Journal of Mathematical Modeling
Vol. 12, No. 3, 2024, pp. 463-479. Research Article

JMM

Introducing three new smoothing functions: Analysis on smoothing-Newton algorithms

Nurullah Yilmaz*

Department of Mathematics, Suleyman Demirel University, Isparta, Turkey
Email(s): nurullahyilmaz@sdu.edu.tr

Abstract. In this paper, we focus on solving the system of absolute value equations (AVE), which is one of the most popular classes of nonlinear equations. First, a new smoothing technique with three different smoothing functions is introduced, and the AVE is transformed into a family of parametrized smooth equations with the help of these smoothing functions. Then, a smoothing Newton-type algorithm with hybridized inexact line search is developed based on the proposed smoothing technique. The numerical experiments have been carried out on some well-known and randomly generated test problems, and the results are analyzed in terms of line search techniques. The numerical results show that the proposed hybrid approach is more efficient than the other algorithms.

Keywords: Absolute value equation, smoothing function, Newton-type algorithm.
AMS Subject Classification 2010: 65K10, 26D07, 90C30.

1 Introduction

Let us consider the following AVE of the form:

$$Ax + B|x| = c, \tag{1}$$

where $A, B \in \mathbb{R}^{n\times n}$, $B \neq 0$, $c \in \mathbb{R}^n$ and $|x| := (|x_1|, |x_2|, \ldots, |x_n|)$. The AVE in (1) is one of the important sub-class of system of nonlinear equations. Besides being nonlinear, the AVE in (1) is nonsmooth due to the presence of the absolute value term.

The AVE was first introduced by Rohn in [32] as a generalization of the following system of equations:

$$Ax - |x| = c, \tag{2}$$

which is the subject of many interesting research papers [10, 17, 24, 37]. The main motivation of all research on AVE of the form (2) stems from the equivalence between AVE and linear complementarity

*Corresponding author

Received: 22 February 2024 / Revised: 15 March 2024 / Accepted: 22 March 2024
DOI: 10.22124/jmm.2024.26827.2361

problems (LCP). Moreover, the relations between AVE of the form (2) and mixed integer programming, bimatrix games and interval linear equations have increased the popularity of the problem [15, 22, 24].

Two main research directions have been adopted for AVE as theoretical and numerical. In theoretical studies, it is concentrated on finding the conditions for the existence and uniqueness of the solutions [21, 29]. In addition, various generalizations of AVE have been introduced such as new generalization of AVE (NGAVE) [20, 39], non-Lipschitz generalization of AVE (NAVE) [41, 43], AVE associated with circular cone (CCAVE) [26], AVE associated with second-order cone (SOCAVE) [27, 34] due to application areas especially engineering sciences [28]. The relation between generalizations of AVE and optimization problems (LCP and others) is theoretically investigated. The sufficient conditions for solvability and nonsolvability of generalizations of AVE with unique and multiple solutions have been discussed. We also recall the following results about the existence and uniqueness of the solution of AVE of type (1).

Theorem 1 ([17, 34]). *If the minimal singular values of the matrix A is strictly greater than the maximal singular value of the matrix B, then AVE of type* (1) *is uniquely solvable for any $b \in \mathbb{R}^n$.*

From numerical point of view, various multi-step algorithms have been proposed to solve AVE of the forms (1) and (2) such as generalized Newton algorithms [22, 23], Picard iteration [18], fixed point iteration [3], Levenberg-Marquardt algorithm [16] and others [1, 10, 19]. In [22], generalized Newton method and in [23] a hybrid algorithm is proposed by Mangasarian. A quadratically convergent descend method to solve AVE of type (1) is presented in [36] and a dynamic model to solve AVE is proposed in [25]. Also, recently several algorithms have been proposed to solve AVE [2, 4, 14].

Among the numerical techniques, the smoothing Newton algorithms received considerable attention. Smoothing can be briefly expressed as approximating the nonsmooth function by a family of parametrized smooth functions. Mathematically, the smoothing function is defined as follows:

Definition 1 ([30]). *Let $H : \mathbb{R}^n \to \mathbb{R}^n$ be a locally Lipschitz continuous function.*

i. The function $\tilde{H} : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}$ is called a smoothing function of H, if $\tilde{H}(\cdot, \tau)$ is continuously differentiable in $\mathbb{R}^n$ for any fixed τ, and for any $x \in \mathbb{R}^n$,

$$\|H(x) - \tilde{H}(z, \tau)\| \to 0, \qquad \text{as } \tau \downarrow 0 \text{ and } z \to x.$$

ii. $\tilde{H}(x, \tau)$ is said to approximate H at x superlinearly if, for any $y \to x$ and $\tau \downarrow 0$, we have

$$\tilde{H}(x, \tau) - H(x) - \tilde{H}'(x, \tau)(y - x) = o(\|y - x\|) + O(\tau).$$

iii. $\tilde{H}(x, \tau)$ is said to approximate H at x quadratically if, for any $y \to x$ and $\tau \downarrow 0$, we have

$$\tilde{H}(x, \tau) - H(x) - \tilde{H}'(x, \tau)(y - x) = O(\|y - x\|^2) + O(\tau).$$

Notation: $\mathbb{R}^+$ denotes positive real numbers, $\|\cdot\|$ denotes the Euclidean norm and x^* denotes the optimal solution. Also $D \subset \mathbb{R}^n$, we define the complement of D as D^c.

Many interesting smoothing techniques have been proposed to solve nonsmooth problems in the literature [6, 9, 40, 44]. They are one of the important tools in solving nonsmooth problems such as image processing in [8], exact penalty formulations of constrained optimization problems [12, 31], complementarity problems [11], in smoothing process of piecewise smooth functions [35, 42]. The smoothing approaches have been used for solving AVE and smoothing Newton method was first proposed to solve AVE by Caccetta et al. [7]. The smoothing approaches increase its popularity in solving AVE [1,17]. Applications and numerical comparisons of different smoothing approaches are also presented in [13,33,34,38]. It is observed from all of these studies the smoothing techniques are effective tools for solving AVEs. New combinations of smoothing techniques with Newton-like algorithms contribute to be constituted promising, fast and robust methods for solving problem of AVEs.

In this study, we propose a new class of smoothing technique with three different types of smoothing functions which are constructed by using S-shaped functions. Based on the smoothing techniques, a Newton-type algorithm hybridized line search with Armijo-type technique is developed. Finally, we demonstrate the implementation and efficiency of the algorithm on some numerical examples, and compare the the numerical results with the existing algorithms.

The remaining parts of the paper are organized as follows: In Section 2, we propose new smoothing approximations with error estimates and introduce a new Newton type algorithm with hybridized line search technique. In Section 3, the algorithm is applied on some test problems and obtained results are compared with the other Newton type algorithms. Finally, the concluding remarks are given.

2 Main Results

2.1 Smoothing Techniques

Let us define $H : \mathbb{R}^n \to \mathbb{R}^n$ as

$$H(x) = Ax + B|x| - c. \tag{3}$$

Now, we aim to solve the problem $H(x) = 0$. Since the system of equations $H(x) = 0$ includes absolute value term, Jacobian-based methods cannot be used to solve it. Therefore, we plan to construct new smoothing approaches for absolute value function and by using these smoothing approaches, we obtain a family of smooth approximation.

The function $\varphi(t) = |t|$ is expressed as

$$\varphi(t) = \begin{cases} t, & t \geq 0, \\ -t, & t < 0, \end{cases}$$

or equivalently as

$$\varphi(t) = t\phi(t), \tag{4}$$

where

$$\phi(t) = \begin{cases} -1, & t \leq 0, \\ 1, & t > 0. \end{cases}$$

For any $\tau > 0$, the smoothing function of φ in (4) is defined by

$$\tilde{\varphi}_i(t,\tau) = t\tilde{\phi}_i(t,\tau), \tag{5}$$

where $\tilde{\phi}_i(t,\tau)$ is designed by considering $S-$shaped functions

$$\tilde{\phi}_1(t,\tau) = \frac{t}{\sqrt{\tau^2 + t^2}},$$

and

$$\tilde{\phi}_2(t,\tau) = \tanh\left(\frac{t}{\tau}\right).$$

Another alternative is proposed in [35] as

$$\tilde{\phi}_3(t,\tau) = \begin{cases} -1, & t \le -\tau, \\ S(t,\tau), & -\tau \le t \le \tau \\ 1, & t > \tau, \end{cases}$$

where $S(t,\tau) = \frac{-1}{2\tau^3}t^3 + \frac{3}{2\tau}t$. Useful properties of smoothing functions are presented in the following:

Proposition 1. *Let $\tilde{\varphi}_i : \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}$ be defined as in (5) for $i = 1,2,3$. Then, for any $\tau > 0$ and for $i = 1,2,3$ we have*

(i) *$\tilde{\varphi}_i$ is continuously differentiable at $(t,\tau) \in \mathbb{R} \times \mathbb{R}^+$,*

(ii) $0 \le \varphi(t) - \tilde{\varphi}_i(t,\tau) \le \tau$,

(iii) $\lim_{\tau \downarrow 0} \tilde{\varphi}_i(t,\tau) = \varphi(t)$ *for any $t \in \mathbb{R}$,*

(iv) *$\tilde{\varphi}_i(t,\varepsilon)$ approximates $\varphi(t)$ at x quadratically,*

Proof. (i) For $i = 1$, we have

$$\frac{\partial \tilde{\varphi}_1(t,\tau)}{\partial t} = \frac{t^3 + 2t\tau^2}{(t^2+\tau^2)^{\frac{3}{2}}}, \tag{6}$$

and

$$\frac{\partial \tilde{\varphi}_1(t,\tau)}{\partial \tau} = \frac{-\tau t^2}{(t^2+\tau^2)^{\frac{3}{2}}}. \tag{7}$$

For $i = 2$, the partial derivatives are computed as

$$\frac{\partial \tilde{\varphi}_2(t,\tau)}{\partial t} = \frac{1}{\tau}\left(-t\tanh^2\left(\frac{t}{\tau}\right) + \tau\tanh\left(\frac{t}{\tau}\right) + t\right), \tag{8}$$

and

$$\frac{\partial \tilde{\varphi}_2(t,\tau)}{\partial \tau} = \frac{t^2}{\tau^2}\left(\tanh^2\left(\frac{t}{\tau}\right) - 1\right). \tag{9}$$

Finally, partial derivatives of $\tilde{\varphi}_3(t,\tau)$ are computed as

$$\frac{\partial \tilde{\varphi}_3(t,\tau)}{\partial t} = \begin{cases} -1, & t < -\tau, \\ \frac{\partial S(t,\tau)}{\partial t}, & -\tau \le t \le \tau, \\ 1, & t > \tau, \end{cases} \tag{10}$$

and

$$\frac{\partial \tilde{\varphi}_3(t,\tau)}{\partial \tau} = \begin{cases} 0, & t < -\tau, \\ \frac{\partial (tS(t,\tau))}{\partial \tau}, & -\tau \le t \le \tau, \\ 0, & t > \tau, \end{cases} \tag{11}$$

where $\frac{\partial (tS(t,\tau))}{\partial t} = \frac{-2}{\tau^3}t^3 + \frac{3}{\tau}t$ and $\frac{\partial (tS(t,\tau))}{\partial \tau} = \frac{3}{2\tau^4}t^4 - \frac{3}{2\tau^2}t^2$. It can be seen from the Eqs. (6)-(11) that the functions $\tilde{\varphi}_i(t,\tau)$ are continuously differentiable for all $i = 1,2,3$.

(ii) Let us start with $i = 1$. For any $\tau > 0$, we first consider the case $t \geq 0$, then we have

$$\begin{aligned}\varphi(t) - \tilde{\varphi}_1(t,\tau) &= t - \frac{t^2}{\sqrt{\tau^2+t^2}} \\ &= \frac{t\sqrt{\tau^2+t^2} - t^2}{\sqrt{\tau^2+t^2}} \\ &\leq \tau.\end{aligned}$$

For the case $t < 0$, we obtain

$$\begin{aligned}\varphi(t) - \tilde{\varphi}_1(t,\tau) &= -t - \frac{t^2}{\sqrt{\tau^2+t^2}} \\ &= \frac{-t\sqrt{\tau^2+t^2} - t^2}{\sqrt{\tau^2+t^2}} \\ &\leq \tau.\end{aligned}$$

Let $i = 2$ and consider the case $t \geq 0$, then we have

$$\begin{aligned}\varphi(t) - \tilde{\varphi}_2(t,\tau) &= t - t\frac{e^{\frac{2t}{\tau}} - 1}{e^{\frac{2t}{\tau}} + 1} \\ &= \frac{2t}{e^{\frac{2t}{\tau}} + 1} \\ &\leq \tau\end{aligned}$$

and for the case $t < 0$, we obtain

$$\begin{aligned}\varphi(t) - \tilde{\varphi}_2(t,\tau) &= -t - t\frac{e^{\frac{2t}{\tau}} - 1}{e^{\frac{2t}{\tau}} + 1} \\ &= \frac{-2te^{\frac{2t}{\tau}}}{e^{\frac{2t}{\tau}} + 1} \\ &\leq \tau.\end{aligned}$$

For $i = 3$, it is sufficient to investigate the case $-\tau \leq t \leq \tau$ since $\varphi(t) = \tilde{\varphi}_3(t,\tau)$ outside of the interval $[-\tau,\tau]$. Therefore, we obtain

$$\begin{aligned}\varphi(t) - \tilde{\varphi}_3(t,\tau) &= |t| - tS(t,\tau) \\ &\leq \tau.\end{aligned}$$

(iii) Since $0 \leq \varphi(t) - \tilde{\varphi}_i(t,\tau) \leq \tau$ for $i = 1,2,3$, we obtain the desired result. Moreover, we have

$$\lim_{\tau\to 0} \frac{\partial \tilde{\varphi}_i(t,\tau)}{\partial t} = \begin{cases} -1, & t < 0, \\ 1, & t > 0. \end{cases} \tag{12}$$

for $i = 1,2,3$.

(iv) For any $\tau > 0$, we have to show that the following equality

$$\tilde{\varphi}_i(y,\tau) - \varphi(t) - \frac{\partial \tilde{\varphi}_i(y,\tau)}{\partial y}(y-t) = O(|y-t|^2) + O(\tau) \tag{13}$$

holds for $i = 1,2,3$. We first consider the smoothing function $\tilde{\varphi}_1(t,\tau)$ with the following two cases:

Case 1. Let $t = 0$, then we have

$$\begin{aligned}\frac{y^2}{(y^2+\tau^2)^{\frac{1}{2}}} - \sqrt{t^2} - \frac{y^3+2y\tau^2}{(y^2+\tau^2)^{\frac{3}{2}}}(y-t) &= \frac{y^2}{(y^2+\tau^2)^{\frac{1}{2}}} - \frac{y^3+2y\tau^2}{(y^2+\tau^2)^{\frac{3}{2}}}y \\ &= O(\tau) \\ &= O(|y-t|^2) + O(\tau).\end{aligned}$$

Case 2. Let $t \neq 0$ then we have

$$\begin{aligned}&\frac{y^2}{(y^2+\tau^2)^{\frac{1}{2}}} - \sqrt{t^2} - \frac{y^3+2y\tau^2}{(y^2+\tau^2)^{\frac{3}{2}}}(y-t) \\ &\qquad = \frac{y^2}{(y^2+\tau^2)^{\frac{1}{2}}} - \sqrt{t^2} - \frac{y^3+2y\tau^2}{(y^2+\tau^2)^{\frac{3}{2}}}(y-t) + \frac{\sqrt{t^2}}{(y^2+\tau^2)^{\frac{1}{2}}} - \frac{\sqrt{t^2}}{(y^2+\tau^2)^{\frac{1}{2}}} \\ &\qquad = -\frac{(y-t)^2\tau^2}{(y^2+\tau^2)^{\frac{3}{2}}} + \frac{y^3t - (y^2+\tau^2)^{\frac{3}{2}}\sqrt{t^2} + t^2\tau^2}{(y^2+\tau^2)^{\frac{3}{2}}} \\ &\qquad = O(|y-t|^2) + O(\tau).\end{aligned}$$

The proofs for $i = 2$ and $i = 3$ can be obtained similarly.

□

By replacing the smoothing functions $\tilde{\varphi}_i(t)$ with the each component of $|x|$ for $i = 1,2,3$, the smooth approximation of $|x|$ is obtained. Therefore, corresponding smoothed version of $H(x) = 0$ in (3) is defined by:

$$\tilde{H}_i(x,\tau) = \begin{bmatrix} Ax + B\tilde{\Phi}_i(x,\tau) - c \\ \tau \end{bmatrix} = 0, \tag{14}$$

where $\tilde{\Phi}_i(x,\tau) = (\tilde{\varphi}_i(x_1,\tau), \tilde{\varphi}_i(x_2,\tau), \ldots, \tilde{\varphi}_i(x_n,\tau))$ and $\tau > 0$. After smoothing process, it is permitted to use Newton type algorithms to solve the system of equations of the form $\tilde{H}_i(x,\tau) = 0$.

Proposition 2. *For any $\tau > 0$, the Jacobian of $H(x,\tau)$ at $x \in \mathbb{R}^n$ is*

$$\tilde{H}_i'(x,\tau) = \begin{bmatrix} A + B\nabla_x\tilde{\Phi}_i(x,\tau) & B\nabla_\tau\tilde{\Phi}_i(x,\tau) \\ 0 & 1 \end{bmatrix}, \tag{15}$$

where

$$\nabla_x\tilde{\Phi}_i(x,\tau) = \begin{bmatrix} \frac{\partial\varphi_i(x_1,\tau)}{\partial x_1} & 0 & \cdots & 0 \\ 0 & \frac{\partial\varphi_i(x_2,\tau)}{\partial x_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{\partial\varphi_i(x_n,\tau)}{\partial x_n} \end{bmatrix},$$

and

$$\nabla_\tau \Phi_i(x,\tau) = \begin{bmatrix} \frac{\partial \varphi_i(x_1,\tau)}{\partial \tau} \\ \frac{\partial \varphi_i(x_2,\tau)}{\partial \tau} \\ \vdots \\ \frac{\partial \varphi_i(x_n,\tau)}{\partial \tau} \end{bmatrix},$$

for $i = 1,2,3$.

Theorem 2. *Let the functions* $H(x)$ *and* $\tilde{H}_i(x,\tau)$ *be defined as* (3) *and* (14)*, respectively. Then, we have*

$$\|H(x) - \tilde{H}_i(x,\tau)\| \leq \sigma_{\max}(B)\sqrt{n}\tau,$$

where $\sigma_{\max}(B)$ *represents the maximal singular value of the matrix* B*.*

Proof. For any $\tau > 0$,

$$\begin{aligned} \|H(x) - \tilde{H}_i(x,\tau)\|^2 &= \|B|x| - B\tilde{\Phi}_i(x,\tau)\|^2 \\ &= \|B\left(|x| - \tilde{\varphi}_i(x,\tau)\right)\|^2 \\ &\leq (\sigma_{\max}(B))^2 \sum_{j=1}^{n} \left||x_j| - \tilde{\varphi}_i(x_j,\tau)\right|^2 \\ &\leq (\sigma_{\max}(B))^2 n\tau^2, \end{aligned}$$

for $i = 1,2,3$. This completes the proof. □

Theorem 3. *Let the functions* $H(x)$ *and* $\tilde{H}_i(x,\tau)$ *be defined as* (3) *and* (14)*, respectively.* $\tilde{H}_i(x,\tau)$ *approximates* $H(x)$ *at* x *quadratically.*

Proof. For any $\tau > 0$, we have to show that the equality

$$\tilde{H}_i(y,\tau) - H(x) - \frac{\partial \tilde{H}_i(y,\tau)}{\partial y}(y-x) = O(|y-x|^2) + O(\tau), \tag{16}$$

holds for $i = 1,2,3$. From the Proposition 1 (iv), it is easy to see that each component of (16) holds the above property. Then, the desired result is obtained. □

Theorem 4 ([17,34])**.** *If the minimal singular values of the matrix* A *is strictly greater than the maximal singular value of the matrix* B*, then* $\tilde{H}_i'(x,\tau)$ *is nonsingular.*

Theorem 5. *Suppose that* x^* *is a solution* $H(x) = 0$ *in* (3) *and* $\bar{x}$ *is a solution of* $\tilde{H}_i(x,\tau) = 0$ *in* (14)*. Then,*

$$\|H(x^*) - \tilde{H}_i(\bar{x},\tau)\| \leq \sigma_{\max}(B)\sqrt{n}\tau,$$

for $i = 1,2,3$*. Moreover, we have* $\tilde{H}_i(\bar{x},\tau) \to H(x^*)$ *for* $\tau \to 0$*.*

Proof. It is easy to see that $H(x^*) = \tilde{H}_i(\bar{x},\tau) = 0$ and $\|H(\bar{x})\| \geq 0$. By considering Theorem 2, we obtain

$$\begin{aligned} \|H(x^*) - \tilde{H}_i(\bar{x},\tau)\|^2 &\leq \|H(\bar{x}) - \tilde{H}_i(\bar{x},\tau)\|^2 \\ &\leq (\sigma_{\max}(B))^2 n\tau^2, \end{aligned}$$

for any $\tau > 0$ and for $i = 1,2,3$. □

2.2 Algorithm

In this section, we present the algorithm to solve problem (1). First of all, let us define the function as

$$G(x) = \frac{1}{2}\|H(x)\|^2,$$

and smoothing function $\tilde{G}_i(x,\tau) : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$ as

$$\tilde{G}_i(x,\tau) = \frac{1}{2}\|\tilde{H}_i(x,\tau)\|^2,$$

for $i = 1,2,3$.

Assumption 1. *The maximal singular value of the matrix B is strictly lower than the minimal singular value of the matrix A.*

Under the above condition, it is proved in [17] that the AVE of the form (1) is uniquely solvable for any $c \in \mathbb{R}^n$.

Smoothing Newton Algorithm (SNA)

Step 0 Select $\sigma,\beta,\delta \in (0,1)$, $\tau^0 > 0$ and denote $e^0 := (0,1) \in \mathbb{R}^n \times \mathbb{R}$. Let $\beta_0 = \min\{\beta,\tau^0\}$ and $w^0 := (x^0,\tau^0)$ such that $x^0 \in \mathbb{R}^n$. Set $k := 0$.

Step 1 If $\|\tilde{H}_i(\omega^k)\| = 0$, then stop otherwise go to Step 2.

Step 2 Find the search direction $d^k := (d_{x^k}, d_{\tau^k}) \in \mathbb{R}^n \times \mathbb{R}$ by

$$\tilde{H}_i(\omega^k) + \tilde{H}_i'(\omega^k)d^k = \beta_k\tau_k e^0. \tag{17}$$

Step 3 Compute the step size $\alpha_k := \max\{\delta^m : m = 1,2,\dots\}$ such that

$$\|\tilde{H}_i(\omega^k + \alpha_k d^k)\| \le \max\Big\{[1-\sigma(1-\beta_k)\alpha_k]\,\|\tilde{H}_i(\omega^k)\|, \\ \left(\|\tilde{H}_i(\omega^k)\|^2 + 2\sigma\alpha_k\tilde{H}_i(\omega^k)^T\tilde{H}_i'(\omega^k)d^k\right)^{\frac{1}{2}}\Big\}. \tag{18}$$

Step 4 Set $\omega^{k+1} = \omega^k + \alpha_k d^k$ and $k = k+1$ and go to Step 1.

The solvability of equation (17) can be proved as in [17] by Theorem 3.2. Thus, we deal with the convergence of the SNA. The line search process of SNA is designed inspiring from [43,45] and Armijo rule [5] which is the following.

Theorem 6. *Let the sequence $\omega^k = (x^k,\tau^k)$ be generated by SNA iteratively and Assumption 1 is satisfied. Then, the sequence $\{\omega^k\}$ is bounded and the accumulation point ω^* is a solution of (1) with $\tilde{H}_i(\omega^*) = 0$ for $i = 1,2,3$.*

Proof. The boundedness of $\{\omega^k\}$ is obtained by considering Lemma 4.1 in [17]. Let us define the first part of the right hand side of the inequality (18) as $m(k) = [1-\sigma(1-\beta_k)\alpha_k]\,\|\tilde{H}_i(\omega^k)\|$ and the second part as $n(k) = \left(\|\tilde{H}_i(\omega^k)\|^2 + 2\sigma\alpha_k\tilde{H}_i(\omega^k)^T\tilde{H}_i'(\omega^k)d^k\right)^{\frac{1}{2}}$ for $k = 1,2,\ldots$. Without loss of generality let us assume that $m(k) \geq n(k)$ for any $k \in N$. Then the inequality in (18) transforms into the following form:

$$\|\tilde{H}_i(\omega^k + \alpha_k d^k)\| \leq [1-\sigma(1-\beta_k)\alpha_k]\,\|\tilde{H}_i(\omega^k)\|. \tag{19}$$

At this stage, we first show that $\tau^k \to 0$. Let $\lim_{k\to\infty}\tau_k = \hat{\tau}$. If $\hat{\tau} = 0$, then the desired result is obtained. Assume that $\hat{\tau} > 0$, then we have $\tau^0 \geq \tau^k \geq \hat{\tau} > 0$. Since the iteration sequence $\left\{\omega^k\right\}$ is bounded and it has at least one accumulation point $\omega^* = (x^*,\tau^*)$ with $\tau^* = \hat{\tau} > 0$, then we obtain $\|\tilde{H}_i(\omega^*)\| \geq \tau^* > 0$. Suppose that $\omega^k = (x^k,\tau^k) \to \omega^* = (x^*,\tau^*)$. Then, it follows from (17) and Step 4 that

$$\tau^{k+1} = \tau^k + \alpha_k d_{\tau^k} < [1-(1-\beta_k)\alpha_k]\,\tau^k,$$

which implies that $\lim_{k\to\infty}\alpha_k = 0$ by $0<\sigma<1$ and $\tau^* > 0$. Let $\mu_k = \frac{\alpha_k}{\delta}$, then it follows from (19) that

$$\|\tilde{H}_i(\omega^{k+1} + \mu_k d^k)\| > [1-\sigma(1-\beta_k)\mu_k]\,\|\tilde{H}_i(\omega^k)\|, \tag{20}$$

and we have

$$\frac{\|\tilde{H}_i(\omega^k + \mu_k d^k)\| - \|\tilde{H}_i(\omega^k)\|}{\mu_k} > -\sigma(1-\beta_k)\|\tilde{H}_i(\omega^k)\|. \tag{21}$$

Since $\|H(\omega^*)\| > 0$, by taking the limit of (21) as $k\to\infty$ we have

$$\tilde{H}_i(\omega^*)^T\tilde{H}_i'(\omega^*)d^* \geq -\sigma(1-\beta_*)\|\tilde{H}_i(\omega^*)\|^2, \tag{22}$$

where $\beta_* = \min\{\beta,\tau^*\}$. On the other hand, by considering (17) we obtain

$$\begin{aligned}\tilde{H}_i(\omega^*)^T\tilde{H}_i'(\omega^*)d^* &= -\,\|\tilde{H}_i(\omega^*)\|^2 + \beta_*\tau^*\|\tilde{H}_i(\omega^*)\|\tilde{H}_i(\omega^*)^T e^0 \\ &\leq -\,\|\tilde{H}_i(\omega^*)\|^2 + \beta_*\tau^*\|\tilde{H}_i(\omega^*)\|^2.\end{aligned}$$

Therefore, we have

$$\tilde{H}_i(\omega^*)^T\tilde{H}_i'(\omega^*)d^* \leq (-1+\beta_*)\,\|\tilde{H}_i(\omega^*)\|^2. \tag{23}$$

By considering the inequalities in (22) and (23), we have $-1+\beta_* \geq -\sigma\,(1-\beta_*)$ which contradicts with $\sigma < 1$.

Now we are ready to show that $\tilde{H}_i(\omega^*) = 0$. Assume to contrary that $\tilde{H}_i(\omega^*) \neq 0$. Then, we have $\|\tilde{H}_i(\omega^*)\| > 0$. We know from the previous stage of this proof that $\tau^* = 0$ and $\beta_k \to 0$ as $k\to\infty$. The inequality (18) implies that $\lim_{k\to\infty}\alpha_k = 0$ by $0<\sigma<1$ and $\|\tilde{H}_i(\omega^*)\| > 0$. Let $\mu_k = \frac{\alpha_k}{\delta}$ by considering (19)

$$\|\tilde{H}_i(\omega^{k+1} + \mu_k d^k)\| > [1-\sigma(1-\beta_k)\mu_k]\,\|\tilde{H}_i(\omega^k)\|, \tag{24}$$

and taking the limit in (24), we have

$$\tilde{H}_i(\omega^*)^T\tilde{H}_i'(\omega^*)d^* \geq -\sigma\|\tilde{H}_i(\omega^*)\|^2. \tag{25}$$

On the other hand, by considering (17) we obtain

$$\tilde{H}_i(\omega^*)^T\tilde{H}_i'(\omega^*)d^* = -\|\tilde{H}_i(\omega^*)\|^2. \tag{26}$$

From (25) and (26), the following inequality

$$-\sigma\|\tilde{H}_i(\omega^*)\|^2 \le -\|\tilde{H}_i(\omega^*)\|^2, \tag{27}$$

is obtained. The inequality (27) implies that, since $\|\tilde{H}_i(\omega^*)\| > 0$, which contradicts with $\sigma < 1$. Now we complete the first part of the proof. Let $m(k) < n(k)$, then (18) transforms into the following form:

$$\|\tilde{H}_i(\omega^k + \alpha_k d^k)\| \le \left(\|\tilde{H}_i(\omega^k)\|^2 + 2\sigma\alpha_k \tilde{H}_i(\omega^k)^T \tilde{H}_i'(\omega^k) d^k\right)^{\frac{1}{2}}. \tag{28}$$

The inequality (28) is derived from Armijo rule

$$\tilde{G}_i(\omega^k + \alpha_k d^k) - \tilde{G}_i(\omega^k) \le \sigma\alpha_k \nabla\tilde{G}_i(\omega^k)^T d^k. \tag{29}$$

The proof of inequality (29) can be obtained by using the same way at second part of the proof of the Theorem 3.5 in [43]. □

Theorem 7. *Let w^* be any accumulation point of the iteration sequence $\{w^k\}$ generated by SNA. Then $\{w^k\}$ converges to w^* quadratically.*

Theorem 8. *Let the sequence ω^k be generated by SNA iteratively and Assumption 1 is satisfied. Then, the sequence $\{\omega^k\}$ converges to the unique solution of the AVE (1) quadratically.*

Proof. The proof is obtained similar to the proof of Theorem 4.1 in [45] and Theorem 3.7 in [7]. □

Remark 1. *According to Theorems 6,7 and 8, under the Assumption 1, it is seen that the smoothing Newton method is well defined and the generated sequence $\{\omega_k\}$ globally and quadratically converges to the unique solution of the AVE (1).*

3 Numerical Examples

In this section, we consider several numerical examples in order to show the effectiveness of our methods. The proposed algorithm is programmed in MATLAB 2016A and has been implemented on Intel Core i5-3337U 1.8GHz with 8 Gb RAM. The proposed algorithm applied to the following problems:

Problem 1 ([33])**.** Consider AVE of the form $Ax + B|x| = c$, where

$$A = \begin{pmatrix} 10 & 1 & 2 & 0 \\ 1 & 11 & 3 & 1 \\ 0 & 2 & 12 & 1 \\ 1 & 7 & 0 & 13 \end{pmatrix}, \quad c = \begin{pmatrix} 12 \\ 15 \\ 14 \\ 20 \end{pmatrix},$$

and $B = -I$. The exact solution for this problem is $x^* = (1,1,1,1)$.

Problem 2 ([33])**.** Consider the AVE of the form $Ax + B|x| = c$, where

$$A = \begin{pmatrix} 2 & -3 & 6 & -12 \\ 0 & 2 & -3 & 6 \\ 0 & 0 & 2 & -3 \\ 0 & 0 & 0 & 2 \end{pmatrix}, \quad c = \begin{pmatrix} 24 \\ -12 \\ 6 \\ -3 \end{pmatrix},$$

and $B = -I$. The exact solution for this problem is $x^* = (-3,3,3,-1)$.

Problem 3 ([33])**.** Consider the AVE of the form $Ax + B|x| = c$, where

$$A = \begin{pmatrix} -1 & 8 & -2 & 8 \\ 0 & -1 & 0 & -2 \\ 2 & -8 & 1 & -8 \\ 0 & 2 & 0 & 1 \end{pmatrix}, \quad c = \begin{pmatrix} -24 \\ 8 \\ 22 \\ -10 \end{pmatrix},$$

and $B = -I$. The exact solution for this problem is $x^* = (-1, -1, -8, -4)$.

Problem 4 ([16])**.** Consider the AVE of the form $Ax + B|x| = c$, where the matrix A is chosen

```
A=round(100*(eye(n,n)-0.002*(2*rand(n,n)-1)))
```

and $B = -I$. The exact solution $x^* \in \mathbb{R}^n$ for this problem is chosen randomly and $c = Ax^* - |x^*|$ and 5 different problems are generated with dimensions from 10 to 6000.

Problem 5 ([36])**.** Consider the AVE of the form $Ax + B|x| = c$, where the matrices A and B are generated by the following MATLAB procedure:

```
function [A,B,mat] = createmat(n)
D = diag(randperm(n)');
U = orth(rand(n));
A = U'*D*U;
A = 5*round(A,2);
B = diag(rand(n,1));
B = 5*round(B,2);
mat = A'*A - norm(abs(B)') * norm(abs(B)) * eye(n);
end

function [A,B,positivemat] = createpositivemat(n);
[AA,BB,mata] = createmat(n);
while 1 == 1
if eig(mata)>0
A=AA; B=BB; positivemat=mata;
break;
else
[AA,BB,mata] = createmat(n);
end
end
```

The exact solution $x^* \in \mathbb{R}^n$ for this problem is chosen randomly by $x^* = 2 * rand(n,1) - 2 * rand(n,1)$ and $c = Ax^* - B|x^*|$. Using the code above, 5 different problems are generated with dimensions from 10 to 6000.

The obtained results from application SNA to Problems $1-5$ are reported in Tables 1 and 2. The following list of symbols is used for abbreviations:

Table 1: The numerical results.

Prob. No	n	Function	F_{iter}	F_{eval}	ErF	F_{val}	Time
		φ_1	8	9	$3.3396e-08$	$2.7541e-07$	0.0048
1	4	φ_2	7	8	$2.4825e-16$	$3.5527e-15$	0.0132
		φ_3	4	5	$2.5629e-07$	$2.5629e-07$	0.0077
		φ_1	21	23	$6.8065e-06$	$5.6947e-07$	0.0146
2	4	φ_2	8	9	$5.0856e-06$	$1.986e-15$	0.0174
		φ_3	7	12	$3.3483e-11$	$3.3483e-11$	0.0122
		φ_1	31	37	$1.0719e-07$	$7.5075e-07$	0.0209
3	4	φ_2	6	8	$2.7363e-08$	$3.5527e-15$	0.0292
		φ_3	3	4	$1e-08$	$1e-08$	0.0053
		φ_1	3	4	$8.2182e-10$	$8.3307e-08$	0.0042
	10	φ_2	3	4	$7.9897e-10$	$7.976e-08$	0.0066
		φ_3	4	5	$3.8466e-16$	$6.5538e-12$	0.0431
		φ_1	4	5	$5.0466e-11$	$5.0604e-09$	0.0109
	50	φ_2	4	5	$9.3494e-11$	$9.3281e-09$	0.0111
		φ_3	4	5	$2.0977e-12$	$2.0606e-10$	0.0436
		φ_1	4	5	$1.5013e-11$	$1.4748e-09$	0.0617
4	250	φ_2	4	5	$1.5635e-10$	$1.5248e-08$	0.0308
		φ_3	4	5	$3.659e-09$	$3.6054e-07$	0.0998
		φ_1	4	5	$1.614e-09$	$1.4025e-07$	1.0950
	1250	φ_2	5	6	$1.735e-13$	$1.6061e-11$	1.1125
		φ_3	5	6	$1.3584e-13$	$1.2763e-11$	1.2142
		φ_1	5	6	$2.6528e-12$	$9.8384e-11$	104.23
	6000	φ_2	5	6	$3.1828e-12$	$1.0515e-10$	94.474
		φ_3	5	6	$2.5422e-12$	$9.7223e-11$	93.315
		φ_1	4	5	$5.1271e-07$	$1.4043e-10$	0.0827
	10	φ_2	4	5	$3.2870e-08$	$6.5536e-12$	0.0363
		φ_3	4	5	$1.5904e-08$	$6.5536e-12$	0.0439
		φ_1	4	5	$1.1897e-08$	$5.0684e-10$	0.0709
	50	φ_2	4	5	$5.8989e-08$	$6.5570e-12$	0.1141
		φ_3	4	5	$0.7215e-08$	$6.5577e-12$	0.0556
		φ_1	4	5	$8.8776e-08$	$1.3806e-08$	0.0898
5	250	φ_2	4	5	$7.8995e-08$	$8.2866e-12$	0.2264
		φ_3	4	5	$0.4878e-08$	$8.0337e-12$	0.0801
		φ_1	4	5	$1.0112e-05$	$3.5343e-08$	1.4628
	1250	φ_2	5	6	$6.5107e-06$	$1.8393e-10$	1.4344
		φ_3	4	5	$0.5266e-07$	$1.8404e-10$	0.9649
		φ_1	5	6	$1.4116e-05$	$4.0788e-09$	102.22
	6000	φ_2	4	5	$6.0965e-06$	$3.9206e-09$	92.946
		φ_3	4	5	$0.5180e-07$	$3.8684e-09$	67.426

φ_j : The smoothing function,
F_{iter} : The number of iterations,
F_{eval} : The number of function evaluations,
ErF : The value of $\|\bar{x}-x^*\|$,
F_{val} : The value of $\|F(\bar{x})\|$.

Table 2: Comparison of the methods.

No	n	Func	SNA			Algorithm 1 in [17]			Algorithm 3.1 in [36]		
			F_{iter}	F_{eval}	Time	F_{iter}	F_{eval}	Time	F_{iter}	F_{eval}	Time
1	4	φ_1	8	9	0.0048	9	10	0.0566	10	22	0.0332
		φ_2	7	8	0.0132	8	9	0.0313	5	11	0.0340
		φ_3	4	5	0.0077	4	5	0.0510	5	11	0.0446
2	4	φ_1	21	23	0.0146	22	24	0.0455	22	43	0.0466
		φ_2	8	9	0.0174	22	23	0.0314	22	45	0.0412
		φ_3	7	12	0.0122	9	18	0.0540	8	19	0.0514
3	4	φ_1	31	37	0.0209	35	48	0.0557	8	20	0.0360
		φ_2	6	8	0.0292	31	37	0.0631	32	68	0.0792
		φ_3	3	4	0.0053	7	11	0.0429	8	20	0.0544
4	10	φ_1	3	4	0.0042	4	5	0.0497	6	8	0.0435
		φ_2	3	4	0.0066	3	4	0.0273	4	5	0.0395
		φ_3	4	5	0.0431	3	4	0.0435	4	5	0.0818
	50	φ_1	4	5	0.0109	4	5	0.0587	28	169	0.1470
		φ_2	4	5	0.0111	4	5	0.2905	4	5	0.0407
		φ_3	4	5	0.0436	3	4	0.0376	4	5	0.0427
	250	φ_1	4	5	0.0617	5	6	0.3598	27	1018	1.7268
		φ_2	4	5	0.0308	4	5	0.0726	4	9	0.0798
		φ_3	4	5	0.0998	4	5	0.0731	4	5	0.1233
	1250	φ_1	4	5	1.0950	5	6	1.3424	24	1038	25
		φ_2	5	6	1.1125	4	5	0.9973	4	5	1.0826
		φ_3	5	6	1.2142	4	5	1.1077	4	5	0.9162
	6000	φ_1	5	6	104.23	5	6	110.58	18	64	318.74
		φ_2	5	6	94.474	4	5	75.796	5	6	104.96
		φ_3	5	6	93.315	4	5	86.333	5	6	100.77
5	10	φ_1	4	5	0.0827	3	4	0.0652	6	7	0.0430
		φ_2	4	5	0.0363	4	5	0.0850	5	6	0.0515
		φ_3	4	5	0.0439	4	5	0.0850	6	7	0.0459
	50	φ_1	4	5	0.0709	5	6	0.0688	2	1010	0.4301
		φ_2	4	5	0.1141	5	6	0.0369	5	6	0.0429
		φ_3	4	5	0.0556	3	4	0.0632	5	6	0.0511
	250	φ_1	4	5	0.0898	5	6	0.2198	34	1053	1.9078
		φ_2	4	5	0.2264	6	9	0.1260	7	12	0.0996
		φ_3	4	5	0.0801	4	5	0.1263	5	6	0.0766
	1250	φ_1	4	5	1.4628	6	7	2.2826	23	1014	25.125
		φ_2	5	6	1.4344	5	6	1.1247	22	1013	24.839
		φ_3	4	5	0.9649	6	9	1.6430	6	7	1.4024
	6000	φ_1	5	6	102.22	6	7	136.74	22	1052	757.01
		φ_2	4	5	92.946	5	6	115.28	22	1048	700.23
		φ_3	4	5	67.426	5	6	94.744	24	1008	813.7

We apply *SNA* with three different smoothing functions to solve *Problems* $1-5$ and the obtained numerical results are reported in Table 1. It can be seen from Table 1 that all problems have been successfully solved and the solutions of all of the problems are obtained within a reasonable computation time. If the results are compared according to different smoothing functions, the function φ_3 is the most effective one in most of the problems especially for large dimensions. Although the function φ_3 has come into prominence in terms of numerical results, the φ_1 and φ_2 functions are advantageous in terms of ease of applicability.

Table 2 is created to compare numerical results with other algorithms. For comparison, Algorithm 1 in [17] and Algorithm 3.1 in [36] are selected as these algorithms are the most efficient algorithms among the smoothing Newton type algorithms. Numerical results are compared with SNA in terms of "F_{iter}", "F_{eval}" and "$Time$" in Table 2. For *Problem* 1, SNA with φ_1 presents the best results according to $Time$. For *Problem* 2 and 3, SNA with φ_3 presents the best results in terms of F_{iter}, F_{eval} and $Time$. The SNA and Algorithm 1 present similar performance on *Problem* 4, but Algorithm 3.1 fails for some cases $n \geq 250$. In most cases, SNA with φ_3 presents the best results according to $Time$. Finally, similar performance results are obtained for both SNA and Algorithm 1 on *Problem* 5. The Algorithm 3.1 again fails for some cases $n \geq 250$.

4 Conclusions

In this paper, solving the AVE of type (1) by using a new class of smoothing technique is studied. Three new members of this class have been introduced and they have been successfully applied to AVE. A smoothing Newton-type algorithm with a hybrid inexact line search technique have been developed. This new line search technique is a combination of two different types of inexact line search techniques and utilizes each of them in each loop. It has been proved theoretically that the algorithm is globally convergent with quadratic convergence rate. This theoretical result is also reflected in the numerical results, confirming the accuracy of the technique. The numerical experiments shows that SNA is effective and promising.

The smoothing techniques and algorithm proposed in this study may be extended to the other subclasses of system of nonlinear equations/inequalities such as linear and nonlinear complementarity problems, variational inequality problems. The line search technique proposed in this paper may also be adapted to gradient based unconstrained optimization algorithms. Finally, smoothing functions introduced here, may be considered for solving many nonsmooth optimization problems such as l_1 penalty, image restoration and etc.

Acknowledgements

The author thanks anonymous reviewers for their valuable comments, which helped to improve the paper.

References

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

[2] J.H. Alcantara, J.S. Chen, M.K. Tam, *Method of alternating projections for the general absolute value equation*, J. Fixed Point Theory Appl. **25** (2023) 39.

[3] R. Ali, K. Pan, *The solution of a type of absolute value equations using two new matrix splitting iterative techniques*, Port. Math. **79** (2023) 241–252.

[4] R. Ali, K. Pan, *New generalized GaussSeidel iteration methods for solving absolute value equations*, Math. Methods Appl. Sci. (2023) Doi: https://doi.org/10.1002/mma.9062.

[5] L. Armijo, *Minimization of functions having Lipschitz continuous first partial derivatives*, Pacific J. Math. **16** (1966) 1–3.

[6] D. Bertsekas, *Nondifferentiable optimization via approximation*, Math. Programming Stud.**3** (1975) 1–25.

[7] L. Caccetta, B. Qu, G. Zhou, *A globally and quadratically convergent method for absolute value equations*, Comput. Optim. Appl. **48** (2011) 45–58.

[8] X. Chen, *Smoothing methods for nonsmooth, nonconvex minimization*, Math. Program. 134 (2012) 71–99.

[9] C. Chen, O.L. Mangasarian, *A class of smoothing functions for nonlinear and mixed complementarity problem*, Comput. Optim. Appl. **5** (1996) 97–138.

[10] C. Chen, Y. Yang, D. Yu, D. Han, *An inverse-free dynamical system for solving the absolute value equations*, Appl. Numer. Math. **168** (2021) 170–181.

[11] X. Fan, M. Zeng, L. Jiang, *Smoothing homotopy method for solving second-order cone complementarity problem*, Asia-Pac. J. Oper. Res. **37** (2020) 2050023.

[12] C. Grossmann, *Smoothing techniques for exact penalty function methods*, Contemporary Mathematics, In book Panorama of Mathematics: Pure and Applied, **658** (2016) 249–265.

[13] F. Hashemi, S. Ketabchi, *Numerical comparison of smoothing functions for optimal correction of an infeasible system of absolute value equations*, Numer. Algebra Control Optim. **10** (2019) 13–21.

[14] M. Hladik, H. Moosaei, *Some notes on the solvability conditions for absolute value equations*, Optim. Lett. **17** (2023) 211–218.

[15] S. L. Hu, Z. H. Hu, *A note on absolute value equations*, Optim. Lett. **4** (2010) 417–424.

[16] J. Iqbal, A. Iqbal, M. Arif, *LevenbergMarquardt method for solving systems of absolute value equations*, J. Comput. Appl. Math. **282** (2015), 134–138.

[17] X. Jiang, Y. Zhang, *A smoothing-type algorithm for absolute value equations*, J. Ind. Manag. Optim. **9** (2013) 789–798.

[18] Y. F. Ke, C. F. Ma, *SOR-like iteration method for solving absolute value equations*, Appl. Math. Comput. **311** (2017) 195–202.

[19] S. Ketabchi, H. Moosaei, *Minimum norm solution to the absolute value equation in the convex case*, J. Optim. Theory Appl. **154** (2012) 1080–1087.

[20] Y. Li, S. Du, *Modified HS conjugate gradient method for solving generalized absolute value equations*, J. Inequal. Appl. **2019** (2019) 68.

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

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

[23] O. L. Mangasarian, *A hybrid algorithm for solving the absolute value equation*, Optim. Lett. **9** (2015) 1469–1474.

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

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

[26] X. H. Miao, J. Yang, S. Hu, *A generalized Newton method for absolute value equations associated with circular cones* Appl. Math. Comput. **269** (2015) 155–168.

[27] X. H. Miao, K. Yao, C. Y. Yang, J. S. Chen, *Levenberg-Marquardt method for absolute value equation associated with second-order cone*, Numer. Algebra Control Optim. **12** (2022) 47–61.

[28] C. T. Nguyen, B. Saheya, Y. L. Chang, J. S. Chen, *Unified smoothing functions for absolute value equation associated with second-order cone*, Appl. Numer. Math. **135** (2019) 206–227.

[29] O. Prokopyev, *On equivalent reformulations for absolute value equations*, Comput. Optim. Appl. **44** (2009) 363–372.

[30] L. Qi, D. Sun, *Smoothing functions and smoothing Newton method for complementarity and variational inequality problems*, J. Optim. Theory Appl. **113** (2002) 121–147.

[31] J. Qiu, J. Yu, S. Lian, *Smoothing approximation to the new exact penalty function with two parameters*, Asia-Pac. J. Oper. Res. **38** (2021) 2140010.

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

[33] B. Saheya, C. T. Nguyen, J. S. Chen, *Neural network based on systematically generated smoothing functions for absolute value equations*, J. Appl. Math. Comput. **61** (2019) 533–558.

[34] B. Saheya, C. H. Yu, J. S. Chen, *Numerical comparisons based on four smoothing functions for absolute value equations*, J. Appl. Math. Comput. **56** (2018) 131–149.

[35] A. Sahiner, N. Yilmaz, S. A. Ibrahem, *Smoothing approximations to non-smooth functions*, Journal of Multidisciplinary Modeling and Optimizaiton **1** (2018) 69–74.

[36] J. Tang, J. Zhou, *A quadratically convergent descent method for absolute value equation* $Ax + B|x| = b$, Oper. Res. Lett., **47** (2019) 229–234.

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

[38] F. Wang, Z. Yu, C. Gao, *A smoothing neural network algorithm for absolute value equations*, Engineering **7** (2015) 567–576.

[39] S.L. Wu, *The unique solution of a class of the new generalized absolute value equation*, Appl. Math. Lett. **116** (2021) 107029.

[40] A.E. Xavier, *The hyperbolic smoothing clustering method*, Patt. Recog. **43** (2010) 731–737.

[41] N. Yilmaz, A. Kayacan, *Smoothing LevenbergMarquardt algorithm for solving non-Lipschitz absolute value equations*, J. Appl. Anal. **29** (2023) 277–286.

[42] N. Yilmaz, A. Sahiner, *New smoothing approximations to piecewise smooth functions and applications*, Numer. Funct. Anal. Optim. **40** (2019) 523–534.

[43] N. Yilmaz, A. Sahiner, *Smoothing techniques in solving non-Lipschitz absolute value equations*, Int. J. Comput. Math. **100** (2023) 867–879.

[44] I. Zang, *A smoothing out technique for min-max optimization*. Math. Program. **19** (1980) 61–77.

[45] L. Zhang, X. Zhang, *Global linear and quadratic one-step smoothing Newton method for P_0-LCP*, J. Global Optim. **25** (2003) 363–376.