Journal of Applied Mathematics and Computing (2019) 61:533–558
https://doi.org/10.1007/s12190-019-01262-1

ORIGINAL RESEARCH

Neural network based on systematically generated smoothing functions for absolute value equation

B. Saheya[1] · Chieu Thanh Nguyen[2] · Jein-Shan Chen[2]

Received: 4 January 2019 / Published online: 20 April 2019

Abstract
In this paper, we summarize several systematic ways of constructing smoothing functions and illustrate eight smoothing functions accordingly. Then, based on these systematically generated smoothing functions, a unified neural network model is proposed for solving absolute value equation. The issues regarding the equilibrium point, the trajectory, and the stability properties of the neural network are addressed. Moreover, numerical experiments with comparison are presented, which suggests what kind of smoothing functions work well along with the neural network approach.

Keywords Absolute value equations · Neural network · Smoothing function

Mathematics Subject Classifications 65K10 · 93B40 · 26D07

1 Introduction

The main target that we tackle with in this paper is the so-called absolute value equation (AVE for short), whose mathematical format is as below. In fact, the original standard AVE is described by

B. Saheya: The author's work is supported by National Key R&D Program of China (Award No.: 2017YFC1405605) and Foundation of Inner Mongolia Normal University (Award No.: 2017YJRC003)
J.-S. Chen: The author's work is supported by Ministry of Science and Technology, Taiwan.

✉ Jein-Shan Chen
jschen@math.ntnu.edu.tw

B. Saheya
saheya@imnu.edu.cn

Chieu Thanh Nguyen
thanhchieu90@gmail.com

1 College of Mathematical Science, Inner Mongolia Normal University, Hohhot 010022, Inner Mongolia, People's Republic of China

2 Department of Mathematics, National Taiwan Normal University, Taipei 11677, Taiwan

$$Ax - |x| = b, \tag{1}$$

where $A \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$. Here $|x|$ means the componentwise absolute value of vector $x \in \mathbb{R}^n$. Another generalized absolute value equation is in the form of

$$Ax + B|x| = b, \tag{2}$$

where $B \in \mathbb{R}^{n \times n}$, $B \neq O$. When $B = -I$, I is the identity matrix, the AVE (2) reduces to the special form (1).

The AVE (1) was first introduced by Rohn in [29] and recently has been investigated by many researchers, for example, Hu and Huang [7], Saheya and Chen [32], Jiang and Zhang [9], Ketabchi and Moosaei [10], Mangasarian [14–18], Mangasarian and Meyer [21], Prokopyev [25], and Rohn [31]. In particular, Mangasarian and Meyer [21] show that the AVE (1) is equivalent to the bilinear program, the generalized LCP (linear complementarity problem), and the standard LCP provided 1 is not an eigenvalue of A. With these equivalent reformulations, they also prove that the AVE (1) is NP-hard in its general form and provide existence results. Prokopyev [25] further obtain an improvement indicating that the AVE (1) can be equivalently recast as (a larger) LCP without any assumption on A and B, and also provides a relationship with mixed integer programming. It is known that, if solvable, the AVE (1) can have either unique solution or multiple (e.g., exponentially many) solutions. Indeed, various sufficiency conditions on solvability and non-solvability of the AVE (1) with unique and multiple solutions are discussed in [21,25,30]. Some variants of the AVEs including the absolute value equation associated with second-order cone (SOCAVE) and the absolute value programs, are investigated in [8] and [38], respectively. Furthermore, some other type of absolute value equation, an extension of the AVE (2), is considered [8,19,20].

Roughly, there have three approaches for dealing with the AVEs (1)–(2). The first one is reformulating the AVEs (1)–(2) as complementarity problem and then solve it accordingly. The second one is to recast the AVEs (1)–(2) as a system of nonsmooth equations and then tackle with the nonsmooth equations by applying nonsmooth Newton algorithm [26] or smoothing Newton algorithm [27]. The third one is applying the neural network approach. In this paper, we follow the third idea for solving the AVEs (1)–(2). Inspired by our another recent work [24], we will combine various smoothing functions with the neural network approach. Different from [24,32], the smoothing functions studied in this paper are not only constructed from one way, they are generated by several systematic ways. Accordingly, this one can be viewed as a follow-up of [24,32].

Now, we quickly go over neural network approach which is different from traditional optimization methods. To consider this approach, the main reason lies on the real-time solutions of optimization problems, which are sometimes required in practice. It is well known that the neural networks approach is a very promising approach to solving the real-time optimization problem. In general, the neural networks can be implemented using integrated circuits and were first introduced in the 1980s by Hopfield and Tank [6,34] for optimization problems. Since then, significant research results have been achieved for various optimization problems, including linear programming [39], quadratic programming [1], linear complementarity problems [12], nonlinear complementarity problem [13] and nonlinear programming [5]. In general, the essence of neural network approach is to construct a nonnegative energy function and establish a dynamic system that represents an artificial neural network. A first order differential equation represents the dynamic system. Furthermore, it is expected that the dynamic system will converge to its static state (or an equilibrium point), which corresponds to the solution for the underlying optimization problem, starting from an initial point.

Although similar idea was employed by Wang, Yu and Guo in [36], only one smoothing function was studied therein. In this paper, we present systematical ways about how to construct smoothing functions for AVE (1) and illustrate eight smoothing functions accordingly. After that, we design a gradient descent neural network model by using these eight different smoothing functions. We not only discuss the stability of the neural networks, but also give numerical comparison for these smoothing functions. In fact, the new upshot of this paper lies on the numerical comparison, which suggest what kind of smoothing functions work well along with the neural network approach.

2 Preliminaries

By looking into the mathematical format of the aforementioned AVEs, it is observed that the absolute value function $|x|$ is the key component. Indeed, the absolute value function also plays an important role in a lot of applications, like machine learning and image processing, etc. In particular, the absolute value function $|x|$ is not differentiable at $x = 0$, which causes limits in analysis and application. To conquer this hurdle, researchers consider smoothing methods and construct smoothing functions for it. We summarize all possible ways to construct smoothing functions for $|x|$ as below. For more details, please refer to [2,4,11,23,28,35].

1. Smoothing by the convex conjugate

Let X be a real topological vector space, and let X^* be the dual space to X. For any function $f : \mathrm{dom} f \to \mathbb{R}$, its convex conjugate $f^* : (\mathrm{dom} f)^* \to \mathbb{R}$ is defined in terms of the supremum by

$$f^*(y) := \sup_{x\in\mathrm{dom} f} \left\{ x^T y - f(x) \right\}.$$

In light of this, one can build up smooth approximation of f, denoted by f_μ, by adding strongly convex component to its dual $g := f^*$, namely,

$$f_\mu(x) = \sup_{z\in\mathrm{dom} g} \left\{ z^T x - g(x) - \mu d(z) \right\} = (g + \mu d)^*(x),$$

for some 1-strongly convex and continuous function $d(\cdot)$ (called proximity function). Here, $d(\cdot)$ is 1-strongly convex which satisfies

$$d((1-t)x + ty) \le (1-t)d(x) + td(y) - \frac{1}{2}t(1-t)\|x - y\|^2,$$

for all x, y and $t \in (0, 1)$. Note that $|x| = \sup_{|z|\le 1} zx$. If we take $d(z) := z^2/2$, then the constructed smoothing function via conjugation leads to

$$\phi_1(\mu, x) = \sup_{|z| \leq 1} \left\{ zx - \frac{\mu}{2} z^2 \right\} = \begin{cases} \frac{x^2}{2\mu}, & \text{if } |x| \leq \mu, \\ |x| - \frac{\mu}{2}, & \text{otherwise.} \end{cases} \tag{3}$$

which is the traditional Huber function.

It is also possible to consider another expression:

$$|x| = \sup_{\substack{z_1 + z_2 = 1 \\ z_1, z_2 \succeq 0}} (z_1 - z_2)x.$$

Under this case, if we take $d(z) := z_1 \log z_1 + z_2 \log z_2 + \log 2$, the constructed smoothing function by conjugation becomes

$$\phi_2(\mu, x) = \mu \log \left(\cosh \left(\frac{x}{\mu} \right) \right), \tag{4}$$

where $\cosh(x) := \dfrac{e^x + e^{-x}}{2}$. Alternatively, choosing $d(y) := 1 - \sqrt{1 - y^2}$ gives another smoothing function:

$$\phi_3(\mu, x) = \sup_{-1 \leq y \leq 1} \left(xy + \mu \sqrt{1 - y^2} - \mu \right) = \sqrt{x^2 + \mu^2} - \mu. \tag{5}$$

2. The Moreau proximal smoothing

Suppose that $\mathbb{E}$ is an Euclidean space and $f : \mathbb{E} \to (-\infty, \infty]$ is a closed and proper convex function. One natural tool for generating an approximate smoothing function is through the use of the so-called proximal map introduced by Moreau [22]. The Moreau proximal approximation yields a family of approximations $\{f_\mu^{\text{px}}\}_{\mu > 0}$ as below:

$$f_\mu^{\text{px}}(x) = \inf_{u \in \mathbb{E}} \left\{ f(u) + \frac{1}{2\mu} \|u - x\|^2 \right\}. \tag{6}$$

It is known that the Moreau proximal approximation $f_\mu^{\text{px}}(x)$ is convex continuous, finite-valued, and differentiable with gradient ∇f_μ^{px} which is Lipschitz continuous with constant $\frac{1}{\mu}$, see [22]. When applying the Moreau proximal smoothing way to construct the smoothing function for the absolute value function $|x|$, it also yields the Huber smoothing function $\phi_1(\mu, x)$ by using the Moreau envelope [2].

3. Nesterov's smoothing

There is a class of nonsmooth convex functions considered in [23], which is given by

$$q(x) = \max\{\langle u, Ax \rangle - \phi(u) \mid u \in Q\}, \quad x \in \mathbb{E},$$

where $\mathbb{E}$, V are finite dimensional vector spaces, $Q \subseteq V^*$ is compact and convex, ϕ is a continuous convex function on Q, and $A : \mathbb{E} \to V$ is a linear map. The smooth approximation of q suggested in [23] is described by the convex function

$$q_\mu(x) = \max\{\langle u, Ax\rangle - \phi(u) - \mu d(u) \mid u \in Q\}, \quad x \in \mathbb{E}, \tag{7}$$

where $d(\cdot)$ is a prox-function for Q. It was proved in [23, Theorem 1] that the convex function $q_\mu(x)$ is $C^{1,1}(\mathbb{E})$. More specifically, its gradient mapping is Lipschitz continuous with constant $L_\mu = \dfrac{\|A\|^2}{\sigma\mu}$ and the gradient is described by $\nabla q_\mu(x) = Au_\mu(x)$, where $u_\mu(x)$ is the unique minimizer of (7).

For the absolute value function $q(x) = |x|$ with $x \in \mathbb{R}^1$. Let $A = 1$, $b = 0$, $\mathbb{E} = \mathbb{R}^1$, $Q = \{u \in \mathbb{R}^1 \mid |u| \leq 1\}$ and taking $d(u) := \frac{1}{2}u^2$. Then, we have

$$\begin{aligned} q_\mu(x) &= \max_u \{\langle Ax - b, u\rangle - \mu d(u) \mid u \in Q\} \\ &= \max_u \left\{ xu - \frac{\mu}{2}u^2 \right\} \\ &= \begin{cases} \frac{x^2}{2\mu}, & \text{if } |x| \leq \mu, \\ |x| - \frac{\mu}{2}, & \text{otherwise.} \end{cases} \end{aligned}$$

As we see, it also yields the Huber smoothing function $\phi_1(\mu, x)$ defined by (3) through this approximation way.

4. The infimal-convolution smoothing technique

Suppose that $\mathbb{E}$ is a finite vector space and $f, g : \mathbb{E} \to (-\infty, \infty]$. The infimal convolution of f and g, $f \,\Box\, g : \mathbb{E} \to [-\infty, +\infty]$ is defined by

$$(f \,\Box\, g)(x) = \inf_{y\in\mathbb{E}} \{f(y) + g(x - y)\}.$$

In light of the concept of infimal convolution, one can also construct smoothing approximation functions. More specifically, we consider $f : \mathbb{E} \to (-\infty, \infty]$ which is a closed proper convex function and let $\omega : \mathbb{E} \to \mathbb{R}$ be a $C^{1,1}$ convex function with Lipschitz gradient constant $1/\sigma$ ($\sigma > 0$). Suppose that for any $\mu > 0$ and any $x \in \mathbb{E}$, the following infimal convolution is finite:

$$f_\mu^{\mathrm{ic}}(x) = \inf_{u\in\mathbb{E}} \left\{ f(u) + \mu\omega\left(\frac{x-u}{\mu}\right) \right\} = (f \,\Box\, \omega_\mu)(x), \tag{8}$$

where $\omega_\mu(\cdot) = \mu\omega\left(\frac{\cdot}{\mu}\right)$. Then, f_μ^{ic} is called the infimal-convolution μ-smooth approximation of f. In particular, when $\mu \in \mathbb{R}_{++}$ and $p \in (1, +\infty)$, the infimal convolution of a convex function and a power of the norm function is obtained as below:

$$f \,\Box\, \left(\frac{1}{\mu p}\|\cdot\|^p\right) = \inf_{u\in\mathbb{E}} \left\{ f(u) + \left(\frac{1}{\mu p}\|x - u\|^p\right) \right\}. \tag{9}$$

For the absolute value function, it can be verified that $f_\mu(x) = (|\cdot|) \,\Box\, \left(\frac{1}{\mu * p}|\cdot|^p\right)$ is the Huber function of order p, i.e.,

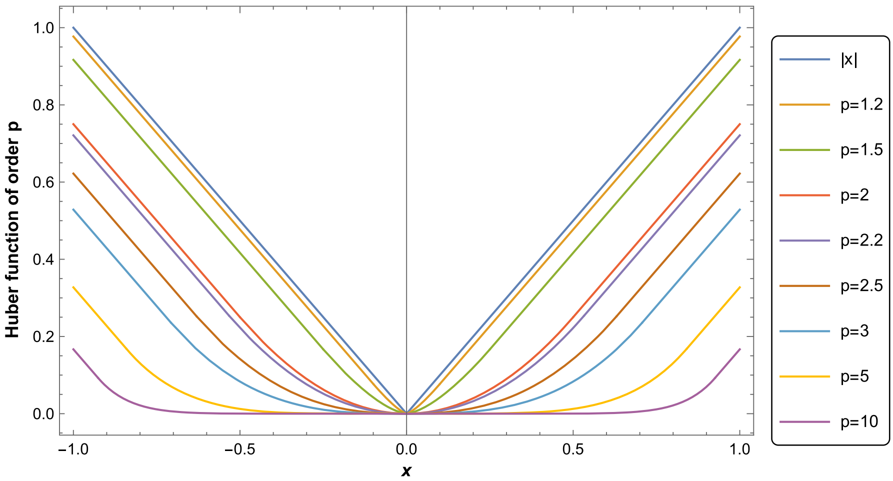

Fig. 1 $|x|$ and Huber function of order p ($\mu = 0.3$)

$$f_\mu(x) = \begin{cases} |x| - \frac{p-1}{p}\mu^{\frac{1}{p-1}}, & \text{if} \quad |x| > \mu^{\frac{1}{p-1}}, \\ \frac{|x|^p}{\mu p}, & \text{if} \quad |x| \le \mu^{\frac{1}{p-1}}. \end{cases} \tag{10}$$

Note that when $p = 2$ in the above expression (10), the Huber function of order p reduces to the Huber function $\phi_1(\mu, x)$ as shown in (3). Figure 1 depicts the Huber function of order p with various value of p. To the contrast, plugging $p = 2$ into infimal convolution formula (9) yields the Moreau approximation (6). For more details about infimal convolution and its induces approximation functions, please refer to [2,3].

5. The convolution smoothing technique

The smoothing approximation via convolution for the absolute value function is a popular way, which can be found in [4,11,28,35]. Its construction idea is described as follows. First, one constructs a smoothing approximation for the plus function $(x)_+ = \max\{0, x\}$. To this end, we consider the piecewise continuous function $d(x)$ with finite number of pieces which is a density (kernel) function, that is, it satisfies

$$d(x) \ge 0 \quad \text{and} \quad \int_{-\infty}^{+\infty} d(x)dx = 1.$$

Next, define $\hat{s}(\mu, x) := \frac{1}{\mu} d\left(\frac{x}{\mu}\right)$, where μ is a positive parameter. Suppose that $\int_{-\infty}^{+\infty} |x|\, d(x)dx < +\infty$, then a smoothing approximation (denoted by $\hat{p}(\mu, x)$) for $(x)_+$ is obtained as below:

$$\hat{p}(\mu, x) = \int_{-\infty}^{+\infty} (x - s)_+ \hat{s}(\mu, s)ds = \int_{-\infty}^{x} (x - s)\hat{s}(\mu, s)ds.$$

The following are four well-known smoothing functions for the plus function [4,28]:

$$\hat{\phi}_1(\mu, x) = x + \mu \log\left(1 + e^{-\frac{x}{\mu}}\right). \tag{11}$$

$$\hat{\phi}_2(\mu, x) = \begin{cases} x & \text{if} \quad x \geq \frac{\mu}{2}, \\ \frac{1}{2\mu}\left(x + \frac{\mu}{2}\right)^2 & \text{if} \quad -\frac{\mu}{2} < x < \frac{\mu}{2}, \\ 0 & \text{if} \quad x \leq -\frac{\mu}{2}. \end{cases} \tag{12}$$

$$\hat{\phi}_3(\mu, x) = \frac{\sqrt{4\mu^2 + x^2} + x}{2}. \tag{13}$$

$$\hat{\phi}_4(\mu, x) = \begin{cases} x - \frac{\mu}{2} & \text{if} \quad x > \mu, \\ \frac{x^2}{2\mu} & \text{if} \quad 0 \leq x \leq \mu, \\ 0 & \text{if} \quad x < 0. \end{cases} \tag{14}$$

where their corresponding kernel functions are

$$d_1(x) = \frac{e^{-x}}{(1 + e^{-x})^2},$$

$$d_2(x) = \begin{cases} 1 & \text{if } -\frac{1}{2} \leq x \leq \frac{1}{2}, \\ 0 & \text{otherwise}, \end{cases}$$

$$d_3(x) = \frac{2}{(x^2 + 4)^{\frac{3}{2}}},$$

$$d_4(x) = \begin{cases} 1 & \text{if } 0 \leq x \leq 1, \\ 0 & \text{otherwise}. \end{cases}$$

Using the fact that $|x| = (x)_+ + (-x)_-$. Then, the smoothing function of $|x|$ via convolution can be written as

$$\hat{p}(\mu, |x|) = \hat{p}(\mu, x) + \hat{p}(\mu, -x) = \int_{-\infty}^{+\infty} |x - s|\, \hat{s}(\mu, s) ds.$$

Analogous to (11)–(14), we reach the following smoothing functions for $|x|$:

$$\phi_4(\mu, x) = \mu\left[\log\left(1 + e^{-\frac{x}{\mu}}\right) + \log\left(1 + e^{\frac{x}{\mu}}\right)\right]. \tag{15}$$

$$\phi_5(\mu, x) = \begin{cases} x & \text{if} \quad x \geq \frac{\mu}{2}, \\ \frac{x^2}{\mu} + \frac{\mu}{4} & \text{if} \quad -\frac{\mu}{2} < x < \frac{\mu}{2}, \\ -x & \text{if} \quad x \leq -\frac{\mu}{2}. \end{cases} \tag{16}$$

$$\phi_6(\mu, x) = \sqrt{4\mu^2 + x^2}. \tag{17}$$

as well as the Huber function (3). If we take a Epanechnikov kernel function

$$K(x) = \begin{cases} \frac{3}{4}(1 - x^2) & \text{if } |x| \leq 1, \\ 0 & \text{otherwise}, \end{cases}$$

we achieve the smoothing function for $|x|$:

$$\phi_7(\mu,x)=\begin{cases} x & \text{if} \quad x>\mu, \\ -\frac{x^4}{8\mu^3}+\frac{3x^2}{4\mu}+\frac{3\mu}{8} & \text{if} \quad -\mu\le x\le\mu, \\ -x & \text{if} \quad x<-\mu. \end{cases} \tag{18}$$

Moreover, if we take a Gaussian kernel function $K(x)=\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$ for all $x\in\mathbb{R}$. Then, it yields

$$\hat{s}(\mu,x):=\frac{1}{\mu}K\left(\frac{x}{\mu}\right)=\frac{1}{\sqrt{2\pi\mu^2}}e^{-\frac{x^2}{2\mu^2}}.$$

Hence, we obtain the below smoothing function [35] for $|x|$:

$$\phi_8(\mu,x)=x\,\mathrm{erf}\left(\frac{x}{\sqrt{2}\mu}\right)+\sqrt{\frac{2}{\pi}}\mu e^{-\frac{x^2}{2\mu^2}}. \tag{19}$$

where the error function is defined by

$$\mathrm{erf}(x)=\frac{2}{\sqrt{\pi}}\int_0^x e^{-u^2}du, \quad \forall x\in\mathbb{R}.$$

To sum up, we have eight smoothing functions in total through the above constructions. Figure 2 depicts the graphs of all the aforementioned smoothing functions ϕ_i, $i=1,\ldots,8$ and the absolute value equation. Not only from the geometric view, ϕ_i, $i=1,\ldots,8$ are clearly smoothing functions of $|x|$, it can be also verified theoretically in Proposition 2.1.

$$\phi_1(\mu,x)=\sup_{|z|\le1}\left\{zx-\frac{\mu}{2}z^2\right\}=\begin{cases} \frac{x^2}{2\mu}, & \text{if } |x|\le\mu, \\ |x|-\frac{\mu}{2}, & \text{otherwise.} \end{cases}$$

$$\phi_2(\mu,x)=\mu\log\left(\cosh\left(\frac{x}{\mu}\right)\right).$$

$$\phi_3(\mu,x)=\sup_{-1\le y\le1}\left(xy+\mu\sqrt{1-y^2}-\mu\right)=\sqrt{x^2+\mu^2}-\mu.$$

$$\phi_4(\mu,x)=\mu\left[\log\left(1+e^{-\frac{x}{\mu}}\right)+\log\left(1+e^{\frac{x}{\mu}}\right)\right].$$

$$\phi_5(\mu,x)=\begin{cases} x & \text{if} \quad x\ge\frac{\mu}{2}, \\ \frac{x^2}{\mu}+\frac{\mu}{4} & \text{if} \quad -\frac{\mu}{2}<x<\frac{\mu}{2}, \\ -x & \text{if} \quad x\le-\frac{\mu}{2}. \end{cases}$$

$$\phi_6(\mu,x)=\sqrt{4\mu^2+x^2}.$$

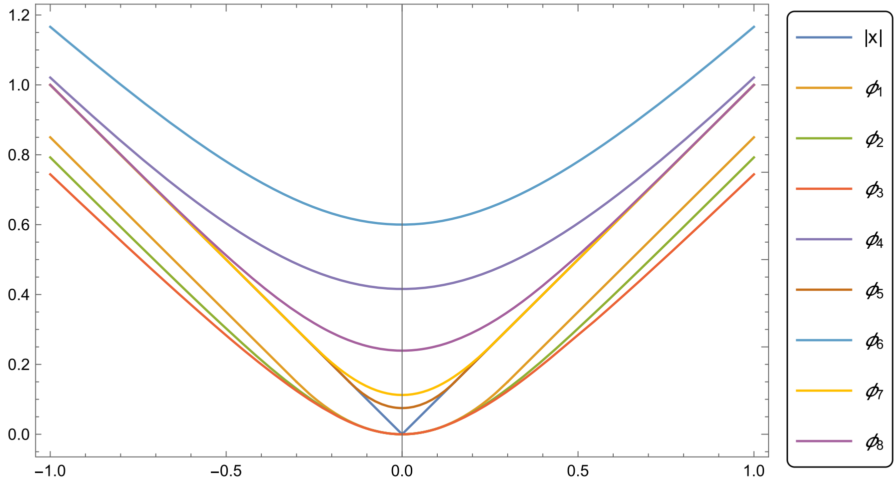

Fig. 2 The graphs of $|x|$ and the smoothing functions ϕ_i, $i = 1, \ldots, 8$ ($\mu = 0.3$)

$$\phi_7(\mu, x) = \begin{cases} x & \text{if} \quad x > \mu, \\ -\frac{x^4}{8\mu^3} + \frac{3x^2}{4\mu} + \frac{3\mu}{8} & \text{if} \quad -\mu \leq x \leq \mu, \\ -x & \text{if} \quad x < -\mu. \end{cases}$$

$$\phi_8(\mu, x) = x \operatorname{erf}\left(\frac{x}{\sqrt{2}\mu}\right) + \sqrt{\frac{2}{\pi}}\mu e^{-\frac{x^2}{2\mu^2}}.$$

From Fig. 2, we see that the local behavior of all eight smoothing functions can be described as

$$\phi_3 \leq \phi_2 \leq \phi_1 \leq |x| \leq \phi_5 \leq \phi_7 \leq \phi_8 \leq \phi_4 \leq \phi_6. \tag{20}$$

In particular, three smoothing function ϕ_1, ϕ_2, ϕ_3 approach to $|x|$ from below with $\phi_1 \geq \phi_2 \geq \phi_3$. To the contrast, the other five smoothing functions $\phi_4, \phi_5, \phi_6, \phi_7, \phi_8$ appraoch to $|x|$ from above with $\phi_5 \leq \phi_7 \leq \phi_8 \leq \phi_4 \leq \phi_6$. Apparently, the smoothing function ϕ_1 and ϕ_5 are closest to $|x|$ among these smoothing functions.

Besides the geometric observation, we also provide algebraic analysis for (20). Noting that each function $\phi_i(\mu, x)$, for $i = 1, 2, \ldots, 8$, is symmetric, so we only need to prove (20) with $x \geq 0$. To proceed, for fixed $\mu > 0$, we let $y = \frac{x}{\mu}$. The verifications consist of seven parts.

Part (1): $\phi_3(\mu, x) \leq \phi_2(\mu, x)$. To verify this inequality, we consider

$$f(y) = \log\left(\frac{e^y + e^{-y}}{2}\right) - \sqrt{y^2 + 1} + 1.$$

Then, we compute the derivation of $f(y)$ as below:

$$f'(y) = \frac{e^y - e^{-y}}{e^y + e^{-y}} - \frac{y}{\sqrt{y^2 + 1}}$$

$$
\begin{aligned}
&= \frac{e^{2y}-1}{e^{2y}+1} - \frac{y}{\sqrt{y^2+1}} \\
&= 1 - \frac{2}{e^{2y}+1} - 1 + \frac{\sqrt{y^2+1}-y}{\sqrt{y^2+1}} \\
&= \frac{1}{\sqrt{y^2+1}\left(\sqrt{y^2+1}+y\right)} - \frac{2}{e^{2y}+1} \\
&= \frac{e^{2y}-1-2y^2-2y\sqrt{y^2+1}}{\left(e^{2y}+1\right)\sqrt{y^2+1}\left(\sqrt{y^2+1}+y\right)}.
\end{aligned}
$$

For convenience, we denote $g(y) = e^{2y} - 1 - 2y^2 - 2y\sqrt{y^2+1}$. It is known that the function e^x can be expressed as

$$
e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}, \tag{21}
$$

which indicates $e^x - 1 \geq x + \frac{x^2}{2} + \frac{x^3}{6}$. Then, it follows that

$$
\begin{aligned}
g(y) &\geq 2y + \frac{(2y)^2}{2} + \frac{(2y)^3}{6} - 2y^2 - 2y\sqrt{y^2+1} \\
&= \frac{4y^3}{3} + 2y\left(1 - \sqrt{y^2+1}\right) \\
&= \frac{4y^3}{3} - \frac{2y^3}{1+\sqrt{y^2+1}} \\
&= 2y^3\left(\frac{2}{3} - \frac{1}{1+\sqrt{y^2+1}}\right) \\
&\geq 2y^3\left(\frac{2}{3} - \frac{1}{2}\right) \\
&\geq 0, \quad \forall y \geq 0.
\end{aligned}
$$

This implies that $f'(y) \geq 0$ for all $y \geq 0$. Thus, f is monotonically nondecreasing which yields $f(y) \geq f(0) = 0$. Then, we verify the assertion that $\phi_3(\mu, x) \leq \phi_2(\mu, x)$.

Part (2): $\phi_2(\mu, x) \leq \phi_1(\mu, x)$. In order to prove this inequality, we discuss two cases.

(i) For $0 \leq x \leq \mu$, this implies that $0 \leq y \leq 1$. Considering

$$
f(y) = \frac{y^2}{2} - \log\left(\frac{e^y + e^{-y}}{2}\right)
$$

yields that

$$f'(y) = y - \frac{e^{2y}-1}{e^{2y}+1} = \frac{ye^{2y}+y-e^{2y}+1}{e^{2y}+1}.$$

By denoting $g(y) := ye^{2y} + y - e^{2y} + 1$ and using (21) leads to

$$\begin{aligned} g(y) &= ye^{2y} + y - e^{2y} + 1 \\ &= y\sum_{n=0}^{\infty}\frac{(2y)^n}{n!} - \sum_{n=0}^{\infty}\frac{(2y)^n}{n!} + y + 1 \\ &= y\sum_{n=0}^{\infty}\frac{(2y)^n}{n!} - \left(\sum_{n=0}^{\infty}\frac{(2y)^{n+1}}{(n+1)!} + 1\right) + y + 1 \\ &= y\left(\sum_{n=0}^{\infty}\frac{(2y)^n}{n!}\left(1 - \frac{2}{n+1}\right)\right) + y \\ &\geq 0, \quad \forall y \in [0,1]. \end{aligned}$$

Therefor, we obtain that $f'(y) \geq 0$ for all $y \in [0, 1]$.

(ii) For $x > \mu$, this implies that $y > 1$. Considering

$$f(y) = y - \frac{1}{2} - \log\left(\frac{e^y + e^{-y}}{2}\right)$$

gives

$$f'(y) = 1 - \frac{e^{2y}-1}{e^{2y}+1} > 0.$$

To sum up, we obtain that $f'(y) \geq 0$ for all $y \in [0, 1]$ in both cases. Following the same arguments as in part(1), we conclude that $\phi_2(\mu, x) \leq \phi_1(\mu, x)$.

Part (3): $\phi_1(\mu, x) \leq |x|$ and $|x| \leq \phi_5(\mu, x)$. It is easy to verify these inequalities. We omit the verification.

Part (4): $\phi_5(\mu, x) \leq \phi_7(\mu, x)$. We will prove this inequality by discussing three cases.

(i) For $x > \mu$, it is easy to see that $\phi_5(\mu, x) = \phi_7(\mu, x) = x$.

(ii) For $\frac{\mu}{2} \leq x \leq \mu$, it means $\frac{1}{2} \leq y \leq 1$. Considering

$$\begin{aligned} f(y) &= -\frac{y^4}{8} + \frac{3y^2}{4} + \frac{3}{8} - y \\ &= \frac{-y^4 + 6y^2 - 8y + 6}{8} \\ &= \frac{-(y^2-1)^2 + 4(y-1)^2 + 3}{8} \end{aligned}$$

$$= \frac{(y-1)^2\left(4-(y+1)^2\right)+3}{8}$$

ad using the facts of $\frac{1}{2} \le y \le 1$ and $\frac{9}{4} \le (y+1)^2 \le 4$, it follows that $f(y) \ge 0$.

(iii) For $0 \le x < \frac{\mu}{2}$, $0 \le y < \frac{1}{2}$. Considering

$$\begin{aligned} f(y) &= -\frac{y^4}{8} + \frac{3y^2}{4} + \frac{3}{8} - y^2 - \frac{1}{4} \\ &= -\frac{y^4}{8} - \frac{y^2}{4} + \frac{1}{8} \\ &= \frac{-y^4 - 2y^2 + 1}{8} \\ &= \frac{2 - (y^2+1)^2}{8} \end{aligned}$$

and applying the facts $0 \le y < \frac{1}{2}$ and $1 \le (y^2+1)^2 \le \frac{25}{16}$, it follows that $f(y) > 0$. From all the above, we achieve that $\phi_5(\mu, x) \le \phi_7(\mu, x)$.

Part (5): $\phi_7(\mu, x) \le \phi_8(\mu, x)$. To proceed this assertion, we discuss two cases.
(i) For $0 \le x \le \mu$, this implies that $0 \le y \le 1$. Consider

$$f(y) = y\mathrm{erf}\left(\frac{y}{\sqrt{2}}\right) + \sqrt{\frac{2}{\pi}}e^{-\frac{y^2}{2}} + \frac{y^4}{8} - \frac{3y^2}{4} - \frac{3}{8}.$$

By applying [35, Lemma 2.5], we have $\mathrm{erf}\left(\frac{y}{\sqrt{2}}\right) \ge \left(1 - e^{-\frac{y^2}{2}}\right)^{\frac{1}{2}}$. Then, it implies that

$$f(y) \ge y\left(1 - e^{-\frac{y^2}{2}}\right)^{\frac{1}{2}} + \sqrt{\frac{2}{\pi}}e^{-\frac{y^2}{2}} + \frac{y^4}{8} - \frac{3y^2}{4} - \frac{3}{8} := g(y).$$

It is easy to verify $g(y)$ is monotonically decreasing on $[0, 1]$ which indicates that $g(y) \ge g(1) > 0$. Hence, we obtain $f(y) > 0$ and $\phi_7(\mu, x) \le \phi_8(\mu, x)$ is proved.

(ii) For $x > \mu$, it means $y > 1$. Consider

$$f(y) = y\mathrm{erf}\left(\frac{y}{\sqrt{2}}\right) + \sqrt{\frac{2}{\pi}}e^{-\frac{y^2}{2}} - y,$$

which yields $f'(y) = \mathrm{erf}\left(\frac{y}{\sqrt{2}}\right) - 1 < 0$. Hence, $f(y)$ is monotonically decreasing on $[1, +\infty)$. Moreover, using [35, Lemma 2.5] gives

$$f(y) \ge y\left(1 - e^{-\frac{y^2}{2}}\right)^{\frac{1}{2}} + \sqrt{\frac{2}{\pi}}e^{-\frac{y^2}{2}} - y.$$

Taking the limit in this inequality, we obtain $\lim_{y\to\infty} f(y) \geq 0$. Therefore, $f(y) \geq \lim_{y\to\infty} f(y) \geq 0$, which show the assertion $\phi_7(\mu, x) \leq \phi_8(\mu, x)$.

Part (6): $\phi_8(\mu, x) \leq \phi_4(\mu, x)$. Consider

$$f(y) = \log\left(1 + e^{-y}\right) + \log\left(1 + e^{y}\right) - y\mathrm{erf}\left(\frac{y}{\sqrt{2}}\right) - \sqrt{\frac{2}{\pi}}e^{-\frac{y^2}{2}}.$$

Then, we have

$$\begin{aligned} f'(y) &= \frac{e^y - 1}{e^y + 1} - \mathrm{erf}\left(\frac{y}{\sqrt{2}}\right) \\ &= 1 - \left(\frac{2}{e^y + 1} + \mathrm{erf}\left(\frac{y}{\sqrt{2}}\right)\right) := 1 - g(y), \end{aligned}$$

which says that

$$\begin{aligned} g'(y) &= -\frac{2e^y}{e^y + 1} + \frac{2}{\sqrt{\pi}}\frac{1}{\sqrt{2}}e^{-\frac{y^2}{2}} \\ &= 2\left(-\frac{e^y}{e^y + 1}\frac{2}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\right) \\ &= 2\left(\frac{-\sqrt{2\pi}e^y + (1 + e^y)e^{-\frac{y^2}{2}}}{\sqrt{2\pi}(1 + e^y)}\right) \\ &\leq 2\left(\frac{-\sqrt{2\pi}e^y + 1 + e^y}{\sqrt{2\pi}(1 + e^y)}\right), \quad \text{as } y \geq 0, \\ &= 2\left(\frac{(1 - \sqrt{2\pi})e^y + 1}{\sqrt{2\pi}(1 + e^y)}\right) \\ &\leq 2(1 - \sqrt{2\pi} + 1) < 0, \quad \text{as } e^y \geq 1,\ 1 - \sqrt{2\pi} < 0. \end{aligned}$$

Hence, $g(y) < g(0) = 1$ which leads to $f'(y) > 0$ for all $y \geq 0$. Then, it follows that $f(y) > f(0) = 2\log 2 - \sqrt{\frac{2}{\pi}} > 0$, and hence $\phi_8(\mu, x) \leq \phi_4(\mu, x)$ is proved.

Part (7): $\phi_4(\mu, x) \leq \phi_6(\mu, x)$. Consider

$$f(y) = \sqrt{4 + y^2} - \left[\log\left(1 + e^{-y}\right) + \log\left(1 + e^{y}\right)\right],$$

which gives

$$\begin{aligned} f'(y) &= \frac{y}{\sqrt{4 + y^2}} - \frac{e^y - 1}{e^y + 1} \\ &= \frac{y}{\sqrt{4 + y^2}} - 1 + \frac{2}{e^y + 1} \end{aligned}$$

$$= \frac{y - \sqrt{4+y^2}}{\sqrt{4+y^2}} + \frac{2}{e^y+1}$$
$$= \frac{-4(1+e^y) + 2\sqrt{4+y^2}\left(y+\sqrt{4+y^2}\right)}{\sqrt{4+y^2}\left(y+\sqrt{4+y^2}\right)(1+e^y)}.$$

For convenience, we denote

$$g(y) := -2(1+e^y) + \sqrt{4+y^2}\left(y+\sqrt{4+y^2}\right) = -2e^y + 2 + y^2 + y\sqrt{4+y^2}.$$

Because $e^y > 1 + y + \frac{y^2}{2}$, it yields

$$g(y) < -2y + y\sqrt{4+y^2} = y\left(2 - \sqrt{4+y^2}\right) \leq 0, \quad \forall y \geq 0.$$

This means that $f'(y) < 0$, i.e., $f(y)$ is monotonically decreasing on $[0, +\infty)$. On the other hand, we know that

$$\begin{aligned}
\lim_{y\to\infty} f(y) &= \lim_{y\to\infty} \sqrt{4+y^2} - \left[\log\left(1+e^{-y}\right) + \log\left(1+e^y\right)\right] \\
&= \lim_{y\to\infty} \sqrt{4+y^2} - y + y - \left[\log\left(1+e^{-y}\right) + \log\left(1+e^y\right)\right] \\
&= \lim_{y\to\infty} \sqrt{4+y^2} - y + \lim_{y\to\infty} y - \left[\log\left(1+e^{-y}\right) + \log\left(1+e^y\right)\right] \\
&= \lim_{y\to\infty} y - \log\left(1+e^y\right) = \lim_{y\to\infty} \log\frac{e^y}{1+e^y} = 0.
\end{aligned}$$

Thus, $f(y) \geq \lim_{y\to\infty} f(y) = 0$ which implies that $\phi_4(\mu, x) \leq \phi_6(\mu, x)$.

From Parts (1)–(7), the proof of (20) is complete.

Proposition 2.1 *Let $\phi_i : \mathbb{R}^2 \to \mathbb{R}$ for $i = 1, \ldots, 6$ be defined as in (3)–(5) and (15)–(19), respectively. Then, we have*

- **(a)** *ϕ_i is continuously differentiable at $(\mu, x) \in \mathbb{R}_{++} \times \mathbb{R}$;*
- **(b)** $\lim_{\mu\downarrow 0} \phi_i(\mu, x) = |x|$.

Proof The proof is straightforward and we omit it. □

Next, we recall some materials about first order differential equations (ODE):

$$\dot{w}(t) = H(w(t)), \; w(t_0) = w_0 \in \mathbb{R}^n, \tag{22}$$

where $H : \mathbb{R}^n \to \mathbb{R}^n$ is a mapping. We also introduce three kinds of stability that will be discussed later. These materials can be found in usual ODE textbooks. A point

$w^* = w(t^*)$ is called an equilibrium point or a steady state of the dynamic system (22) if $H(w^*) = 0$. If there is a neighborhood $\Omega^* \subseteq \mathbb{R}^n$ of w^* such that $H(w^*) = 0$ and $H(w) \neq 0 \ \forall w \in \Omega^* \backslash \{w^*\}$, then w^* is called an isolated equilibrium point.

Lemma 2.1 *Suppose that $H : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous mapping. Then, for any $t_0 > 0$ and $w_0 \in \mathbb{R}^n$, there exists a local solution $w(t)$ to (22) with $t \in [t_0, \tau)$ for some $\tau > t_0$. If, in addition, H is locally Lipschitz continuous at x_0, then the solution is unique; if H is Lipschitz continuous in $\mathbb{R}^n$, then τ can be extended to ∞.*

Let $w(t)$ be a solution to dynamic system (22). An isolated equilibrium point w^* is Lyapunov stable if for any $w_0 = w(t_0)$ and any $\varepsilon > 0$, there exists a $\delta > 0$ such that $\|w(t) - w^*\| < \varepsilon$ for all $t \geq t_0$ and $\|w(t_0) - w^*\| < \delta$. An isolated equilibrium point w^* is said to be asymptotic stable if in addition to being Lyapunov stable, it has the property that $w(t) \to w^*$ as $t \to \infty$ for all $\|w(t_0) - w^*\| < \delta$. An isolated equilibrium point w^* is exponentially stable if there exists a $\delta > 0$ such that arbitrary point $w(t)$ of (22) with the initial condition $w(t_0) = w_0$ and $\|w(t_0) - w^*\| < \delta$ is well defined on $[0, +\infty)$ and satisfies

$$\|w(t) - w^*\| \leq ce^{-\omega t}\|w(t_0) - w^*\| \quad \forall t \geq t_0,$$

where $c > 0$ and $\omega > 0$ are constants independent of the initial point.

Let $\Omega \subseteq \mathbb{R}^n$ be an open neighborhood of $\bar{w}$. A continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ is said to be a Lyapunov function at the state $\bar{w}$ over the set Ω for Eq. (22) if

$$\begin{cases} V(\bar{w}) = 0, \ V(w) > 0, \ \forall w \in \Omega\backslash\{\bar{w}\}, \\ \dot{V}(w) \leq 0, \ \forall w \in \Omega\backslash\{\bar{w}\}. \end{cases}$$

The Lyapunov stability and asymptotical stability can be verified by using Lyapunov function, which is a useful tool for analysis.

Lemma 2.2 **(a)** *An isolated equilibrium point w^* is Lyapunov stable if there exists a Lyapunov function over some neighborhood Ω^* of w^*.*

(b) *An isolated equilibrium point w^* is asymptotically stable if there exists a Lyapunov function over some neighborhood Ω^* of w^* such that $\dot{V}(w) < 0, \ \forall w \in \Omega^*\backslash\{w^*\}$.*

3 Neural network model for AVE

In order to design a suitable neural network for absolute value Eq. (1), the key step is to construct an appropriate energy function $E(x)$ for which the global minimization x^* is simultaneously a solution of the AVE (1). One approach to constructing a desired energy function is the merit function method. The basic idea in this approach is to transform the AVE (1) into an unconstrained problem.

To this end, we define $H_i : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ as

$$H_i(\mu, x) = \begin{bmatrix} \mu \\ Ax + B\Phi_i(\mu, x) - b \end{bmatrix}, \quad \text{for } \mu \in \mathbb{R}, \text{ and } x \in \mathbb{R}^n, \tag{23}$$

where $\Phi_i : \mathbb{R}^{n+1} \to \mathbb{R}^n$ is given by

$$\Phi_i(\mu, x) := \begin{bmatrix} \phi_i(\mu, x_1) \\ \phi_i(\mu, x_2) \\ \vdots \\ \phi_i(\mu, x_n) \end{bmatrix}, \quad \text{for } \mu \in \mathbb{R}, \text{ and } x \in \mathbb{R}^n, \tag{24}$$

with various smoothing functions $\phi_i : \mathbb{R}^2 \to \mathbb{R}$ that is introduced in Sect. 2. Then, the AVE (1) can be transformed into an unconstrained optimization problem:

$$\min \Psi(\mu, x) = \frac{1}{2}\|H_i(\mu, x)\|^2. \tag{25}$$

Let $w = (\mu, x)$, the AVE (1) is equivalent to $H_i(\mu, x) = 0$. It is clear that if $w^* \in \mathbb{R}_{++} \times \mathbb{R}^n$ solves $H_i(w) = 0$, then w^* solves $\nabla\Psi(w) = 0$. Applying the gradient approach to the minimization of the energy function (25), we obtain the system of differential equation:

$$\begin{cases} \dfrac{du(t)}{dt} = -\rho\, \nabla\Psi(v(t), u(t)) = -\rho \nabla H_i(v(t), u(t))^{\mathrm{T}} H_i(v(t), u(t)), \\ u(t_0) = u_0. \end{cases} \tag{26}$$

where $u_0 = x_0 \in \mathbb{R}^n$, $v(t) = \mu_0 e^{-t}$, $\rho > 0$ is a time scaling factor. In fact, letting $\tau = \rho t$ leads to $\frac{du(t)}{dt} = \rho\frac{du(\tau)}{d\tau}$. Hence, it follows from (26) that $\frac{du(\tau)}{d\tau} = -\nabla(\frac{1}{2}\|H_i(w^*)\|^2)$. In view of this, we set $\rho = 1$ in the subsequent analysis.

Assumption 3.1 The minimal singular value of the matrix A is strictly greater than the maximal singular value of the matrix B.

Proposition 3.1 *The AVE (2) is uniquely solvable for any $b \in \mathbb{R}^n$ if Assumption 3.1 is satisfied.*

Proof Please see [9, Proposition 2.3] for a proof. □

Proposition 3.2 *Let $\Phi_i(\mu, x)$ for $i = 1, \ldots, 8$ be defined as in (24). Then, we have*

(a) *$H_i(\mu, x) = 0$ if and only if x solves the AVE (2);*
(b) *H_i is continuously differentiable on $\mathbb{R}^n \setminus \{\mathbf{0}\}$ with the Jacobian matrix given by*

$$\nabla H_i(\mu, x) := [A + B\, \nabla\Phi_i(\mu, x)], \tag{27}$$

where

$$\nabla\Phi_i(\mu, x) := \begin{bmatrix} \dfrac{\partial\phi_i(\mu, x_1)}{\partial x_1} & 0 & \cdots & 0 \\ 0 & \dfrac{\partial\phi_i(\mu, x_2)}{\partial x_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \dfrac{\partial\phi_i(\mu, x_n)}{\partial x_n} \end{bmatrix}.$$

Proof The arguments are straightforward and we omit them. □

Proposition 3.3 *Let H_i and ∇H_i be given as in (23) and (27), respectively. Suppose that Assumption 3.1 holds. Then, $\nabla H_i(\mu, x)$ is invertible at any $x \in \mathbb{R}^n$ and $\mu > 0$.*

Proof The result follows from Proposition 3.2 immediately. □

Proposition 3.4 *Let $\Psi : \mathbb{R}^n \to \mathbb{R}_+$ be given by (25). Then, the following results hold.*

(a) $\Psi(x) \geq 0, \forall(\mu, t) \in \mathbb{R}_{++} \times \mathbb{R}$ *and* $\Psi(\mu, x) = 0$ *if and only if x solve the AVE* (2).

(b) *The function $\Psi(x)$ is continuously differentiable on $\mathbb{R}^{n+1} \backslash \{\mathbf{0}\}$ with*

$$\nabla\Psi(\mu, x) = \nabla H^{\mathrm{T}} H(\mu, x),$$

where ∇H is the Jacobian of $H(\mu, x)$.

(c) *The function $\Psi(w(t))$ is nonincreasing with respect to t.*

Proof Parts (a)–(b) follow from Proposition 3.2 immediately.
For part (c), we observe that

$$\begin{aligned}\frac{d\Psi(w(t))}{dt} &= \left\langle \frac{dw}{dt}, \nabla\Psi(\mu, x)\right\rangle \\ &= \langle -\rho\, \nabla\Psi(\mu, x), \nabla\Psi(\mu, x)\rangle \\ &= -\rho\, \|\nabla\Psi(\mu, x)\|^2 < 0,\end{aligned}$$

for all $x \in \Omega\backslash\{x^*\}$. Then, the desired result follows. □

4 Stability and existence

In this section, we first adress the relation between the solution of AVE (1) and the equilibrium point of neural network (26). Then, we discuss the issues of the stability and the the solution trajectory of the neural network (26).

Lemma 4.1 *Let x^* be a equilibrium of the neural network (26) and suppose that the singular values of $A \in \mathbb{R}^{n\times n}$ exceed 1. Then x^* solves the system (1).*

Proof Since $\nabla(\Psi(w^*)) = \nabla H_i^{\mathrm{T}} H_i(w^*)$ and from the Proposition 3.3 obtain ∇H is nonsingular. It is clear to see that

$$\nabla\left(\Psi(w^*)\right) = 0,$$

if and only if $H_i(w^*) = 0$. □

Theorem 4.1 **(a)** *For any initial point $w_0 = w(t_0)$, there exists a unique continuously maximal solution $w(t)$ with $t \in [t_0, \tau)$ for the neural network (26).*

(b) *If the level set $\mathcal{L}(w_0) := \left\{w \mid \|H_i(w)\|^2 \leq \|H(w_0)\|^2\right\}$ is bounded, then τ can be extended to ∞.*

Proof This proof is exactly the same as the one in [33, Proposition 3.4], so we omit it here. □

Now, we are going to analyze the stability of an isolated equilibrium x^* of the neural network (26), which is to assert that $\nabla\Psi(x^*) = 0$ and $\nabla\Psi(x) \neq 0$ for $x \in \Omega\backslash\{x^*\}$, Ω is a neighborhood of x^*.

Theorem 4.2 *If the singular values of $A \in \mathbb{R}^{n\times n}$ exceed 1, then the isolated equilibrium x^* of the neural network (26) is asymptotically stable, and hence Lyapunov stable.*

Proof We consider the Lyapunov function $\Psi(w) : \Omega \to \mathbb{R}$ defined by (25). First, it is clear that $\Psi(x) \geq 0$ and from (a) of Proposition 3.4 we have $\Psi(\cdot)$ is continuously differentiable. Considering the singular values of A exceed 1 and Proposition 3.3 we obtain $\nabla H(w^*)$ is nonsingular. Then applying $\nabla H(x^*)$ and Lemma 4.1, we have $H(w^*) = 0$ and $\Psi(w^*) = 0$. Furthermore, if $\Psi(w) = 0$ on Ω, then $H(w) = 0$ and hence $\nabla\Psi = 0$ on Ω. This yields that $w = w^*$ since w^* is isolated.

Secondly, consider the (b) of Proposition 3.4 and Lemma 2.2, it says the isolated equilibrium x^* is asymptotically stable, and hence is Lyapunov stable. □

Theorem 4.3 *If the singular values of $A \in \mathbb{R}^{n\times n}$ exceed 1, then the isolated equilibrium x^* of the neural network (26) is exponentially stable.*

Proof The proof is routine and similar to that in the literature. For completeness, we include it again. Let $\Omega = \mathbb{R}_{++} \times \mathbb{R}^n$, it is clear that $H(\cdot)$ is continuously differentiable, which implies

$$H(w) = H(w^*) + \nabla H(w^*)^T(w - w^*) + o(\|w - w^*\|), \quad \forall\, w \in \Omega. \tag{28}$$

Let $g(t) := \frac{1}{2}\|w(t) - w^*\|^2$ and we compute the derivative of $g(t)$ as below:

$$\begin{aligned}
\frac{dg(t)}{dt} &= \left(\frac{dw}{dt}\right)^T (w(t) - w^*) = -\frac{\rho}{2}\,\nabla(\|H(w)\|^2)^T(w(t) - w^*)\\
&= -\rho\,(\nabla H(w)\cdot H(w))^T(w(t) - w^*)\\
&= -\rho\, H(w)^T\nabla H(w)^T(w(t) - w^*)\\
&= -\rho\,(w(t) - w^*)^T\nabla H(w^*)\nabla H(w)^T(w(t) - w^*)\\
&\quad -\rho\, o(\|w - w^*\|)^T\nabla H(w)^T(w(t) - w^*),
\end{aligned}$$

where the last equality is due to (28). To proceed, we claim two assertions. First, we claim that $(w - w^*)^T\nabla H(w^*)\nabla H(w)^T(w - w^*) \geq \kappa||w - w^*||^2$, for some κ. To see this, from the Propositions 3.3 and 3.4, we know $\nabla H(w)$ is nonsingular and H is a continuously differentiable function, which implies the matrix $\nabla H(w^*)\nabla H(w^*)^T$ is symmetric and positive semi-definite. Hence, we have $(w - w^*)^T\nabla H(w^*)\nabla H(w^*)^T(w - w^*) \geq \kappa_1\|w - w^*\|^2 > 0$ over $\Omega\backslash\{w^*\}$ for some $\kappa_1 \geq 0$. Then, by the continuity of $\nabla H(\cdot)$, we can conclude that

$$(w - w^*)^T\nabla H(w^*)\nabla H(w)^T(w - w^*) \geq \kappa\|w - w^*\|^2 > 0, \quad \text{for some } \kappa \geq 0.$$

Secondly, we claim that

$$-\rho\, o(\|w-w^*\|)^T \nabla H(w)^T (w(t)-w^*) \leq \varepsilon \|w-w^*\|^2, \quad \text{for some } \varepsilon > 0.$$

This is because that

$$\frac{\big| -\rho\, o(\|w-w^*\|)^T \nabla H(w)^T (w(t)-w^*)\big|}{\|w-w^*\|^2} \leq \rho \|\nabla H(w)\| \left(\frac{\|o(\|w-w^*\|)\|}{\|w-w^*\|} \right),$$

where the right-hand side vanishes eventually. Thus, it yields that

$$-\rho\, o(\|w-w^*\|)^T \nabla H(w)^T (w(t)-w^*) \leq \varepsilon \|w-w^*\|^2, \quad \text{for some } \varepsilon > 0.$$

Now, from the above two assertions and noting that $g(t) = \frac{1}{2}\|w(t)-w^*\|^2$, we have

$$\frac{dg(t)}{dt} \leq 2(-\rho\kappa + \varepsilon) g(t),$$

which gives

$$g(t) \leq e^{2(-\rho\kappa+\varepsilon)t} g(t_0).$$

Thus, we have

$$\|w(t)-w^*\| \leq e^{(-\rho\kappa+\varepsilon)t} \|w(t_0)-w^*\|,$$

which says w^* is exponentially stable as we can set ρ larger enough such that $-\rho\kappa + \varepsilon < 0$. Then, the proof is complete. □

5 Numerical results

In order to demonstrate the effectiveness of the proposed neural network, we test several examples for our neural network (26) in this section. The numerical implementation is coded by Mathematica 11.3 and the ordinary differential equation solver adopted is NDSolve[ ], which uses an Runge–Kutta (2,3) formula. The initial point of each problems are selected by randomly and the initial point is same for different smoothing functions. The results are collected together in Tables 1, 2, 3 and 4, where

ϕ_i	Denotes the smoothing functions ϕ_i, $i = 1, \ldots, 8$
N	Denotes the number of iterations
t	Denotes the time when algorithm terminates
Er	Denotes the value of $\|x(t) - x^*\|$ when algorithm terminates
$H(x_t)$	Denotes the value of $\|H(x(t)) = \|Ax - |x| - b\|$ when algorithm terminates
CT	Denotes the CPU time in seconds

Table 1 Computing results of Example 5.1 (dt = 0.2)

Function	N	t	Er	$H(x_0)$	CT
ϕ_1	34	6.8	9.3686×10^{-7}	0.0000136037	1.5090863
ϕ_2	36	7.2	8.70587×10^{-7}	0.0000126414	0.7760444
ϕ_3	38	7.6	8.41914×10^{-7}	0.000012225	0.4980285
ϕ_4	2	0.4	2.90785×10^{-15}	1.59872×10^{-14}	0.0340019
ϕ_5	2	0.4	1.11772×10^{-12}	8.41527×10^{-12}	0.0740043
ϕ_6	10	2.0	7.52691×10^{-7}	0.0000109295	0.1150066
ϕ_7	2	0.4	1.29976×10^{-12}	8.61527×10^{-12}	0.0730042
ϕ_8	34	6.8	9.3686×10^{-7}	0.0000136037	0.6880393

Table 2 Computing results of Example 5.2 (dt = 0.2)

Function	N	T	Er	$H(x_0)$	CT
ϕ_1	55	11	9.84216×10^{-7}	2.03995×10^{-7}	3.1531804
ϕ_2	57	11.4	9.14593×10^{-7}	1.89565×10^{-7}	1.3690783
ϕ_3	59	11.8	8.84473×10^{-7}	1.83322×10^{-7}	0.6920396
ϕ_4	2	0.4	2.67859×10^{-9}	4.86096×10^{-10}	0.0370021
ϕ_5	2	0.4	2.68658×10^{-9}	4.87548×10^{-10}	0.1510086
ϕ_6	18	3.6	9.56666×10^{-7}	2.57215×10^{-7}	0.2210127
ϕ_7	2	0.4	2.16413×10^{-9}	3.92736×10^{-10}	0.1600091
ϕ_8	55	11	9.84216×10^{-7}	2.03995×10^{-7}	1.5320877

Example 5.1 Consider the following absolute value equation where

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

We can verify that one solution of the absolute value equations is $x^* = (1, 1, 1, 1)$. The parameter ρ is set to be 1, time step is set to be $dt = 0.2$ and the initial point is generated by randomly. Table 1 summarizes the computing results for Example 5.1. From Table 1, we see that nor matter from the trajectory convergence time, the error, or the computation time, the smoothing function ϕ_4, ϕ_5, ϕ_7 perform significantly better than other functions. Figure 3 depicts the norm error $\|x(t) - x^*\|$ with various time. This figure indicates the smoothing functions ϕ_4, ϕ_5, ϕ_7 also outperform than others (it follows by ϕ_6).

Table 3 Computing results of Example 5.3 (dt = 0.2)

Function	N	T	Er	$H(x_0)$	CT
ϕ_1	48	9.6	8.77421×10^{-7}	8.27241×10^{-7}	2.6401510
ϕ_2	49	9.8	9.95875×10^{-7}	9.3892×10^{-7}	1.0670610
ϕ_3	51	10.2	9.63078×10^{-7}	9.07998×10^{-7}	0.6060347
ϕ_4	9	1.8	1.88878×10^{-8}	8.88851×10^{-9}	0.2820161
ϕ_5	9	1.8	1.89870×10^{-8}	8.93517×10^{-9}	0.3770216
ϕ_6	15	3.0	7.23527×10^{-7}	1.06615×10^{-6}	0.1870107
ϕ_7	9	1.8	1.85496×10^{-8}	8.72934×10^{-9}	0.4020229
ϕ_8	48	9.6	8.77421×10^{-7}	8.27241×10^{-7}	1.2330706

Table 4 Computing results of Example 5.4

Function	N	T	Er	$H(x_0)$	CT
ϕ_1	58	5.8	9.97651×10^{-7}	0.000118298	465.6496337
ϕ_2	62	6.2	9.27099×10^{-7}	0.000109929	191.1859352
ϕ_3	65	6.5	9.90869×10^{-7}	0.000117487	162.2222786
ϕ_4	4	0.4	1.48112×10^{-7}	5.51822×10^{-7}	11.9286822
ϕ_5	4	0.4	1.49218×10^{-7}	5.55955×10^{-7}	32.2308435
ϕ_6	14	1.4	8.85812×10^{-7}	0.000105037	35.2450159
ϕ_7	4	0.4	1.48181×10^{-7}	5.52006×10^{-7}	33.0088880
ϕ_8	58	5.8	9.97689×10^{-7}	0.000118298	207.7848846

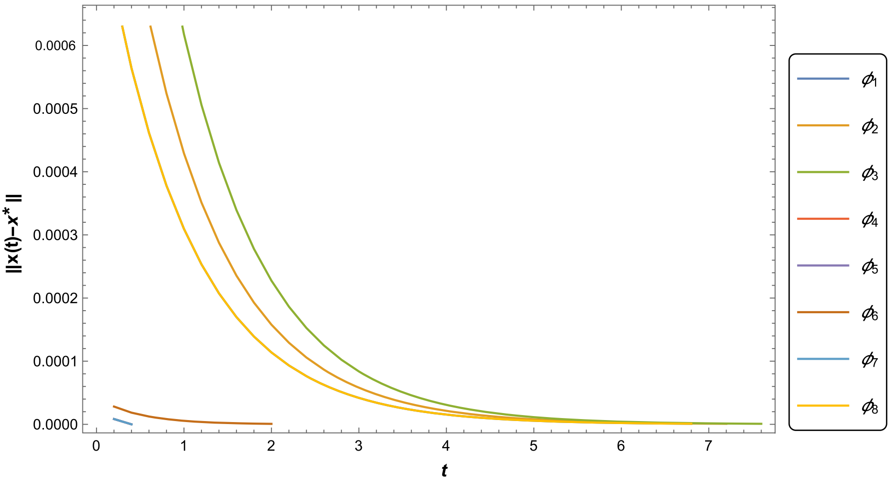

Fig. 3 Convergence behaviour of the error $\|x(t) - x^*\|$ for ϕ_i, $i = 1, \ldots, 8$, in Example 5.1 (dt = 0.2)

Example 5.2 Consider the following linear complementary problem: find $x \in \mathbb{R}^n$ such that

$$x \geq 0, \quad Mx + q \geq 0, \quad \langle x, Mx + q \rangle = 0,$$

where

$$M = \begin{pmatrix} 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad q = \begin{pmatrix} -1 \\ -1 \\ -1 \\ -1 \end{pmatrix}.$$

Because of 1 is the eigenvalue of M, we use the idea given as in [37] in which multiplying by positive constant $\lambda = 3$. Then, we transform the new linear complementary problem into the following AVE by the method introduced in [14]:

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

Again, we can verify that one solution of the absolute value equation is $x^* = (-3, 3, 3, -1)$. The parameter ρ is set to be 1000, time step is set to be $dt = 0.2$ and the initial point is generated by randomly. Table 2 presents the computing results for Example 5.2 and Fig. 4 demonstrates the norm error $\|x(t) - x^*\|$ with various time. From Table 2 and Fig. 4, we also see that the smoothing functions ϕ_4, ϕ_5, ϕ_7 perform better than others (followed by ϕ_6). This phenomenon is similar to that appeared in Example 5.1.

Example 5.3 Consider the following linear complementary problem: find $x \in \mathbb{R}^n$ such that

$$x \geq 0, \quad Mx + q \geq 0, \quad \langle x, Mx + q \rangle = 0,$$

where

$$M = \begin{pmatrix} 1 & -4 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}, \quad q = \begin{pmatrix} -5 \\ -5 \\ 1 \\ 1 \end{pmatrix}.$$

Likewise, we can transform this linear complementary problem into an AVE by the method introduced in [14], where

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

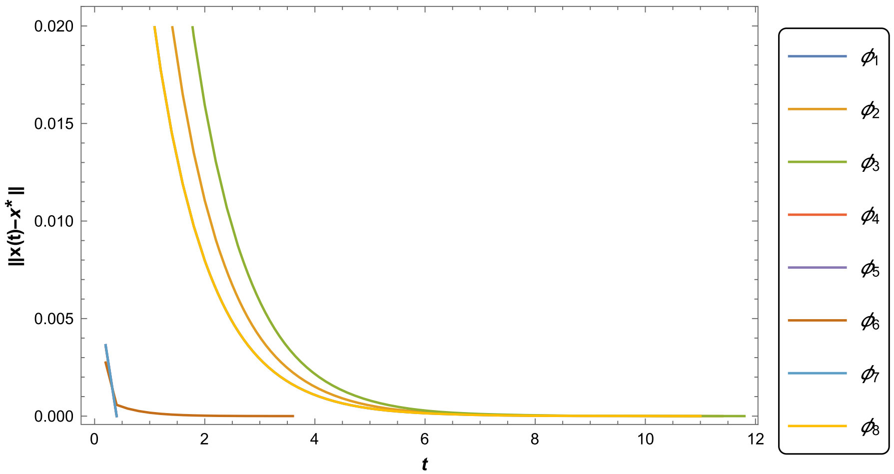

Fig. 4 Convergence behaviour of the error $\|x(t) - x^*\|$ for ϕ_i, $i = 1, \ldots, 8$, in Example 5.2

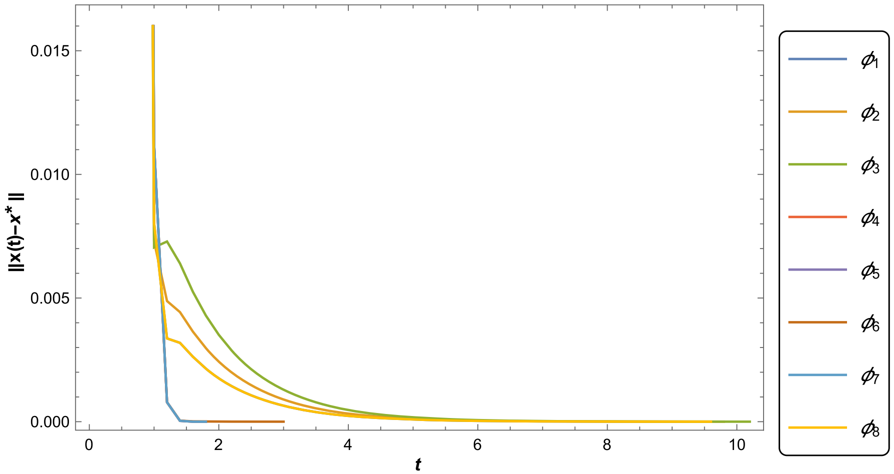

Fig. 5 Convergence behaviour of the error $\|x(t) - x^*\|$ for ϕ_i, $i = 1, \ldots, 8$, in Example 5.3

One solution of this absolute value equation is $x^* = (-1, -1, -8, -4)$. The parameter ρ is set to be 10, time step is set to be $dt = 0.2$ and the initial point is generated by randomly. Table 3 summarizes the computing results for Example 5.3 and Fig. 5 depicts the norm error $\|x(t) - x^*\|$ with various time. From Table 3 and Fig. 5, we see that performance of the smoothing function $\phi_4, \phi_5, \phi_6, \phi_7$ is better than others.

Example 5.4 Consider the AVE, where the matrix A of which all the singular values are > 1 is generated by the following Mathematica procedure:

```
R = RandomInteger[{0,50},{n,n}];
A = R.R + n*IdentityMatrix[n];
b = (A - IdentityMatrix[n]).Table[1,n];
```

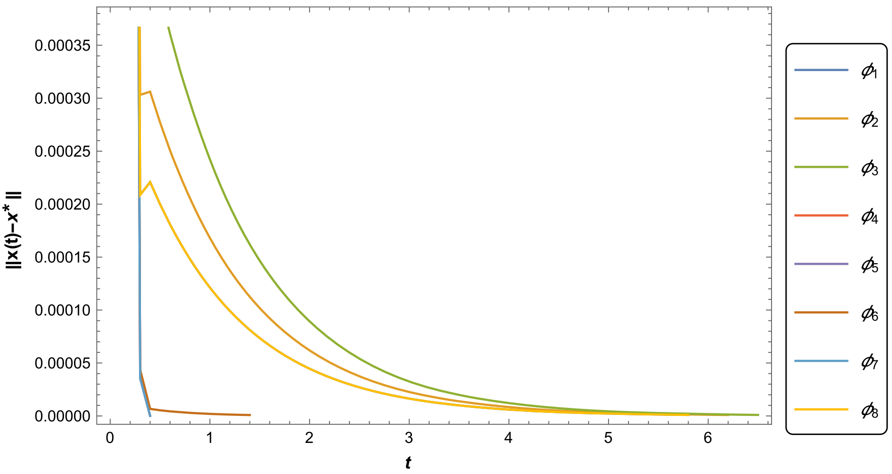

Fig. 6 Convergence behaviour of the error $\|x(t) - x^*\|$ for ϕ_i, $i = 1, \ldots, 8$, in Example 5.4

The parameter ρ is set to be 1, time step is set to be $dt = 0.1$ and the initial point is generated by randomly. When the dimension $n = 50$, In Table 4 presents the computing results for Example 5.4 and Fig. 6 shows the norm error $\|x(t) - x^*\|$ with various time. For this AVE with high dimension, the smoothing functions ϕ_4, ϕ_5, ϕ_6, ϕ_7 is again the leading group regarding the efficiency.

6 Concluding remarks

In this conclusion section, we summarize our findings based from the preceding simulations. In general, we can conclude that the smoothing functions $\phi_4, \phi_5, \phi_6, \phi_7$ are effective functions that work well along with the neural network (26). In particular, all these smoothing functions are produced from the convolution way. The other ways like convex conjugate way, Moreau proximal way, Nesterov's smoothing way, and infimal-convolution way, do not offer effective smoothing functions for the proposed neural network approach. This is a very interesting discovery, which deserves further investigation. For example, what kind of factor causes this phenomenon theoretically? Is similar phenomenon occurs in other algorithms? We leave them for our future study.

To close this section, we point out one observation. If we are given two smoothing functions ψ_1 and ψ_2 for f, then $t\psi_1 + (1-t)\psi_2$ is also a smoothing function for f. This means that any convex combination of two smoothing functions for $|x|$ is again a smoothing function for $|x|$. In particular, we choose $\psi_1 \in \{\phi_1, \phi_2, \phi_3\}$ and pick another $\psi_2 \in \{\phi_4, \phi_5, \phi_6, \phi_7, \phi_8\}$ to make new smoothing functions for $|x|$ (through 15 convex combinations and with different value $t \in [0, 1]$). In other words, we can obtain many more smoothing functions. How do these types of smoothing functions perform? We leave it as our future study.

References

1. Bouzerdoum, A., Pattison, T.R.: Neural network for quadratic optimization with bound constraints. IEEE Trans. Neural Netw. **4**, 293–304 (1993)
2. Beck, A., Teboulle, M.: Smoothing and first order methods: a unified framework. SIAM J. Optim. **22**, 557–580 (2012)
3. Bauschke, H., Combettes, P.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer, New York (2016)
4. Chen, C., Mangasarian, O.L.: A class of smoothing functions for nonlinear and mixed complementarity problems. Comput. Optim. Appl. **5**, 97–138 (1996)
5. Cichochi, A., Unbenhauen, R.: Neural Network for Optimization and Signal Processing. Wiley, New York (1993)
6. Hopfield, J.J., Tank, D.W.: Neural computation of decision in optimization problems. Biol. Cybern. **52**, 141–152 (1985)
7. Hu, S.-L., Huang, Z.-H.: A note on absolute value equations. Optim. Lett. **4**, 417–424 (2010)
8. Hu, S.-L., Huang, Z.-H., Zhang, Q.: A generalized Newton method for absolute value equations associated with second order cones. J. Comput. Appl. Math. **235**, 1490–1501 (2011)
9. Jiang, X., Zhang, Y.: A smoothing-type algorithm for absolute value equations. J. Ind. Manag. Optim. **9**, 789–798 (2013)
10. Ketabchi, S., Moosaei, H.: Minimum norm solution to the absolute value equation in the convex case. J. Optim. Theory Appl. **154**, 1080–1087 (2012)
11. Kreimer, J., Rubinstein, R.Y.: Nondifferentiable optimization via smooth approximation: general analytical approach. Ann. Oper. Res. **39**, 97–119 (1992)
12. Liao, L.-Z., Qi, H.-D.: A neural network for the linear complementarity problem. Math. Comput. Model. **29**, 9–18 (1999)
13. Liao, L., Qi, H.-D., Qi, L.: Solving nonlinear complementarity problem with neural networks: a reformulation method approach. J. Comput. Appl. Math. **131**, 343–359 (2001)
14. Mangasarian, O.L.: Absolute value programming. Comput. Optim. Appl. **36**, 43–53 (2007)
15. Mangasarian, O.L.: Absolute value equation solution via concave minimization. Optim. Lett. **1**, 3–5 (2007)
16. Mangasarian, O.L.: A generalized Newton method for absolute value equations. Optim. Lett. **3**, 101–108 (2009)
17. Mangasarian, O.L.: Primal–dual bilinear programming solution of the absolute value equation. Optim. Lett. **6**, 1527–1533 (2012)
18. Mangasarian, O.L.: Absolute value equation solution via dual complementarity. Optim. Lett. **7**, 625–630 (2013)
19. Miao, X.-H., Hsu, W.-M., Chen, J.-S.: The solvabilities of three optimization problems associated with second-order cone (submitted manuscript) (2018)
20. Miao, X.-H., Yang, J.-T., Saheya, B., Chen, J.-S.: A smoothing Newton method for absolute value equation associated with second-order cone. Appl. Numer. Math. **120**(October), 82–96 (2017)
21. Mangasarian, O.L., Meyer, R.R.: Absolute value equation. Linear Algebra Appl. **419**, 359–367 (2006)
22. Moreau, J.J.: Proximité et dualité dans un espace Hilbertien. Bulletin de la Société Mathématique de France **93**, 273–299 (1965)
23. Nesterov, Y.: Smooth minimization of non-smooth functions. Math. Program. **103**, 127–152 (2005)
24. Nguyen, C.T., Saheya, B., Chang, Y.-L., Chen, J.-S.: Unified smoothing functions for absolute value equation associated with second-order cone. Appl. Numer. Math. **135**, 206–227 (2019)
25. Prokopyev, O.A.: On equivalent reformulations for absolute value equations. Comput. Optim. Appl. **44**, 363–372 (2009)
26. Qi, L.: Convergence analysis of some algorithms for solving nonsmooth equations. Math. Oper. Res. **18**, 227–244 (1993)
27. Qi, L., Sun, D., Zhou, G.-L.: A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequality problems. Math. Program. **87**, 1–35 (2000)
28. Qi, L., Sun, D.: Smoothing functions and smoothing Newton method for complementarity and variational inequality problems. J. Optim. Theory Appl. **113**, 121–147 (2002)
29. Rohn, J.: A theorem of the alternatives for the equation $Ax + B|x| = b$. Linear Multilinear Algebra **52**, 421–426 (2004)

30. Rohn, J.: Solvability of systems of interval linear equations and inequalities. In: Fiedler, M., Nedoma, J., Ramik, J., Rohn, J., Zimmermann, K. (eds.) Linear Optimization Problems with Inexact Data, pp. 35–77. Springer, Berlin (2006)
31. Rohn, J.: An algorithm for solving the absolute value equation. Electron. J. Linear Algebra **18**, 589–599 (2009)
32. Saheya, B., Yu, C.-H., Chen, J.-S.: Numerical comparisons based on four smoothing functions for absolute value equation. J. Appl. Math. Comput. **56**, 131–149 (2018)
33. Sun, J.-H., Chen, J.-S., Ko, C.-H.: Neural networks for solving second-order cone constrained variational inequality problem. Comput. Optim. Appl. **51**, 623–648 (2012)
34. Tank, D.W., Hopfield, J.J.: Simple neural optimization networks: an A/D converter, signal decision circuit, and a linear programming circuit. IEEE Trans. Circuits Syst. **33**, 533–541 (1986)
35. Voronin, S., Ozkaya, G., Yoshida, D.: Convolution based smooth approximations to the absolute value function with application to non-smooth regularization. arXiv:1408.6795v2 [math.NA]
36. Wang, F., Yu, Z., Gao, C.: A smoothing neural network algorithm for absolute value equations. Engineering **7**, 567–576 (2015)
37. Yong, L.Q., Liu, S.Y., Tuo, S.H.: Transformation of the linear complementarity problem and the absolute value equation. J. Jilin Univ. (Sci. Ed.) **4**, 638–686 (2014)
38. Yamanaka, S., Fukushima, M.: A branch and bound method for the absolute value programs. Optimization **63**, 305–319 (2014)
39. Zak, S.H., Upatising, V., Hui, S.: Solving linear programming problems with neural networks: a comparative study. IEEE Trans. Neural Netw. **6**, 94–104 (1995)