finally, we prove the result (v). It follows from the result (iii) that ϕ is semismooth on \mathbb{R}^2 [32]. Note that ϕ is arbitrarily many times differentiable for all $(\mu, x) \in \mathbb{R}^2$ with $(\mu, x) \neq (0, 0)$ and hence strongly semismooth at these points. Therefore, it is sufficient to show that it is strongly semismooth at $(0, 0)$. For any $(\mu, x) \in \mathbb{R}^2 \setminus \{(0, 0)\}$, ϕ is differentiable at (μ, x) , and hence,

$$\partial\phi(\mu, x) = \nabla\phi(\mu, x) = \left[\frac{\partial\phi(\mu, x)}{\partial\mu}, \frac{\partial\phi(\mu, x)}{\partial x} \right]^\top.$$

In addition, by Lemma 2.1, the classic directional derivative of ϕ at $(0, 0)$ exists and

$$\phi'((0, 0); (\mu, x)) = \lim_{t \downarrow 0} \frac{\phi((0, 0) + t(\mu, x)) - \phi(0, 0)}{t} = \phi(\mu, x),$$

from which we have

$$\begin{aligned} & \phi(\mu, x) - \left[\frac{\partial \phi(\mu, x)}{\partial \mu}, \frac{\partial \phi(\mu, x)}{\partial x} \right] \begin{bmatrix} \mu \\ x \end{bmatrix} \\ &= \sqrt{\mu^2 + x^2} - \mu - \left(\frac{\mu}{\sqrt{\mu^2 + x^2}} - 1 \right) \mu - \frac{x}{\sqrt{\mu^2 + x^2}} x \\ &= 0 = \mathcal{O}(\|(\mu, x)\|^2). \end{aligned}$$

Then the result follows from Lemma 2.2. \square

3.2. The reformulation of GAVE (1.1)

In this subsection, based on the earlier subsection, we will give a reformulation of GAVE (1.1) and explore some of its properties.

Let $z := (\mu, x) \in \mathbb{R} \times \mathbb{R}^n$, we first define the function $H : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R} \times \mathbb{R}^n$ as

$$H(z) := \begin{bmatrix} \mu \\ Ax + B\Phi(\mu, x) - b \end{bmatrix}, \quad (3.5)$$

where $\Phi : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ is defined by

$$\Phi(\mu, x) := \begin{bmatrix} \phi(\mu, x_1) \\ \phi(\mu, x_2) \\ \vdots \\ \phi(\mu, x_n) \end{bmatrix}$$

with ϕ being the smoothing function given in (3.1). According to Proposition 3.1(i), it holds that

$$H(z) = 0 \Leftrightarrow \mu = 0 \quad \text{and} \quad x \quad \text{is a solution of GAVE (1.1)}. \quad (3.6)$$

Then it follows from (3.6) that solving GAVE (1.1) is equivalent to solving the system of nonlinear equations $H(z) = 0$. Before giving the algorithm for solving $H(z) = 0$, we will give some properties of the function H .

Proposition 3.2. *Let H be defined by (3.5), then we have*

- (i) *H is continuously differentiable on $\mathbb{R}^{n+1} \setminus \{0\}$, and when $\mu = 0$ and $x_i \neq 0$ for all $i = 1, 2, \dots, n$ or $\mu \neq 0$, the Jacobian matrix of H is given by*

$$H'(z) = \begin{bmatrix} 1 & 0 \\ BV_1 & A + BV_2 \end{bmatrix} \quad (3.7)$$

with

$$V_1 = \begin{bmatrix} \frac{\mu}{\sqrt{\mu^2 + x_1^2}} - 1 \\ \frac{\mu}{\sqrt{\mu^2 + x_2^2}} - 1 \\ \vdots \\ \frac{\mu}{\sqrt{\mu^2 + x_n^2}} - 1 \end{bmatrix}, \quad V_2 = \begin{bmatrix} \frac{x_1}{\sqrt{\mu^2 + x_1^2}} & 0 & 0 & 0 \\ 0 & \frac{x_2}{\sqrt{\mu^2 + x_2^2}} & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \frac{x_n}{\sqrt{\mu^2 + x_n^2}} \end{bmatrix}. \quad (3.8)$$

- (ii) *H is strongly semismooth on \mathbb{R}^{n+1} .*

Proof. The result (i) holds from Proposition 3.1(ii). Now we turn to prove the result (ii). Since H is strongly semismooth on \mathbb{R}^{n+1} if and only if its component function H_i , $i = 1, 2, \dots, n$, are [32], and the composition of strongly semismooth functions is a strongly semismooth function [9, Theorem 19], the result (ii) follows from Proposition 3.1(v) and the fact that a continuously differentiable function with a Lipschitz continuous gradient is strongly semismooth [14]. \square

3.3. The non-monotone smoothing Newton algorithm for GAVE (1.1)

Now we are in position to develop a non-monotone smoothing Newton algorithm to solve the system of nonlinear equations $H(z) = 0$, and so is GAVE (1.1).

Let $H(z)$ be given in (3.5) and define the merit function $\mathcal{M} : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}_+$ by

$$\mathcal{M}(z) := \|H(z)\|^2.$$

Clearly, solving the system of nonlinear equations $H(z) = 0$ is equivalent to solving the following unconstrained optimization problem

$$\min_{z \in \mathbb{R}^{n+1}} \mathcal{M}(z)$$

with the vanished objective function value. We now propose a non-monotone smoothing Newton algorithm to solve $H(z) = 0$ by minimizing the merit function $\mathcal{M}(z)$, which is described in Algorithm 3.1.

Remark 3.1. The development of Algorithm 3.1 is inspired by the non-monotone smoothing Newton algorithm for the weighted complementarity problem [39] and the non-monotone Levenberg-Marquardt type method for the weighted nonlinear complementarity problem [41].

Before ending this section, we will show that Algorithm 3.1 is well-defined. To this end, we need the following lemma.

Lemma 3.1. *Let $H'(z)$ be defined by (3.7) and (3.8). If Assumption 1.1 holds, then $H'(z)$ is nonsingular at any $z = (\mu, x) \in \mathbb{R}_{++} \times \mathbb{R}^n$.*

Proof. From (3.7), we need only to show that $A + BV_2$ is nonsingular. Since Assumption 1.1 holds and $|x_i/\sqrt{x_i^2 + \mu^2}| < 1$, $i = 1, 2, \dots, n$, the result immediately follows from Lemma 2.3. The proof is complete. \square

Then we have the following theorem.

Theorem 3.1. *If Assumption 1.1 holds, Algorithm 3.1 is well defined and either terminates in finitely many steps or generates an infinite sequence $\{z^{(k)}\}$ satisfying $\mathcal{M}(z^{(k)}) \leq \mathcal{C}^{(k)}$, $\mu^{(k)} > 0$ and $\beta^{(k)} < \mu^{(k)}$ for all $k \geq 0$.*

Proof. We will prove it by mathematical induction. Suppose that $\mathcal{M}(z^{(k)}) \leq \mathcal{C}^{(k)}$, $\mu^{(k)} > 0$ and $\beta^{(k)} < \mu^{(k)}$ for some k . Since $\mu^{(k)} > 0$, it follows from Lemma 3.1 that $H'(z^{(k)})$ is nonsingular. Hence, $\Delta z^{(k)}$ can be uniquely determined by (3.9). If $\|H(z^{(k)})\| = 0$, then Algorithm 3.1 terminates. Otherwise, $\|H(z^{(k)})\|^2 = \mathcal{M}(z^{(k)}) \leq \mathcal{C}^{(k)}$ implies that $\mathcal{C}^{(k)} > 0$, from which and the second equation in (3.12) we have $\beta^{(k)} = \gamma \mathcal{C}^{(k)} > 0$. In the following, we divide our proof into three parts.

Algorithm 3.1: A Non-Monotone Smoothing Newton Algorithm (NSNA) for GAVE (1.1).

1 Choose $\theta, \delta \in (0, 1)$ and $z^{(0)} := (\mu^{(0)}, x^{(0)}) \in \mathbb{R}_{++} \times \mathbb{R}^n$. Let $\mathcal{C}^{(0)} := \mathcal{M}(z^{(0)})$. Choose $\gamma \in (0, 1)$ such that $\beta^{(0)} = \gamma\mathcal{C}^{(0)} < \mu^{(0)}$ and $\gamma\mu^{(0)} < 1$. Set $k := 0$.

2 If $\|H(z^{(k)})\| = 0$, then stop. Else, compute the search direction $\Delta z^{(k)} = (\Delta\mu^{(k)}, \Delta x^{(k)}) \in \mathbb{R} \times \mathbb{R}^n$ by solving the perturbed Newton system

$$H'(z^{(k)})\Delta z^{(k)} = -H(z^{(k)}) + \beta^{(k)}e^{(1)}, \quad (3.9)$$

where $e^{(1)} = [1, 0]^\top \in \mathbb{R} \times \mathbb{R}^n$. If $\Delta z^{(k)}$ satisfies

$$\|H(z^{(k)} + \Delta z^{(k)})\| \leq \theta \|H(z^{(k)})\|, \quad (3.10)$$

then set $z^{(k+1)} := z^{(k)} + \Delta z^{(k)}$ and go to step 4. Otherwise, go to step 3.

3 Let $\alpha^{(k)}$ be the maximum of the values $1, \delta, \delta^2, \dots$ such that

$$\mathcal{M}(z^{(k)} + \alpha^{(k)}\Delta z^{(k)}) \leq \mathcal{C}^{(k)} - \gamma\|\alpha^{(k)}\Delta z^{(k)}\|^2. \quad (3.11)$$

Set $z^{(k+1)} := z^{(k)} + \alpha^{(k)}\Delta z^{(k)}$ and go to step 4.

4 Compute $\mathcal{M}(z^{(k+1)}) = \|H(z^{(k+1)})\|^2$ and set

$$\mathcal{C}^{(k+1)} := \frac{(\mathcal{C}^{(k)} + 1)\mathcal{M}(z^{(k+1)})}{\mathcal{M}(z^{(k+1)}) + 1}, \quad \beta^{(k+1)} := \gamma\mathcal{C}^{(k+1)}. \quad (3.12)$$

5 Set $k := k + 1$ and go to step 2.

Firstly, we will show that $\mu^{(k+1)} > 0$. On one hand, if $z^{(k+1)}$ is generated by step 2, it follows from (3.9) that

$$\mu^{(k+1)} = \mu^{(k)} + \Delta\mu^{(k)} = \mu^{(k)} + (-\mu^{(k)} + \beta^{(k)}) = \beta^{(k)} > 0.$$

On the other hand, if $z^{(k+1)}$ is generated by step 3, we first show that there exists at least a nonnegative integer l satisfying (3.11). On the contrary, for any nonnegative integer l , we have

$$\mathcal{M}(z^{(k)} + \delta^l \Delta z^{(k)}) > \mathcal{C}^{(k)} - \gamma\|\delta^l \Delta z^{(k)}\|^2,$$

which together with $\mathcal{M}(z^{(k)}) \leq \mathcal{C}^{(k)}$ gives

$$\frac{\mathcal{M}(z^{(k)} + \delta^l \Delta z^{(k)}) - \mathcal{M}(z^{(k)})}{\delta^l} + \gamma\delta^l \|\Delta z^{(k)}\|^2 > 0.$$

Since \mathcal{M} is differentiable at $z^{(k)}$ and $\delta \in (0, 1)$, by letting $l \rightarrow +\infty$ in the above inequality, we have

$$\mathcal{M}'(z^{(k)})\Delta z^{(k)} \geq 0. \quad (3.13)$$

In addition, from (3.9) we have

$$\mathcal{M}'(z^{(k)})\Delta z^{(k)} = 2H(z^{(k)})^\top H'(z^{(k)})\Delta z^{(k)}$$

$$\begin{aligned}
&= -2\|H(z^{(k)})\|^2 + 2\mu^{(k)}\beta^{(k)} \\
&= 2\mu^{(k)}(\beta^{(k)} - \mu^{(k)}) - 2\|Ax^{(k)} + B\Phi(\mu^{(k)}, x^{(k)}) - b\|^2.
\end{aligned} \tag{3.14}$$

Since $\mu^{(k)} > 0$ and $\beta^{(k)} < \mu^{(k)}$, (3.14) implies that $\mathcal{M}'(z^{(k)})\Delta z^{(k)} < 0$, which contradicts to (3.13). Therefore, there exists $\alpha^{(k)} \in (0, 1]$ such that

$$z^{(k+1)} = z^{(k)} + \alpha^{(k)}\Delta z^{(k)}$$

in step 3. In this case, it follows from (3.9) that

$$\mu^{(k+1)} = (1 - \alpha^{(k)})\mu^{(k)} + \alpha^{(k)}\beta^{(k)} > 0.$$

Secondly, we will show that $\mathcal{M}(z^{(k+1)}) < \mathcal{C}^{(k+1)}$. Indeed, if $z^{(k+1)}$ is generated by step 2, then it follows from $\theta \in (0, 1)$ and (3.10) that

$$\mathcal{M}(z^{(k+1)}) \leq \theta^2 \mathcal{M}(z^{(k)}) < \mathcal{M}(z^{(k)}) \leq \mathcal{C}^{(k)}.$$

Otherwise, by step 3, we can also obtain $\mathcal{M}(z^{(k+1)}) < \mathcal{C}^{(k)}$. In fact, $\beta^{(k)} < \mu^{(k)}$ implies that $\Delta z^{(k)} \neq 0$. Thereby, (3.11) implies that $\mathcal{M}(z^{(k+1)}) < \mathcal{C}^{(k)}$. Consequently, $\mathcal{M}(z^{(k+1)}) < \mathcal{C}^{(k)}$ and the first equation in (3.12) imply

$$\mathcal{C}^{(k+1)} = \frac{(\mathcal{C}^{(k)} + 1)\mathcal{M}(z^{(k+1)})}{\mathcal{M}(z^{(k+1)}) + 1} > \frac{(\mathcal{M}^{(k+1)} + 1)\mathcal{M}(z^{(k+1)})}{\mathcal{M}(z^{(k+1)}) + 1} = \mathcal{M}(z^{(k+1)}).$$

Finally, we will show that $\mu^{(k+1)} > \beta^{(k+1)}$. As mentioned earlier, we have $\mu^{(k+1)} = \beta^{(k)}$ by step 2 and

$$\mu^{(k+1)} = (1 - \alpha^{(k)})\mu^{(k)} + \alpha^{(k)}\beta^{(k)}$$

by step 3, respectively. For the latter, since $\alpha^{(k)} \in (0, 1]$ and $\mu^{(k)} > \beta^{(k)} > 0$,

$$\mu^{(k+1)} = (1 - \alpha^{(k)})\mu^{(k)} + \alpha^{(k)}\beta^{(k)} \geq (1 - \alpha^{(k)})\beta^{(k)} + \alpha^{(k)}\beta^{(k)} = \beta^{(k)}.$$

In a word, $\mu^{(k+1)} \geq \beta^{(k)}$. In addition, it follows from $\mathcal{M}(z^{(k+1)}) < \mathcal{C}^{(k)}$ and the first equation in (3.12) that

$$0 \leq \mathcal{C}^{(k+1)} = \frac{\mathcal{C}^{(k)}\mathcal{M}(z^{(k+1)}) + \mathcal{M}(z^{(k+1)})}{\mathcal{M}(z^{(k+1)}) + 1} < \frac{\mathcal{C}^{(k)}\mathcal{M}(z^{(k+1)}) + \mathcal{C}^{(k)}}{\mathcal{M}(z^{(k+1)}) + 1} = \mathcal{C}^{(k)},$$

from which and $\gamma > 0$ we obtain

$$\mu^{(k+1)} \geq \beta^{(k)} = \gamma\mathcal{C}^{(k)} > \gamma\mathcal{C}^{(k+1)} = \beta^{(k+1)}.$$

The proof is completed by letting $\mathcal{M}(z^{(0)}) \leq \mathcal{C}^{(0)}$, $\mu^{(0)} > 0$ and $\beta^{(0)} < \mu^{(0)}$. \square

Remark 3.2. We should mention that the equation (3.14) plays the key role in the proof of Theorem 3.1. This equation motivates us to develop the algorithm with the property $\beta^{(k)} < \mu^{(k)}$, which is slightly different from that given in [39] ($\beta^{(k)} \leq \mu^{(k)}$ was proved there).

4. Convergence Analysis

In this section, we will analyze the convergence of Algorithm 3.1. In what follows, we assume that $\|H(z^{(k)})\| \neq 0$ for all $k \geq 0$. To establish the global convergence of Algorithm 3.1, we need the following lemmas.

Lemma 4.1. *Suppose that Assumption 1.1 holds. Let $\{z^{(k)} = (\mu^{(k)}, x^{(k)})\}$ be the iteration sequence generated by Algorithm 3.1. Then $\mathcal{C}^{(k)} > \mathcal{C}^{(k+1)}$ and $\mu^{(k)} > \mu^{(k+1)}$ for all $k \geq 0$.*

Proof. The proof of $\mathcal{C}^{(k)} > \mathcal{C}^{(k+1)}$ for all $k \geq 0$ can be found in the proof of Theorem 3.1. It follows from Theorem 3.1 that $\mu^{(k)} > \beta^{(k)}$ for all $k \geq 0$. Then, by step 2, we have $\mu^{(k+1)} = \beta^{(k)} < \mu^{(k)}$. By step 3,

$$\mu^{(k+1)} = (1 - \alpha^{(k)})\mu^{(k)} + \alpha^{(k)}\beta^{(k)} < (1 - \alpha^{(k)})\mu^{(k)} + \alpha^{(k)}\mu^{(k)} = \mu^{(k)}.$$

The proof is complete. \square

Lemma 4.2. *If Assumption 1.1 holds, then $\{z^{(k)}\}$ generated by Algorithm 3.1 is bounded.*

Proof. We first prove that the level set

$$\mathcal{L}(\Lambda) := \{z = (\mu, x) \in \mathbb{R}^{n+1} : \|H(z)\| \leq \Lambda\}$$

is bounded for any $\Lambda > 0$. On the contrary, there exists a sequence $\{\bar{z}^{(k)} = (\bar{\mu}^{(k)}, \bar{x}^{(k)})\}$ such that $\lim_{k \rightarrow \infty} \|\bar{z}^{(k)}\| = \infty$ and $\|H(\bar{z}^{(k)})\| \leq \Lambda$, where $\Lambda > 0$ is some constant. Since

$$\|H(\bar{z}^{(k)})\|^2 = (\bar{\mu}^{(k)})^2 + \|A\bar{x}^{(k)} + B\Phi(\bar{\mu}^{(k)}, \bar{x}^{(k)}) - b\|^2, \quad (4.1)$$

we can conclude that $\{\bar{\mu}^{(k)}\}$ is bounded. It follows from this and the unboundedness of $\{(\bar{\mu}^{(k)}, \bar{x}^{(k)})\}$ that $\lim_{k \rightarrow \infty} \|\bar{x}^{(k)}\| = \infty$. Since the sequence $\{\bar{x}^{(k)}/\|\bar{x}^{(k)}\|\}$ is bounded, it has at least one accumulation point \hat{x} . Then, there exists a subsequence $\{\bar{x}^{(k)}\}_{k \in K}$ such that $\lim_{k \in K, k \rightarrow +\infty} (\bar{x}^{(k)}/\|\bar{x}^{(k)}\|) = \hat{x}$ with $K \subset \{0, 1, 2, \dots\}$. It follows from the continuity of the 2-norm that $\|\hat{x}\| = 1$. In the following, we remain $k \in K$. From (4.1), we have

$$\begin{aligned} \frac{\Lambda^2}{\|\bar{x}^{(k)}\|^2} &\geq \frac{\|H(\bar{z}^{(k)})\|^2}{\|\bar{x}^{(k)}\|^2} \\ &= \frac{(\bar{\mu}^{(k)})^2}{\|\bar{x}^{(k)}\|^2} + \left\| A \frac{\bar{x}^{(k)}}{\|\bar{x}^{(k)}\|} + B \frac{\Phi(\bar{\mu}^{(k)}, \bar{x}^{(k)})}{\|\bar{x}^{(k)}\|} - \frac{b}{\|\bar{x}^{(k)}\|} \right\|^2. \end{aligned} \quad (4.2)$$

Since

$$\frac{\sqrt{(\bar{\mu}^{(k)})^2 + (\bar{x}_i^{(k)})^2} - \bar{\mu}^{(k)}}{\|\bar{x}^{(k)}\|} = \sqrt{\left(\frac{\bar{\mu}^{(k)}}{\|\bar{x}^{(k)}\|}\right)^2 + \left(\frac{\bar{x}_i^{(k)}}{\|\bar{x}^{(k)}\|}\right)^2} - \frac{\bar{\mu}^{(k)}}{\|\bar{x}^{(k)}\|}, \quad i = 1, 2, \dots, n,$$

from the boundedness of $\{\bar{\mu}^{(k)}\}$, we have

$$\lim_{k \rightarrow \infty} \frac{\sqrt{(\bar{\mu}^{(k)})^2 + (\bar{x}_i^{(k)})^2} - \bar{\mu}^{(k)}}{\|\bar{x}^{(k)}\|} = \sqrt{\hat{x}_i^2} = |\hat{x}_i|.$$

Hence, by letting $k \rightarrow \infty$ in (4.2), we have $A\hat{x} + B|\hat{x}| = 0$, i.e. $[A + BD(\hat{x})]\hat{x} = 0$. Since $D(\hat{x}) \in [-I, I]$, it follows from Lemma 4.3 that $A + BD(\hat{x})$ is nonsingular. Thus, we have $\hat{x} = 0$, which contradicts to the fact that $\|\hat{x}\| = 1$.

If $\{z^{(k)}\}$ is generated by Algorithm 3.1, then $\|H(z^{(k)})\| \leq \sqrt{\mathcal{C}^{(0)}}$ for all $k \geq 0$. Hence, $\{z^{(k)}\}$ is bounded based on the aforementioned discussion. \square

Remark 4.1. The proof of Lemma 4.2 is inspired by that of [40, Theorem 2.3], which was considered in the case that the interval matrix $[A - |B|, A + |B|]$ is regular. In addition, similar to the proof of [17, Lemma 4.1], the boundedness of $\{z^{(k)}\}$ can be derived under the assumption that $\sigma_{\min}(A) > \sigma_{\max}(B)$. Our result here seems more general than those in [17, Lemma 4.1].

Now we show the global convergence of Algorithm 3.1.

Theorem 4.1. *Assume that Assumption 1.1 holds. Let $\{z^{(k)} = (\mu^{(k)}, x^{(k)})\}$ be the iteration sequence generated by Algorithm 3.1. Then any accumulation point z^* of $\{z^{(k)}\}$ satisfies*

$$H(z^*) = 0,$$

i.e. $z^* = (0, x^*)$ and x^* is a solution of GAVE (1.1).

Proof. Lemma 4.2 implies the existence of the accumulation point of $\{z^{(k)}\}$ generated by Algorithm 3.1. Let z^* be any accumulation point of $\{z^{(k)}\}$, then there exists a subsequence of $\{z^{(k)}\}$ converging to z^* . For convenience, we still denote the subsequence by $\{z^{(k)}\}$.

By Lemma 4.1, $\{\mathcal{C}^{(k)}\}$ is convergent because it is monotonically decreasing. Thus, there exists a constant $\mathcal{C}^* \geq 0$ such that $\lim_{k \rightarrow +\infty} \mathcal{C}^{(k)} = \mathcal{C}^*$. As $\mathcal{M}(z^{(k)}) = \|H(z^{(k)})\|^2 \leq \mathcal{C}^{(k)}$ for all $k \geq 0$, $\lim_{k \rightarrow +\infty} \|H(z^{(k)})\| = 0$ provided that $\mathcal{C}^* = 0$. Then, from the continuity of $H(z)$ we have $H(z^*) = 0$. In the following, we assume that $\mathcal{C}^* > 0$ and derive a contradiction.

According to the first equation in (3.12), we have

$$\lim_{k \rightarrow +\infty} \mathcal{M}(z^{(k+1)}) = \lim_{k \rightarrow +\infty} \left(\frac{\mathcal{C}^{(k+1)}}{1 + \mathcal{C}^{(k)} - \mathcal{C}^{(k+1)}} \right) = \mathcal{C}^* > 0. \quad (4.3)$$

By the fact that $\beta^{(k)} = \gamma\mathcal{C}^{(k)}$, we have

$$\beta^* = \lim_{k \rightarrow +\infty} \beta^{(k)} = \gamma\mathcal{C}^* > 0.$$

Based on Theorem 3.1 and Lemma 4.1, we have

$$\mu^* = \lim_{k \rightarrow +\infty} \mu^{(k)} \geq \lim_{k \rightarrow +\infty} \beta^{(k)} = \beta^* > 0.$$

Since $\mu^* > 0$, $H'(z^*)$ is nonsingular and \mathcal{M} is continuously differentiable at z^* .

Let

$$\mathcal{N} := \{k : \|H(z^{(k)} + \Delta z^{(k)})\| \leq \theta \|H(z^{(k)})\|\}.$$

We claim that \mathcal{N} must be a finite set. In fact, if \mathcal{N} is an infinite set, then

$$\|H(z^{(k)} + \Delta z^{(k)})\| \leq \theta \|H(z^{(k)})\|,$$

i.e.

$$\mathcal{M}(z^{(k+1)}) \leq \theta^2 \mathcal{M}(z^{(k)})$$

holds for infinitely many k . By letting $k \rightarrow +\infty$ with $k \in \mathcal{N}$, we have $\mathcal{C}^* \leq \theta^2 \mathcal{C}^*$. This leads to a contradiction due to $\theta \in (0, 1)$ and $\mathcal{C}^* > 0$. Hence, we can suppose that there exists an index $\bar{k} > 0$ such that

$$\|H(z^{(k)} + \Delta z^{(k)})\| > \theta \|H(z^{(k)})\|, \quad \forall k \geq \bar{k}.$$

Then, for all $k \geq \bar{k}$, $z^{(k+1)} = z^{(k)} + \alpha^{(k)} \Delta z^{(k)}$ (generated by step 3) satisfies

$$\mathcal{M}(z^{(k+1)}) \leq \mathcal{C}^{(k)} - \gamma \|\alpha^{(k)} \Delta z^{(k)}\|^2,$$

i.e.

$$\gamma \|\alpha^{(k)} \Delta z^{(k)}\|^2 \leq \mathcal{C}^{(k)} - \mathcal{M}(z^{(k+1)}),$$

from which and (4.3) we have

$$\lim_{k \rightarrow +\infty} \alpha^{(k)} \|\Delta z^{(k)}\| = 0.$$

On one hand, if $1 \geq \alpha^{(k)} = \delta^{l_k} \geq \varrho > 0$ for all $k \geq \bar{k}$ with ϱ being a fixed constant, then

$$\Delta z^* = \lim_{\bar{k} \leq k \rightarrow +\infty} \Delta z^{(k)} = 0,$$

which implies that

$$\mathcal{M}'(z^*) \Delta z^* = 0. \quad (4.4)$$

Here and in the sequel, Δz^* is the unique solution of

$$H'(z^*) \Delta z^* = -H(z^*) + \beta^* e^{(1)}.$$

On the other hand, $\{\alpha^{(k)}\}_{k \geq \bar{k}}$ has a subsequence converging to 0. Without loss of generality, we may assume that $\lim_{\bar{k} \leq k \rightarrow +\infty} \alpha^{(k)} = 0$. Let $\hat{\alpha}^{(k)} := \delta^{l_k} / \delta$, then $\lim_{\bar{k} \leq k \rightarrow +\infty} \hat{\alpha}^{(k)} = 0$. Moreover, for all $k \geq \bar{k}$, it follows from the definition of $\alpha^{(k)}$ and Theorem 3.1 that

$$\mathcal{M}(z^{(k)} + \hat{\alpha}^{(k)} \Delta z^{(k)}) > \mathcal{C}^{(k)} - \gamma \|\hat{\alpha}^{(k)} \Delta z^{(k)}\|^2 \geq \mathcal{M}(z^{(k)}) - \gamma \|\hat{\alpha}^{(k)} \Delta z^{(k)}\|^2.$$

Thus,

$$\frac{\mathcal{M}(z^{(k)} + \hat{\alpha}^{(k)} \Delta z^{(k)}) - \mathcal{M}(z^{(k)})}{\hat{\alpha}^{(k)}} + \gamma \hat{\alpha}^{(k)} \|\Delta z^{(k)}\|^2 > 0.$$

By letting $k \rightarrow +\infty$ in the above inequality, we have

$$\mathcal{M}'(z^*) \Delta z^* \geq 0. \quad (4.5)$$

Since $\mu^* \gamma \leq \gamma \mu^{(0)} < 1$, it follows from (3.9) and (4.3) that

$$\begin{aligned} \frac{1}{2} \mathcal{M}'(z^*) \Delta z^* &= H(z^*)^\top H'(z^*) \Delta z^* = -\mathcal{M}(z^*) + \mu^* \beta^* \\ &= -\mathcal{C}^* + \mu^* \gamma \mathcal{C}^* = (\mu^* \gamma - 1) \mathcal{C}^* < 0, \end{aligned}$$

which is contrary to (4.4) and (4.5). The proof of the theorem is complete. \square

Under Assumption 1.1, GAVE (1.1) has a unique solution and thus Lemma 4.2 and Theorem 4.1 imply that the sequence generated by Algorithm 3.1 has a unique accumulation z^* and $\lim_{k \rightarrow +\infty} z^{(k)} = z^*$. In the following, we will discuss the local quadratic convergence of Algorithm 3.1.

Lemma 4.3. *Assume that Assumption 1.1 holds and $z^* = (0, x^*)$ is the accumulation point of the sequence $\{z^{(k)}\}$ generated by Algorithm 3.1. We have*

$$(i) \quad \partial H(z^*) \subseteq \left\{ V : V = \begin{bmatrix} 1 & 0 \\ B \text{vec}(\kappa_i - 1) & A + B \text{diag}(\chi_i) \end{bmatrix}, \kappa_i, \chi_i \in [-1, 1], i=1, 2, \dots, n \right\}.$$

(ii) *All $V \in \partial H(z^*)$ are nonsingular.*

(iii) *There exists a neighborhood $\mathcal{N}(z^*)$ of z^* and a constant $C > 0$ such that for any $z := (\mu, x) \in \mathcal{N}(z^*)$ with $\mu > 0$, $H'(z)$ is nonsingular and $\|H'(z)^{-1}\| \leq C$.*

Proof. A direct computation yields the result (i). The result (ii) follows from (i) and Lemma 2.3, and the result (iii) follows from [31, Lemma 2.6]. \square

Owing to Proposition 3.2(ii) and Lemma 4.3, we can obtain the following local quadratic convergence theorem of Algorithm 3.1. The theorem was well known in the application of smoothing-type Newton methods. The theorem as a whole can be implied by [39, Theorem 8] and thus we omit the proof here.

Theorem 4.2. *Assume that Assumption 1.1 holds and z^* is the accumulation point of the sequence $\{z^{(k)}\}$ generated by Algorithm 3.1. Then the whole sequence $\{z^{(k)}\}$ converges to z^* with*

$$\|z^{(k+1)} - z^*\| = \mathcal{O}(\|z^{(k)} - z^*\|^2).$$

5. Numerical Results

In this section, we will present two numerical examples to illustrate the performance of Algorithm 3.1. Three algorithms will be tested, i.e. Algorithm 3.1 (denoted by ‘‘NSNA’’), the monotone smoothing Newton algorithm proposed by Jiang and Zhang [17] (denoted by ‘‘JZ-MSNA’’), and the monotone smoothing Newton algorithm proposed by Tang and Zhou [40] (denoted by ‘‘TZ-MSNA’’). All experiments are implemented in MATLAB R2018b with a machine precision 2.22×10^{-16} on a PC Windows 10 operating system with an Intel i7-9700 CPU and 8 GB RAM.

We will apply the aforementioned algorithms to solve GAVE (1.1) arising from HLCP. Given $M, N \in \mathbb{R}^{n \times n}$ and $q \in \mathbb{R}^n$, HLCP is to find a pair $(z, w) \in \mathbb{R}^n \times \mathbb{R}^n$ such that

$$Mz - Nw = q, \quad z \geq 0, \quad w \geq 0, \quad z^\top w = 0. \quad (5.1)$$

The equivalent relationship between GAVE (1.1) and HLCP (5.1) can be found in [27, Proposition 1]. The following two examples are frequently used in the literature.

Example 5.1. Consider HLCP (5.1) with $M = \hat{A} + \xi I$ and $N = \hat{B} + \zeta I$ [28], where

$$\hat{A} = \begin{bmatrix} S & -I & & & & & & & \\ -I & S & -I & & & & & & \\ & -I & S & -I & & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & -I & S & -I & & \\ & & & & & -I & S & & \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} S & & & & & & & \\ & S & & & & & & \\ & & S & & & & & \\ & & & \ddots & & & & \\ & & & & S & & & \\ & & & & & S & & \\ & & & & & & S & \end{bmatrix},$$

$$S = \begin{bmatrix} 4 & -1 & & & & \\ -1 & 4 & -1 & & & \\ & -1 & 4 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 4 & -1 \\ & & & & -1 & 4 \end{bmatrix}.$$

Example 5.2. Consider HLCP (5.1) with $M = \hat{A} + \xi I$ and $N = \hat{B} + \zeta I$ [28], where

$$\hat{A} = \begin{bmatrix} S & -0.5I & & & & \\ -1.5I & S & -0.5I & & & \\ & -1.5I & S & -0.5I & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1.5I & S & -0.5I \\ & & & & -1.5I & S \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} S & & & & & \\ & S & & & & \\ & & S & & & \\ & & & \ddots & & \\ & & & & S & \\ & & & & & S \end{bmatrix},$$

$$S = \begin{bmatrix} 4 & -0.5 & & & & \\ -1.5 & 4 & -0.5 & & & \\ & -1.5 & 4 & -0.5 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1.5 & 4 & -0.5 \\ & & & & -1.5 & 4 \end{bmatrix}.$$

In Examples 5.1 and 5.2, we define $q = Mz^* - Nw^*$ with

$$z^* = (0, 1, 0, 1, \dots, 0, 1)^\top, \quad w^* = (1, 0, 1, 0, \dots, 1, 0)^\top.$$

In addition, four sets of values of ξ and ζ are used, i.e. $(\xi, \zeta) = (0, 0)$, $(\xi, \zeta) = (0, 4)$, $(\xi, \zeta) = (4, 0)$ and $(\xi, \zeta) = (-0.5, -0.5)$. When $\xi, \zeta \geq 0$, matrices M and N in Example 5.1 are symmetric positive definite while the corresponding matrices in Example 5.2 are nonsymmetric positive definite and both $\{M, N\}$ in Examples 5.1 and 5.2 have the column \mathcal{W} -property [28]. Hence, for Examples 5.1 and 5.2 with $\xi, \zeta \geq 0$, HLCP (5.1) has a unique solution for any $q \in \mathbb{R}^n$ [37, Theorem 2]. Correspondingly, GAVE (1.1) with $A = M + N$ and $B = M - N$ satisfies Assumption 1.1 and has a unique solution for any $b = q \in \mathbb{R}^n$.

For NSNA, we set $\theta = 0.2$, $\delta = 0.8$, $\mu^{(0)} = 0.01$ and choose

$$\gamma := \min \left\{ \frac{\mu^{(0)}}{\mathcal{C}^{(0)} + 1}, \frac{1}{\mu^{(0)} + 1}, 10^{-12} \right\}$$

such that $\gamma\mathcal{C}^{(0)} < \mu^{(0)}$, $\gamma\mu^{(0)} < 1$ and $\gamma \in (0, 1)$. For JZ-MSNA, we set $\delta = 0.8$, $\sigma = 0.2$, $\mu^{(0)} = 0.01$, $p = 2$ and choose $\beta := \max\{100, 1.01 * (\tau^{(0)})^2 / \mu_0\}$ to satisfy the conditions needed for this algorithm [17] (we refer to [17] for the definition of $\tau^{(0)}$). For TZ-MSNA, as in [40], we set $\sigma = 0.2$, $\delta = 0.8$, $\gamma = 0.001$ and $\mu^{(0)} = 0.01$. For all methods, $x^{(0)} = (2, 2, \dots, 2)^\top$ and methods are stopped if

$$\text{Res} = \|Ax^{(k)} + B|x^{(k)}| - b\| \leq 10^{-7}$$

or the maximum number of iteration step $it_max = 2000$ is exceeded. We use the symbol “—” to denote the latter case.

We first report the results in the case that $\xi, \zeta \geq 0$. For Example 5.1, numerical results are shown in Tables 5.1-5.3, from which we can find that NSNA is better than JZ-MSNA and TZ-MSNA in terms of Iter (the number of iterations) and Cpu (the elapsed CPU time in seconds). Fig. 5.1 plots the convergence curves of the tested methods, from which the monotone convergence properties of all methods are shown¹⁾. For Example 5.2, numerical results are shown in Tables 5.4-5.6, from which we can also find that NSNA is superior to JZ-MSNA and TZ-MSNA in terms of Iter and Cpu. Fig. 5.2 plots the convergence curves of the tested methods, from which the monotone convergence properties of JZ-MSNA and TZ-MSNA are shown and the nonmonotone convergence property of NSNA occurs.

Table 5.1: Numerical results for Example 5.1 with $\xi = \zeta = 0$.

Method		n			
		256	1024	2304	4096
NSNA	Iter	5	5	6	6
	Cpu	0.0044	0.0619	0.3364	0.9985
	Res	1.1392×10^{-14}	5.7894×10^{-14}	2.8847×10^{-13}	1.5603×10^{-12}
JZ-MSNA	Iter	6	6	8	7
	Cpu	0.0052	0.0724	0.4497	1.2020
	Res	2.3488×10^{-11}	4.9355×10^{-10}	1.4026×10^{-11}	9.4826×10^{-11}
TZ-MSNA	Iter	6	6	7	7
	Cpu	0.0049	0.0664	0.3860	1.1631
	Res	1.1147×10^{-14}	2.2960×10^{-14}	3.4131×10^{-14}	4.6847×10^{-14}

Table 5.2: Numerical results for Example 5.1 with $\xi = 0$ and $\zeta = 4$.

Method		n			
		256	1024	2304	4096
NSNA	Iter	5	6	7	7
	Cpu	0.0035	0.0745	0.3975	1.3254
	Res	2.9913×10^{-14}	7.8455×10^{-13}	4.8343×10^{-12}	2.3106×10^{-11}
JZ-MSNA	Iter	7	8	8	9
	Cpu	0.0055	0.1102	0.4676	1.6890
	Res	6.4061×10^{-11}	1.7803×10^{-9}	4.4746×10^{-8}	7.1534×10^{-8}
TZ-MSNA	Iter	6	7	8	8
	Cpu	0.0039	0.0862	0.4599	1.5145
	Res	1.7297×10^{-14}	3.4071×10^{-14}	5.3415×10^{-14}	6.7748×10^{-14}

We then report the results for $(\xi, \zeta) = (-0.5, -0.5)$. Numerical results for Example 5.1 with $n = 16^2$ and Example 5.2 with $n = 24^2$ are reported in Table 5.7 and Fig. 5.3, from which we find that NSNA can find a solution of the GAVE (1.1) in Example 5.1 while JZ-MSNA and TZ-MSNA fail and NSNA is better than JZ-MSNA and TZ-MSNA for solving GAVE (1.1) in Example 5.2. The results further illustrate the superior performance of NSNA.

In conclusion, under our setting, NSNA is a competitive method for solving GAVE (1.1).

¹⁾ For JZ-MSNA and TZ-MSNA, $\|H(z^{(k)})\|$ is defined as in (3.5) with $\phi(a, b) = (|a|^p + |b|^p)^{1/p}$.

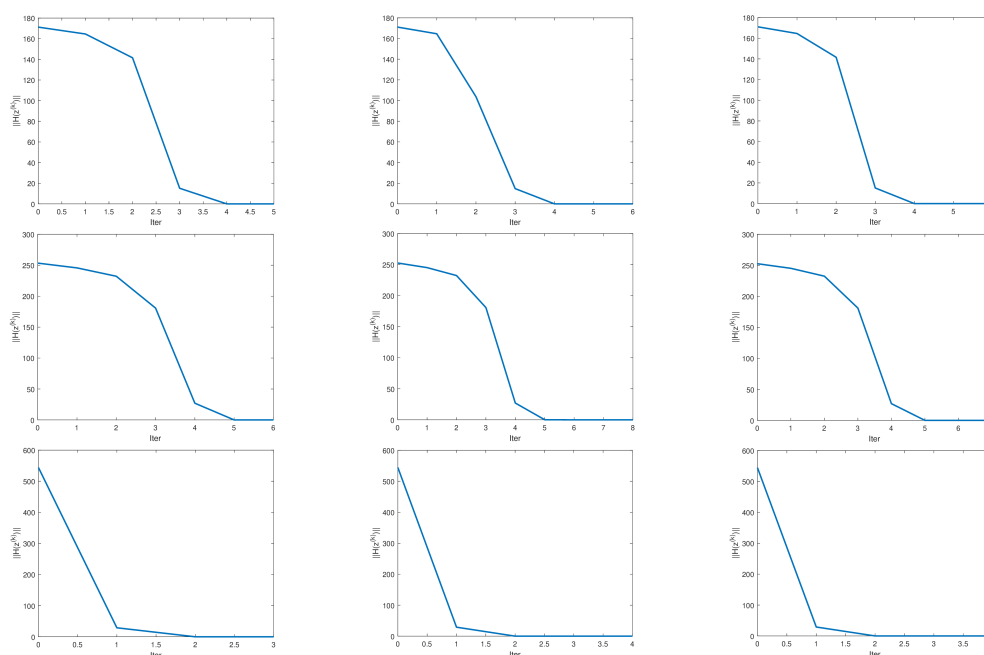


Fig. 5.1. Convergence history curves for Example 5.1 with $n = 32^2$. The plots in the first column are for NSNA, the plots in the second column are for JZ-MSNA and the plots in the third column are for TZ-MSNA, respectively. The plots in the first row are for $\xi = \zeta = 0$, the plots in the second row are for $\xi = 0$ and $\zeta = 4$ and the plots in the third row are for $\xi = 4$ and $\zeta = 0$, respectively.

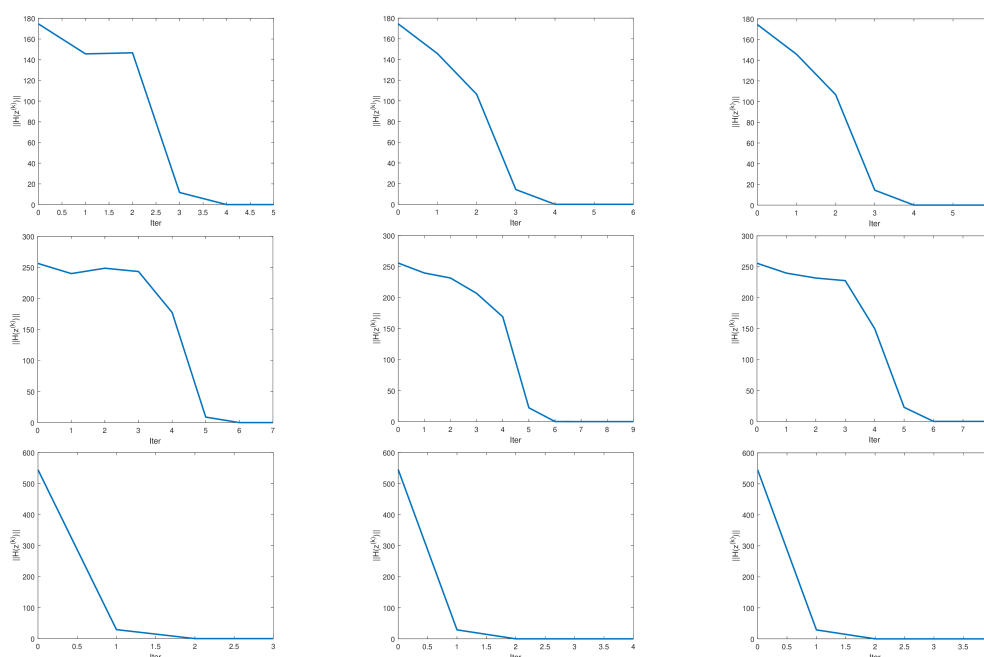


Fig. 5.2. Convergence history curves for Example 5.2 with $n = 32^2$. The plots in the first column are for NSNA, the plots in the second column are for JZ-MSNA and the plots in the third column are for TZ-MSNA, respectively. The plots in the first row are for $\xi = \zeta = 0$, the plots in the second row are for $\xi = 0$ and $\zeta = 4$ and the plots in the third row are for $\xi = 4$ and $\zeta = 0$, respectively.

Table 5.3: Numerical results for Example 5.1 with $\xi = 4$ and $\zeta = 0$.

Method		n			
		256	1024	2304	4096
NSNA	Iter	3	3	3	3
	Cpu	0.0021	0.0361	0.1536	0.5444
	Res	2.0696×10^{-13}	5.8026×10^{-12}	4.2044×10^{-11}	1.7292×10^{-10}
JZ-MSNA	Iter	4	4	5	5
	Cpu	0.0035	0.0477	0.2846	0.9388
	Res	2.5682×10^{-11}	1.1439×10^{-11}	2.1336×10^{-11}	3.0094×10^{-11}
TZ-MSNA	Iter	4	4	4	4
	Cpu	0.0024	0.0458	0.2293	0.7017
	Res	1.7786×10^{-14}	3.4123×10^{-14}	5.2927×10^{-14}	6.8855×10^{-14}

Table 5.4: Numerical results for Example 5.2 with $\xi = \zeta = 0$.

Method		n			
		256	1024	2304	4096
NSNA	Iter	4	5	6	6
	Cpu	0.0031	0.0642	0.3332	1.0440
	Res	1.0500×10^{-14}	5.9914×10^{-14}	2.9611×10^{-13}	1.6331×10^{-12}
JZ-MSNA	Iter	5	6	8	7
	Cpu	0.0036	0.0774	0.4503	1.2214
	Res	4.1569×10^{-11}	2.7899×10^{-10}	1.4829×10^{-11}	6.0501×10^{-9}
TZ-MSNA	Iter	5	6	7	7
	Cpu	0.0035	0.0679	0.3776	1.2130
	Res	1.1200×10^{-14}	2.0773×10^{-14}	3.1480×10^{-14}	4.2238×10^{-14}

Table 5.5: Numerical results for Example 5.2 with $\xi = 0$ and $\zeta = 4$.

Method		n			
		256	1024	2304	4096
NSNA	Iter	6	7	7	8
	Cpu	0.0047	0.0869	0.4074	1.4394
	Res	3.0087×10^{-14}	7.9704×10^{-13}	5.8680×10^{-12}	2.4058×10^{-11}
JZ-MSNA	Iter	8	9	10	11
	Cpu	0.0075	0.1225	0.5793	1.9709
	Res	1.4866×10^{-11}	2.0476×10^{-9}	1.3748×10^{-8}	7.0290×10^{-8}
TZ-MSNA	Iter	7	8	9	9
	Cpu	0.0050	0.0945	0.5193	1.6262
	Res	1.6717×10^{-14}	3.3155×10^{-14}	5.1814×10^{-14}	9.2842×10^{-14}

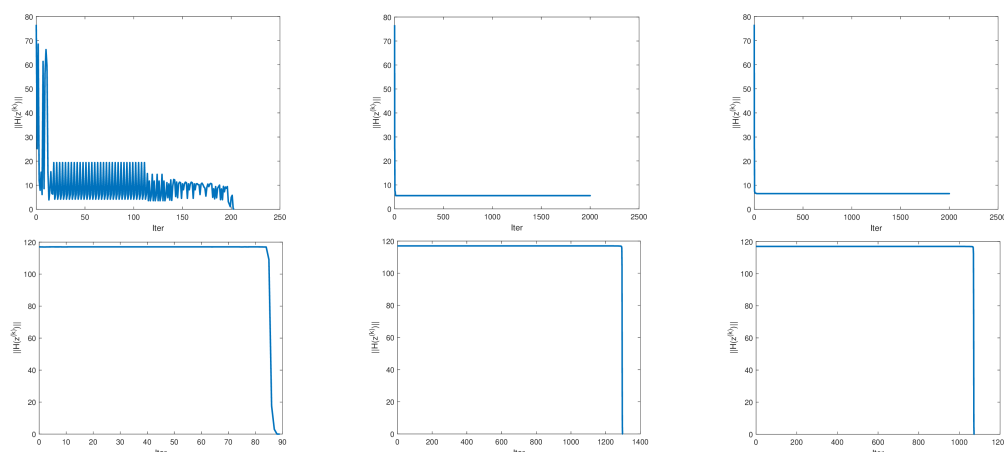


Fig. 5.3. Convergence history curves for Example 5.1 (the above plots) and Example 5.2 (the below plots) with $\xi = \zeta = -0.5$. The plots in the first column are for NSNA, the plots in the second column are for JZ-MSNA and the plots in the third column are for TZ-MSNA, respectively.

Table 5.6: Numerical results for Example 5.2 with $\xi = 4$ and $\zeta = 0$.

Method		n			
		256	1024	2304	4096
NSNA	Iter	3	3	3	3
	Cpu	0.0024	0.0388	0.1623	0.5437
	Res	2.1175×10^{-13}	5.8378×10^{-12}	4.2243×10^{-11}	1.7356×10^{-10}
JZ-MSNA	Iter	4	4	5	5
	Cpu	0.0030	0.0500	0.2979	0.8752
	Res	2.5874×10^{-11}	1.1674×10^{-11}	2.1543×10^{-11}	3.0534×10^{-11}
TZ-MSNA	Iter	4	4	4	4
	Cpu	0.0029	0.0493	0.2448	0.7257
	Res	1.7719×10^{-14}	3.3493×10^{-14}	4.6185×10^{-14}	6.7869×10^{-14}

Table 5.7: Numerical results for $\xi = \zeta = -0.5$.

Example		NSNA	JZ-MSNA	TZ-MSNA
Example 5.1	Iter	203	--	--
	Cpu	0.3295	--	--
	Res	4.3355×10^{-11}	--	--
Example 5.2	Iter	89	1299	1074
	Cpu	1.0028	19.4337	10.0706
	Res	2.0768×10^{-14}	1.0891×10^{-9}	1.7962×10^{-14}

6. Conclusions

In this paper, a non-monotone smoothing Newton method is proposed to solve the system of generalized absolute value equations. Under a weaker assumption, we prove the global and

the local quadratic convergence of our method. Numerical results demonstrate that our method can be superior to two existing methods in some situations.

Acknowledgements. The authors are grateful to the reviewers and the editor for their helpful comments and suggestions that have helped to improve the paper.

C.R. Chen is supported by the Natural Science Foundation of Fujian Province (Grant No. 2021J01661) and by the National Natural Science Foundation of China (Grant No. 11901024). D.M. Yu is supported by the National Natural Science Foundation of China (Grant No. 12201275), by the Ministry of Education in China of Humanities and Social Science Project (Grant No. 21YJCZH204) and by the Liaoning Provincial Department of Education (Grant No. JYTZD2023072). D.R. Han is supported by the National Natural Science Foundation of China (Grant No. 12131004) and by the Ministry of Science and Technology of China (Grant No. 2021YFA1003600). C.F. Ma is supported by the National Key Research and Development Program of China (Grant No. 2019YFC0312003).

References

- [1] M. Achache and N. Hazzam, Solving absolute value equation via complementarity and interior-point methods, *J. Nonlinear Funct. Anal.*, **2018** (2018), 39.
- [2] N. Anane and M. Achache, Preconditioned conjugate gradient methods for absolute value equations, *J. Numer. Anal. Approx. Theory*, **49** (2020), 3–14.
- [3] J.Y. Bello Cruz, O.P. Ferreira, and 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.
- [4] L. Caccetta, B. Qu, and G.L. Zhou, A globally and quadratically convergent method for absolute value equations, *Comput. Optim. Appl.*, **48** (2011), 45–58.
- [5] C.R. Chen, Y.N. Yang, D.M. Yu, and D.R. Han, An inverse-free dynamical system for solving the absolute value equations, *Appl. Numer. Math.*, **168** (2021), 170–181.
- [6] F.H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley, 1983.
- [7] R.W. Cottle, J.S. Pang, and R.E. Stone, *The Linear Complementarity Problem*, SIAM, 2009.
- [8] V. Edalatpour, D. Hezari, and D.K. Salkuyeh, A generalization of the Gauss-Seidel iteration method for solving absolute value equations, *Appl. Math. Comput.*, **293** (2017), 156–167.
- [9] A. Fischer, Solution of monotone complementarity problems with locally Lipschitzian functions, *Math. Program.*, **76** (1997), 513–532.
- [10] X.M. Gu, T.Z. Huang, H.B. Li, S.F. Wang, and L. Li, Two CSCS-based iteration methods for solving absolute value equations, *J. Appl. Anal. Comput.*, **7** (2017), 1336–1356.
- [11] F.K. Haghani, On generalized Traub’s method for absolute value equations, *J. Optim. Theory Appl.*, **166** (2015), 619–625.
- [12] M. Hladík, Bounds for the solution of absolute value equations, *Comput. Optim. Appl.*, **69** (2018), 243–266.
- [13] S.L. Hu and Z.H. Huang, A note on absolute value equations, *Optim. Lett.*, **4** (2010), 417–424.
- [14] S.L. Hu, Z.H. Huang, and J.S. Chen, Properties of a family of generalized NCP-functions and a derivative free algorithm for complementarity problems, *J. Comput. Appl. Math.*, **230** (2009), 69–82.
- [15] S.L. Hu, Z.H. Huang, and Q. Zhang, A generalized Newton method for absolute value equations associated with second order cones, *J. Comput. Appl. Math.*, **235** (2011), 1490–1501.
- [16] Z.H. Huang, S.L. Hu, and J.Y. Han, Convergence of a smoothing algorithm for symmetric cone complementarity problems with a nonmonotone line search, *Sci. China Ser. A-Math.*, **52** (2009), 833–848.

- [17] X.Q. Jiang and Y. Zhang, A smoothing-type algorithm for absolute value equations, *J. Ind. Manag. Optim.*, **9** (2013), 789–798.
- [18] Y.F. Ke, The new iteration algorithm for absolute value equation, *Appl. Math. Lett.*, **99** (2020), 105990.
- [19] Y.F. Ke and C.F. Ma, SOR-like iteration method for solving absolute value equations, *Appl. Math. Comput.*, **311** (2017), 195–202.
- [20] Y.Y. Lian, C.X. Li, and S.L. Wu, Weaker convergent results of the generalized Newton method for the generalized absolute value equations, *J. Comput. Appl. Math.*, **338** (2018), 221–226.
- [21] O.L. Mangasarian, Absolute value programming, *Comput. Optim. Appl.*, **36** (2007), 43–53.
- [22] O.L. Mangasarian, Absolute value equation solution via concave minimization, *Optim. Lett.*, **1** (2007), 3–8.
- [23] O.L. Mangasarian, A generalized Newton method for absolute value equations, *Optim. Lett.*, **3** (2009), 101–108.
- [24] O.L. Mangasarian, Knapsack feasibility as an absolute value equation solvable by successive linear programming, *Optim. Lett.*, **3** (2009), 161–170.
- [25] O.L. Mangasarian and R.R. Meyer, Absolute value equations, *Linear Algebra Appl.*, **419** (2006), 359–367.
- [26] A. Mansoori and M. Erfanian, A dynamic model to solve the absolute value equations, *J. Comput. Appl. Math.*, **333** (2018), 28–35.
- [27] F. Mezzadri, On the solution of general absolute value equations, *Appl. Math. Lett.*, **107** (2020), 106462.
- [28] F. Mezzadri and E. Galligani, Modulus-based matrix splitting methods for horizontal linear complementarity problems, *Numer. Algorithms*, **83** (2020), 201–219.
- [29] X.H. Miao, J.T. Yang, B. Saheya, and J.S. Chen, A smoothing Newton method for absolute value equation associated with second-order cone, *Appl. Numer. Math.*, **120** (2017), 82–96.
- [30] O. Prokopyev, On equivalent reformulations for absolute value equations, *Comput. Optim. Appl.*, **44** (2009), 363–372.
- [31] L.Q. Qi, Convergence analysis of some algorithms for solving nonsmooth equations, *Math. Oper. Res.*, **18** (1993), 227–244.
- [32] L.Q. Qi, D.F. Sun, and G.L. Zhou, A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities, *Math. Program.*, **87** (2000), 1–35.
- [33] J. Rohn, Systems of linear interval equations, *Linear Algebra Appl.*, **126** (1989), 39–78.
- [34] J. Rohn, A theorem of the alternatives for the equation $Ax + B|x| = b$, *Linear Multilinear Algebra*, **52** (2004), 421–426.
- [35] B. Seheya, C.T. Nguyen, and J.S. Chen, Neural network based on systematically generated smoothing functions for absolute value equation, *J. Appl. Math. Comput.*, **61** (2019), 533–558.
- [36] S. Shahsavari and S. Ketabchi, The proximal methods for solving absolute value equation, *Numer. Algebra Control Optim.*, **11** (2021), 449–460.
- [37] R. Sznajder and M.S. Gowda, Generalizations of P_0 - and P -properties; Extended vertical and horizontal linear complementarity problems, *Linear Algebra Appl.*, **223/224** (1995), 695–715.
- [38] J.Y. Tang, A variant nonmonotone smoothing algorithm with improved numerical results for large-scale LWCPs, *Comp. Appl. Math.*, **37** (2018), 3927–3936.
- [39] J.Y. Tang and H.C. Zhang, A nonmonotone smoothing Newton algorithm for weighted complementarity problem, *J. Optim. Theory Appl.*, **189** (2021), 679–715.
- [40] J.Y. Tang and J.C. Zhou, A quadratically convergent descent method for the absolute value equation $Ax + B|x| = b$, *Oper. Res. Lett.*, **47** (2019), 229–234.
- [41] J.Y. Tang and J.C. Zhou, Quadratic convergence analysis of a nonmonotone Levenberg-Marquardt type method for the weighted nonlinear complementarity problem, *Comput. Optim. Appl.*, **80** (2021), 213–244.

- [42] A. Wang, Y. Cao, and J.X. Chen, Modified Newton-type iteration methods for generalized absolute value equations, *J. Optim. Theory Appl.*, **181** (2019), 216–230.
- [43] S.L. Wu and C.X. Li, The unique solution of the absolute value equations, *Appl. Math. Lett.*, **76** (2018), 195–200.
- [44] S.L. Wu and C.X. Li, A note on unique solvability of the absolute value equation, *Optim. Lett.*, **14** (2020), 1957–1960.
- [45] S.L. Wu and S.Q. Shen, On the unique solution of the generalized absolute value equation, *Optim. Lett.*, **15** (2021), 2017–2024.
- [46] D.M. Yu, C.R. Chen, and D.R. Han, A modified fixed point iteration method for solving the system of absolute value equations, *Optimization*, **71** (2022), 449–461.
- [47] N. Zainali and T. Lotfi, On developing a stable and quadratic convergent method for solving absolute value equation, *J. Comput. Appl. Math.*, **330** (2018), 742–747.
- [48] M. Zamani and M. Hladík, A new concave minimization algorithm for the absolute value equation solution, *Optim. Lett.*, **15** (2021), 2241–2254.
- [49] C. Zhang and Q.J. Wei, Global and finite convergence of a generalized Newton method for absolute value equations, *J. Optim. Theory Appl.*, **143** (2009), 391–403.
- [50] H.C. Zhang and W.W. Hager, A nonmonotone line search technique and its application to unconstrained optimization, *SIAM J. Optim.*, **14** (2004), 1043–1056.
- [51] 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** (2021), 113578.
- [52] J.G. Zhu, H.W. Liu, and C.H. Liu, A family of new smoothing functions and a nonmonotone smoothing Newton method for the nonlinear complementarity problems, *J. Appl. Math. Comput.*, **37** (2011), 647–662.