Neurocomputing 173 (2016) 988–993

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A neural network for $\ell_1-\ell_2$ minimization based on scaled gradient projection: Application to compressed sensing ☆

Yongwei Liu, Jianfeng Hu *

School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China

ARTICLE INFO

Article history:
Received 24 January 2015
Received in revised form
1 July 2015
Accepted 23 August 2015
Available online 8 September 2015

Keywords:
Neural network
Scaled gradient projection
$\ell_1-\ell_2$ minimization
Compressed sensing

ABSTRACT

Since compressed sensing was introduced in 2006, $\ell_1-\ell_2$ minimization admits a large number of applications in signal processing, statistical inference, magnetic resonance imaging (MRI), computed tomography (CT), etc. In this paper, we present a neural network for $\ell_1-\ell_2$ minimization based on scaled gradient projection. We prove that it is stable in the sense of Lyapunov and converges to an optimal solution of the $\ell_1-\ell_2$ minimization. We show that the proposed neural network is feasible and efficient for compressed sensing via simulation examples.

1. Introduction

It is well-known that the sparsest solution of an underdetermined linear system of equations can be found by solving the so-called "ℓ_0-norm" minimization problem, i.e.,

$$\min_{\boldsymbol{x}\in R^n} \|\boldsymbol{x}\|_0 \quad \text{s. t.} \quad \boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}, \tag{1}$$

where $\boldsymbol{A}\in R^{k\times n}(k<n)$, $\boldsymbol{b}\in R^k$, $\|\boldsymbol{x}\|_0$ denotes the number of nonzero components in $\boldsymbol{x}$. The problem (1) has a wide range of applications in signal processing. Unfortunately, it is NP-hard [1]. By replacing the ℓ_0-norm by the ℓ_1-norm, an approximation model named "Basic Pursuit problem" [2] was proposed as following:

$$\min_{\boldsymbol{x}\in R^n} \|\boldsymbol{x}\|_1 \quad \text{s. t.} \quad \boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}, \tag{2}$$

since the convex envelope of $\|\boldsymbol{x}\|_0$ is $\|\boldsymbol{x}\|_1$, where $\|\boldsymbol{x}\|_1=\sum_i|\boldsymbol{x}_i|$ is the ℓ_1 norm of $\boldsymbol{x}$.

In 2006, Donoho [3,4] and Candès et al. [5–8] almost simultaneously proposed the compressed sensing (CS) theory. It shows that a sparse original signal $\boldsymbol{x}^0\in R^n$ (i. e., $\|\boldsymbol{x}^0\|_0 \ll n$) can be recovered from very few linear measurements (i. e., $k\ll n$) by solving (2) when the observation matrix $\boldsymbol{A}$ has the restricted isometry property (RIP). Since then, CS has attracted considerable attention in the areas of applied mathematics, computer science and electrical engineering.

In problem (2), if $\boldsymbol{b}$ is contaminated by small dense noise (e.g., Gaussian noise), a natural approach is to relax the constraint in (2) to yield the following problem:

$$\min_{\boldsymbol{x}\in R^n} \|\boldsymbol{x}\|_1 \quad \text{s. t.} \quad \|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|_2<\epsilon, \tag{3}$$

where ϵ is nonnegative parameter and $\|\boldsymbol{x}\|_2$ is the Euclidean norm of $\boldsymbol{x}$. Further, convex analysis shows that (3) is equivalent to its Lagrangian version, i.e., the following $\ell_1-\ell_2$ minimization

$$\min_{\boldsymbol{x}\in R^n} \frac{1}{2}\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\|_2^2+\tau\|\boldsymbol{x}\|_1, \tag{4}$$

for some $\tau>0$. Obviously, the problem (4) is convex, but not differentiable. Other applications of $\ell_1-\ell_2$ minimization can be found in MRI and CT [9,10], face recognition [11,12], denoising [13], inpainting [14], variable selection [15] and LASSO [16].

Nowadays, some numerical iterative algorithms for $\ell_1-\ell_2$ minimization have been introduced, such as interior-point algorithms [17,18], iterative shrinkage thresholds algorithms [19,20], matching pursuit [21,22], GPSR algorithms [23], fixed point method [24], linearized Bregman iterations [25,26] and alternating direction algorithms [27].

However, in practical applications, many problems have to be solved in real-time. Since the time complexity of a solution greatly depends on the dimension and the structure of problems, numerical iterative algorithms for digital computers are usually less efficient and cannot satisfy real-time requirement. Due to the

☆Supported by Natural Science Foundation of Hainan Province of China (No. 114006).

* Corresponding author.
E-mail addresses: liuyongwei2012@163.com (Y. Liu), lakerhjf@163.com (J. Hu).

http://dx.doi.org/10.1016/j.neucom.2015.08.055
0925-2312/

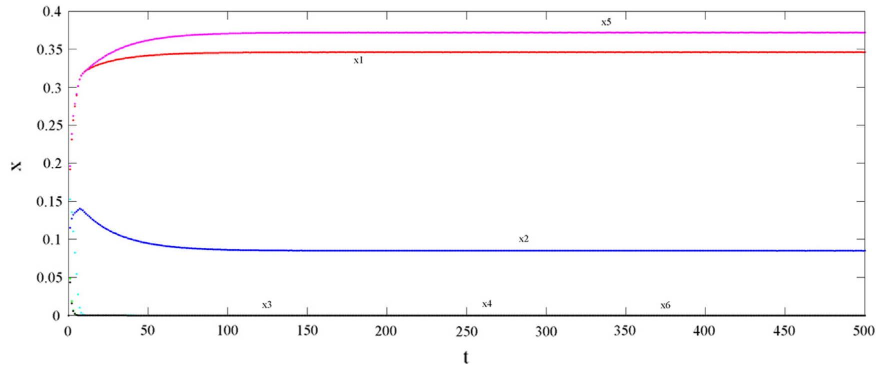

Fig. 1. The solution of the neural network (6) for Example 1.

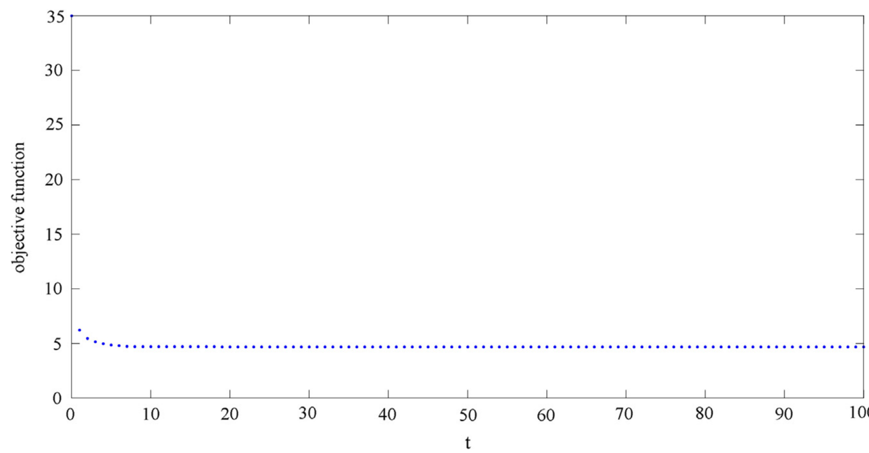

Fig. 2. Transient of objective function in Example 1.

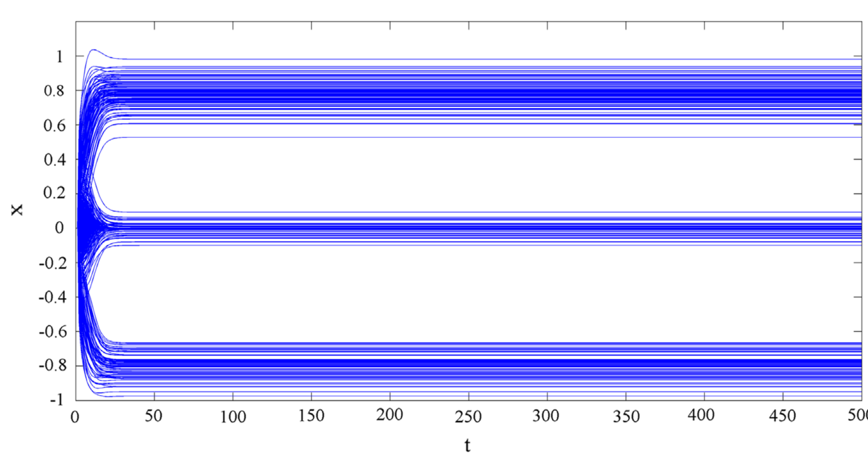

Fig. 3. Transient behavior of Example 2.

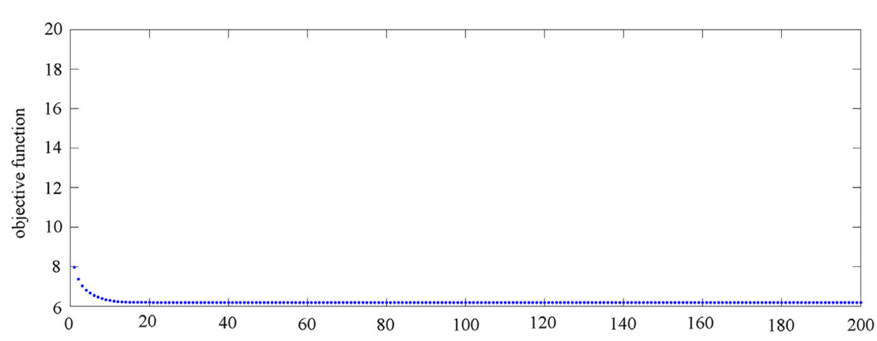

Fig. 4. Transient of objective function in Example 2.

massively paralleled distributed computation and fast convergence, the artificial neural networks can be more efficient approach to solve the real-time optimization problems. Therefore, it is attractive to establish a neural network for solving $\ell_1-\ell_2$ minimization problem.

The introduction of the artificial neural networks in optimization was stated in 1980. Since then, significant achievements have been made. For example, Chua and Lin [28], Tank and Hopfield [29] presented a neural network for solving linear programming problems respectively, though equilibrium points did not correspond to optimal solutions. Kennedy and Chua [30] proposed a class of neural networks for solving nonlinear programming problems based on penalty function method. Zhang and Constantinides [31], Effati and Baymani [32] proposed a neural network for solving constrained optimization problems by the Lagrange multiplier theory, however, it requires the objective function be strictly convex for convergence. In addition, by applying dual theory and gradient projection, many neural networks were presented for solving linear programming, quadratic programming and convex programming, and proved to be globally convergent to an exact optimal solution [33–38]. Motivated by [23] and [33,38], in this paper, we propose a new neural network model based on scaled gradient projection for $\ell_1-\ell_2$ minimization.

The remaining part of this paper is organized as follow. In Section 2, a neural network model is established by using scaled gradient projection. The existence and convergence of the proposed neural network solution are discussed, and the stability of equilibrium points is proved. Furthermore, the equilibrium points of the neural network are proved to be convergent to an exact solution of original optimization problem. In Section 3, the illustrative examples are given to show the feasibility and efficiency of the proposed neural network for compressed sensing. Finally, we conclude the paper in Section 4.

2. The neural network model and stability analysis

Although the objective function of (4) is not differentiable, fortunately, (4) can be transformed into a convex quadratic program [23,39]. We shall establish a neural network model to solve the quadratic program problem based on scaled gradient projection in the following.

Let

$$u_i=(x_i)_+,\ v_i=(-x_i)_+,\quad i=1,2,\dots,n,$$

where $(x_i)_+=\max\{x_i,0\}$. That is $\boldsymbol{x}=\boldsymbol{u}-\boldsymbol{v}$ and $\|\boldsymbol{x}\|_1=\mathbf{1}_n^T\boldsymbol{u}+\mathbf{1}_n^T\boldsymbol{v}$, where $\mathbf{1}_n=(1,\dots,1)^T$ is the vector consisting of n ones.

Therefore, the nondifferentiable problem (4) can be rewritten as follows:

$$\min_{\boldsymbol{u},\boldsymbol{v}\in R^n}\frac{1}{2}\|\boldsymbol{b}-\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{v})\|_2^2+\tau\mathbf{1}_n^T\boldsymbol{u}+\tau\mathbf{1}_n^T\boldsymbol{v},$$
$$\text{s. t. } \boldsymbol{u}\ge 0,$$
$$\boldsymbol{v}\ge 0.$$

The above problem is equivalent to the convex quadratic program:

$$\min_{\boldsymbol{z}\ge 0}\ F(\boldsymbol{z})=\frac{1}{2}\boldsymbol{z}^T\boldsymbol{B}\boldsymbol{z}+\boldsymbol{c}^T\boldsymbol{z},\tag{5}$$

where

$$\bar{\boldsymbol{b}}=\boldsymbol{A}^T\boldsymbol{b},\ \boldsymbol{z}=\begin{pmatrix}\boldsymbol{u}\\ \boldsymbol{v}\end{pmatrix},\ \boldsymbol{c}=\tau\mathbf{1}_{2n}+\begin{pmatrix}-\bar{\boldsymbol{b}}\\ \bar{\boldsymbol{b}}\end{pmatrix},\ \boldsymbol{B}=\begin{pmatrix}\boldsymbol{A}^T\boldsymbol{A} & -\boldsymbol{A}^T\boldsymbol{A}\\ -\boldsymbol{A}^T\boldsymbol{A} & \boldsymbol{A}^T\boldsymbol{A}\end{pmatrix}.$$

For solving problem (5), we establish the following neural network based on scaled gradient projection:

$$\begin{cases}\dfrac{d\boldsymbol{z}}{dt}=-\boldsymbol{z}+P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))\\ \boldsymbol{z}(t_0)=\boldsymbol{z}_0\in\Omega,\end{cases}\tag{6}$$

where Ω is the positive orthant, $\boldsymbol{D}$ is any given positive diagonal matrix, i.e., $\boldsymbol{D}=diag(d_1,\dots,d_{2n})$ and $d_i>0$ $(i=1,\dots,2n)$, and $P_\Omega(\boldsymbol{z})$ is the projection of $\boldsymbol{z}$ on Ω, i.e., $P_\Omega(\boldsymbol{z})=(P_\Omega(z_1),\dots,P_\Omega(z_{2n}))^T$ and $P_\Omega(z_i)=(z_i)_+$.

In the following, we show that an equilibrium point of (6) is also an optimal solution of (5) and prove the stability of equilibrium points. Furthermore, we prove that the solution of (6) will converge to a point in the set of equilibrium points.

Theorem 1. *Let Ω^* be the set of optimal solutions of (5) and Ω^e be the set of the equilibrium points of (6). Then we have $\Omega^*=\Omega^e$.*

Proof. By the optimality condition of convex programming with positive orthant constraint [40], it follows that $\boldsymbol{z}^*=(z_1^*,z_2^*,\dots,z_{2n}^*)^T\in\Omega^*$ if and only if

$$\begin{cases}\frac{\partial F(\boldsymbol{z}^*)}{\partial z_i}=0, & \text{if } z_i^*>0,\\ \frac{\partial F(\boldsymbol{z}^*)}{\partial z_i}\ge 0, & \text{if } z_i^*=0.\end{cases}$$

So, for any $\boldsymbol{z}^*\in\Omega^*$, we can easily obtain

$$-\boldsymbol{z}^*+P_\Omega(\boldsymbol{z}^*-\boldsymbol{D}\nabla F(\boldsymbol{z}^*))=0,$$

that is, $\boldsymbol{z}^*\in\Omega^e$. This means that $\Omega^*\subseteq\Omega^e$.

On the contrary, for any $\boldsymbol{z}^*\in\Omega^e$, it follows that

$$\boldsymbol{z}^*=P_\Omega(\boldsymbol{z}^*-\boldsymbol{D}\nabla F(\boldsymbol{z}^*))\ge 0,$$

i.e.,

$$z_i^*=P_\Omega\left(\boldsymbol{z}_i^*-d_i\frac{\partial F(\boldsymbol{z}^*)}{\partial z_i}\right)\ge 0,\quad i=1,\dots,2n.$$

Since $d_i>0$, it is easy to see that $\frac{\partial F(\boldsymbol{z}^*)}{\partial z_i}=0$ if $\boldsymbol{z}_i^*>0$ and $\frac{\partial F(\boldsymbol{z}^*)}{\partial z_i}\ge 0$ if $\boldsymbol{z}_i^*=0$. Therefore $\boldsymbol{z}^*\in\Omega^*$. It follows that $\Omega^e\subseteq\Omega^*$. Thus we have $\Omega^*=\Omega^e$.□

Theorem 1 shows that $\boldsymbol{z}$ is an equilibrium point of (6) if and only if $\boldsymbol{z}$ is an optimal point of (5). In the following, we prove the stability of equilibrium points of the neural network (6).

Theorem 2. *For any initial point $\boldsymbol{z}(t_0)$, the equilibrium point of the neural network (6) is stable in the sense of Lyapunov.*

Proof. Let $\boldsymbol{z}^*\in\Omega^e$. From Theorem 1, it follows that $\boldsymbol{z}^*\in\Omega^*$. Since $\boldsymbol{z}\in\Omega$ if and only if $\boldsymbol{D}^{-1/2}\boldsymbol{z}\in\Omega$, where $\boldsymbol{D}^{-1/2}=diag(1/\sqrt{d_1},\dots,1/\sqrt{d_{2n}})$, then by the projection theorem [40], we have

$$(\bar{\boldsymbol{z}}-P_\Omega(\bar{\boldsymbol{z}}))^T(P_\Omega(\bar{\boldsymbol{z}})-\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{z}^*)\ge 0,\quad\forall\,\bar{\boldsymbol{z}}\in R^{2n}.$$

It is easy to see that $P_\Omega(\bar{\boldsymbol{z}})=\boldsymbol{D}^{1/2}P_\Omega(\boldsymbol{D}^{1/2}\bar{\boldsymbol{z}})$. Thus we obtain

$$\left(\boldsymbol{D}^{\frac{1}{2}}\bar{\boldsymbol{z}}-P_\Omega(\boldsymbol{D}^{\frac{1}{2}}\bar{\boldsymbol{z}})\right)^T\boldsymbol{D}^{-1}\left(P_\Omega(\boldsymbol{D}^{\frac{1}{2}}\bar{\boldsymbol{z}})-\boldsymbol{z}^*\right)\ge 0\quad\forall\,\bar{\boldsymbol{z}}\in R^{2n}.\tag{7}$$

Let $\bar{\boldsymbol{z}}=\boldsymbol{D}^{-1/2}\boldsymbol{z}-\boldsymbol{D}^{1/2}\nabla F(\boldsymbol{z})$ in (7). It follows that

$$(-\dot{\boldsymbol{z}}-\boldsymbol{D}\nabla F(\boldsymbol{z}))^T\boldsymbol{D}^{-1}(\dot{\boldsymbol{z}}+\boldsymbol{z}-\boldsymbol{z}^*)\ge 0,\quad\forall\,\boldsymbol{z}\in\Omega,\tag{8}$$

therefore

$$(\nabla F(\boldsymbol{z})+\boldsymbol{D}^{-1}(\boldsymbol{z}-\boldsymbol{z}^*))^T\dot{\boldsymbol{z}}\le-\dot{\boldsymbol{z}}\boldsymbol{D}^{-1}\dot{\boldsymbol{z}}-\nabla F^T(\boldsymbol{z})(\boldsymbol{z}-\boldsymbol{z}^*),\quad\forall\,\boldsymbol{z}\in\Omega,\tag{9}$$

where $\dot{\boldsymbol{z}}=\frac{d\boldsymbol{z}}{dt}=-\boldsymbol{z}+P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))$.

Further, by the optimality condition $\nabla F^T(\boldsymbol{z})(\boldsymbol{z}-\boldsymbol{z}^*)\ge 0$ and (9), we have

$$\left(\nabla F(\boldsymbol{z})+\boldsymbol{D}^{-1}(\boldsymbol{z}-\boldsymbol{z}^*)\right)^T\dot{\boldsymbol{z}}\le-\dot{\boldsymbol{z}}\boldsymbol{D}^{-1}\dot{\boldsymbol{z}}\le 0,\ \forall\,\boldsymbol{z}\in\Omega.$$

We construct the Lyapunov function as following:

$$V(\boldsymbol{z})=F(\boldsymbol{z})-F(\boldsymbol{z}^*)+\frac{1}{2}(\boldsymbol{z}-\boldsymbol{z}^*)^T\boldsymbol{D}^{-1}(\boldsymbol{z}-\boldsymbol{z}^*).$$

It is easy to see that

$$V(\boldsymbol{z})\ge 0$$

and

$$\frac{\mathrm{d}\boldsymbol{V}}{\mathrm{d}t}=\left(\frac{\partial\boldsymbol{V}}{\partial\boldsymbol{z}}\right)^T\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}=\left(\nabla F(\boldsymbol{z})+\boldsymbol{D}^{-1}(\boldsymbol{z}-\boldsymbol{z}^*)\right)^T\dot{\boldsymbol{z}}\le-\dot{\boldsymbol{z}}\boldsymbol{D}^{-1}\dot{\boldsymbol{z}}\le 0.$$

This implies that the equilibrium point of (6) is stable in the sense of Lyapunov.□

Further, we prove the existence and convergence of the solution for the neural network (6).

Theorem 3. *For any initial point $\boldsymbol{z}(t_0)\in\Omega$, there exists a unique continuous solution $\boldsymbol{z}(t)\in\Omega$ with $[t_0,\infty)$.*

Proof. First, we prove $-\boldsymbol{z}+P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))$ is Lipschitz continuous.

Obviously, $\nabla F(\boldsymbol{z})=\boldsymbol{Bz}+\boldsymbol{c}$ is Lipschitz continuous. Let L be Lipschitz constant, $d_{min}=\min_i\{d_i\}$ and $d_{max}=\max_i\{d_i\}$. Thus, for any $\boldsymbol{z}_1,\boldsymbol{z}_2\in\Omega$, we have

$$\begin{aligned}&\|-\boldsymbol{z}_1+P_\Omega(\boldsymbol{z}_1-\boldsymbol{D}\nabla F(\boldsymbol{z}_1))+\boldsymbol{z}_2-P_\Omega(\boldsymbol{z}_2-\boldsymbol{D}\nabla F(\boldsymbol{z}_2))\|_2\\&\le\|\boldsymbol{z}_1-\boldsymbol{z}_2\|_2+\|P_\Omega(\boldsymbol{z}_1-\boldsymbol{D}\nabla F(\boldsymbol{z}_1))-P_\Omega(\boldsymbol{z}_2-\boldsymbol{D}\nabla F(\boldsymbol{z}_2))\|_2\\&\le\|\boldsymbol{z}_1-\boldsymbol{z}_2\|_2+\|\boldsymbol{z}_1-\boldsymbol{D}\nabla F(\boldsymbol{z}_1)-\boldsymbol{z}_2+\boldsymbol{D}\nabla F(\boldsymbol{z}_2))\|_2\\&\le(2+d_{max}L)\|\boldsymbol{z}_1-\boldsymbol{z}_2\|_2.\end{aligned}\tag{10}$$

It follows that the function $-\boldsymbol{z}+P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))$ is Lipschitz continuous on Ω. Then according to the local existence theorem of ordinary differential equation, there exists a unique continuous solution $\boldsymbol{z}(t)$ on $[t_0,T)$.

Next, we prove that Ω is an invariant set of (6), i.e., $\boldsymbol{z}(t)\in\Omega$.

Since $\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}=-\boldsymbol{z}+P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))$, we have

$$\int_{t_0}^t\left(\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}+\boldsymbol{z}\right)e^t\,\mathrm{d}t=\int_{t_0}^t P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))e^t\,\mathrm{d}t,$$

and thus

$$\boldsymbol{z}(t)=e^{-(t-t_0)}\boldsymbol{z}(t_0)+e^{-t}\int_{t_0}^t P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))e^t\,\mathrm{d}t.$$

By the integral mean value theorem, it follows that

$$\boldsymbol{z}(t)=e^{t_0-t}\boldsymbol{z}(t_0)+(1-e^{t_0-t})P_\Omega(\boldsymbol{z}(\bar{t})-\boldsymbol{D}\nabla F(\boldsymbol{z}(\bar{t}))),\quad\bar{t}\in(t_0,t).$$

Since $\boldsymbol{z}(t_0)\in\Omega$, $P_\Omega(\boldsymbol{z}(\bar{t})-\boldsymbol{D}\nabla F(\boldsymbol{z}(\bar{t})))\in\Omega$ and $e^{t_0-t}\in(0,1)$, then by the convexity of Ω, we have $\boldsymbol{z}(t)\in\Omega$.

Finally, we prove that $\boldsymbol{z}(t)$ is bounded on $[t_0,T)$.

Let $\boldsymbol{z}^*$ be an equilibrium point of (3). In the view of the definition of $\boldsymbol{V}(z)$, we have

$$0\le\frac{1}{2}(\boldsymbol{z}(t)-\boldsymbol{z}^*)^T\boldsymbol{D}^{-1}(\boldsymbol{z}(t)-\boldsymbol{z}^*)\le V(\boldsymbol{z}(t))\le V(\boldsymbol{z}(t_0)).$$

and hence

$$0\le\|\boldsymbol{z}(t)-\boldsymbol{z}^*\|_{\boldsymbol{D}^{-1}}^2\le 2V(\boldsymbol{z}(t_0)),$$

where $\|\boldsymbol{x}\|_{\boldsymbol{D}^{-1}}=\sqrt{\boldsymbol{x}^T\boldsymbol{D}^{-1}x}$.

Note that

$$\frac{1}{d_{max}}\|\boldsymbol{z}(t)-\boldsymbol{z}^*\|_2^2\le\|\boldsymbol{z}(t)-\boldsymbol{z}^*\|_{\boldsymbol{D}^{-1}}^2\le\frac{1}{d_{\min}}\|\boldsymbol{z}(t)-\boldsymbol{z}^*\|_2^2,$$

we obtain

$$\|\boldsymbol{z}(t)\|_2\le\|\boldsymbol{z}^*\|_2+\sqrt{2d_{max}V(\boldsymbol{z}(t_0))},\quad\forall\,t\in[t_0,T).$$

It shows that the solution $\boldsymbol{z}(t)$ is bounded on $[t_0,T)$, i.e., $T=\infty$.□

Theorem 4. *For any initial point $\boldsymbol{z}(t_0)\in\Omega$, the solution of (6) converges to a point in Ω^e. In particular, the neural network (6) is asymptotically stable when Ω^e contains exactly one point.*

Proof. From the proof of Theorem 3, for any $\boldsymbol{z}(t_0)\in\Omega$, $\boldsymbol{z}(t)$ is bounded for all $t\ge t_0$. Thus, there exists a subsequence $\{t_k\}$ such that

$$\lim_{t_k\to\infty}\boldsymbol{z}(t_k)=\hat{\boldsymbol{z}},$$

where $\hat{\boldsymbol{z}}$ satisfies $\frac{\mathrm{d}V(\boldsymbol{z})}{\mathrm{d}t}=0$. This indicates that $\hat{\boldsymbol{z}}\in L^+$, where L^+ is the positive limit set of (6) [41]. Then, by the LaSalle invariant set theorem [41], it follows that $L^+\subseteq\Omega^e$. Thus, we have $\hat{\boldsymbol{z}}\in\Omega^e$.

Define function

$$\Phi(\boldsymbol{z})=F(\boldsymbol{z})-F(\hat{\boldsymbol{z}})+\frac{1}{2}(\boldsymbol{z}-\hat{\boldsymbol{z}})^T\boldsymbol{D}^{-1}(\boldsymbol{z}-\hat{\boldsymbol{z}}).$$

Then, by the continuity of $\Phi(\boldsymbol{z})$, we have

$$\lim_{t\to\infty}\Phi(\boldsymbol{z}(t))=\lim_{t_k\to\infty}\Phi(\boldsymbol{z}(t_k))=\Phi(\hat{\boldsymbol{z}})=0.$$

Note that $\frac{1}{2}(\boldsymbol{z}-\hat{\boldsymbol{z}})^T\boldsymbol{D}^{-1}(\boldsymbol{z}-\hat{\boldsymbol{z}})\le\Phi(\boldsymbol{z})$. It yields $\lim_{t\to\infty}\boldsymbol{z}(t)=\hat{\boldsymbol{z}}$. So the solution trajectory of (6) is global to an equilibrium point $\hat{\boldsymbol{z}}$.

In particular, if Ω^e contains exactly one point $\hat{\boldsymbol{z}}$, the solution of the neural network (6) approaches to $\hat{\boldsymbol{z}}$ for any $\boldsymbol{z}(t_0)\in\Omega$, which shows that the neural network is asymptotically stable.□

Remark. If we choose $\boldsymbol{D}=\alpha\boldsymbol{I}\,(\alpha>0)$, where $\boldsymbol{I}$ is the identity matrix, the neural network is established by using the gradient projection. Further, we can choose the scaled gradient direction to be an approximation to Newton direction.

3. Simulation examples

For a given $\boldsymbol{z}=(\boldsymbol{u}^T,\boldsymbol{v}^T)^T$, we have

$$\boldsymbol{Bz}=\boldsymbol{B}\begin{pmatrix}\boldsymbol{u}\\\boldsymbol{v}\end{pmatrix}=\begin{pmatrix}\boldsymbol{A}^T\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{v})\\-\boldsymbol{A}^T\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{v})\end{pmatrix}.$$

It is clear that $\boldsymbol{Bz}$ can be obtained by computing the difference $\boldsymbol{u}-\boldsymbol{v}$ and then multiplying its result by $\boldsymbol{A}$ and $\boldsymbol{A}^T$ sequently for one time. Since $\nabla F(\boldsymbol{z})=\boldsymbol{Bz}+\boldsymbol{c}$, we conclude that the computation of $\nabla F(\boldsymbol{z})$ requires one multiplication each by $\boldsymbol{A}$ and $\boldsymbol{A}^T$. Furthermore, by splitting $\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}=-\boldsymbol{z}+P_\Omega(\boldsymbol{z}-\boldsymbol{D}\nabla F(\boldsymbol{z}))$ into the following two parts

$$\frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}=\begin{pmatrix}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t}\\\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t}\end{pmatrix}=\begin{pmatrix}-\boldsymbol{u}+P_\Omega(\boldsymbol{u}-\boldsymbol{D}_1(\boldsymbol{A}^T\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{v})-\boldsymbol{A}^T\boldsymbol{b}+\tau\boldsymbol{1}_n))\\-\boldsymbol{v}+P_\Omega(\boldsymbol{v}-\boldsymbol{D}_2(-\boldsymbol{A}^T\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{v})+\boldsymbol{A}^T\boldsymbol{b}+\tau\boldsymbol{1}_n))\end{pmatrix},$$

where $\boldsymbol{D}=\begin{pmatrix}\boldsymbol{D}_1&\\&\boldsymbol{D}_2\end{pmatrix}$, we can improve the efficiency of computations.

In this paper, we set $\boldsymbol{z}(t_0)=0$ and take the following d_i to be an approximation of the inverted second partial derivative,

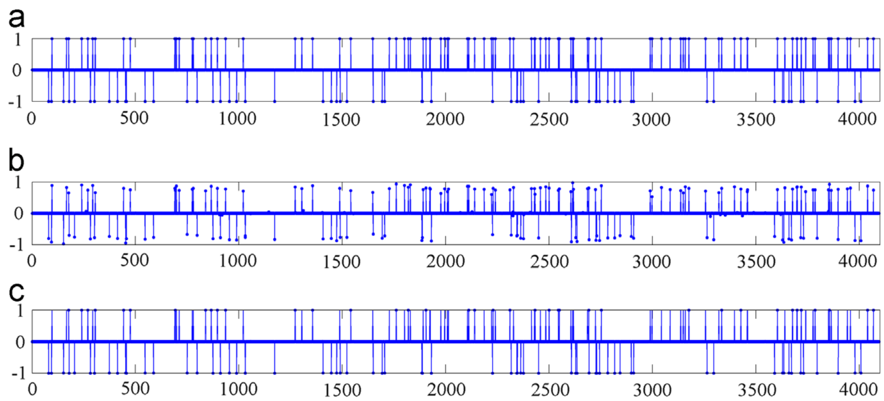

Fig. 5. (a) Original signal, (b) reconstruction by (6), (c) reconstruction after debiasing.

$$d_i = \begin{cases} \left(\frac{\partial^2 F(\boldsymbol{z})}{(\partial z_i)^2}\right)^{-1}, & \frac{\partial^2 F(\boldsymbol{z})}{(\partial z_i)^2} > \epsilon, \\ & , \qquad i = 1, \ldots, 2n, \\ 1 & , \frac{\partial^2 F(\boldsymbol{z})}{(\partial z_i)^2} \le \epsilon, \end{cases} \tag{11}$$

where $\epsilon = 10^{-6}$. This results in a diagonal approximation of Newton direction. Since $\nabla^2 F(\boldsymbol{z}) = \boldsymbol{B}$, it easily follows that $\boldsymbol{D}_1 = \boldsymbol{D}_2$ and $\frac{\partial^2 F(\boldsymbol{z})}{(\partial z_i)^2} = \frac{\partial^2 F(\boldsymbol{z})}{(\partial z_{i+n})^2} = a_{\cdot i}^T a_{\cdot i}$, where $a_{\cdot i}$ denotes the ith column of $\boldsymbol{A}$.

In the following, two simulation examples give an insight into the behavior of the proposed neural network. The implementation is in Matlab 2010b(7.11) code running on Windows 7 on an Intel Core i3-2310 Quad 2.10 GHz PC with 2.92 GB of RAM and about 16 digits of precision.

Example 1. Consider the following convex problem:

$$\min_{\boldsymbol{x}} \frac{1}{2} \| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} \|_2^2 + \tau \| \boldsymbol{x} \|_1,$$

where

$$\tau = 5, \quad \boldsymbol{y} = (2, 4, 1, 7)^T, \quad \boldsymbol{A} = \begin{pmatrix} 3 & 5 & 8 & 4 & 1 & 5 \\ 2 & 9 & 6 & 5 & 7 & 4 \\ 3 & 4 & 7 & 2 & 1 & 6 \\ 8 & 9 & 6 & 5 & 7 & 4 \end{pmatrix}.$$

This problem has a unique optimal solution $\boldsymbol{x}^* = (0.3461, 0.0852, 0, 0, 0.3719, 0)^T$. In Fig. 1, we plot the solution of the neural network (6) for the problem. In addition, we also display the transient behavior of the objective function value in Fig. 2. Fig. 1 shows that the solution of the neural network (6) is asymptotically stable and converges to the unique optimal solution $\boldsymbol{x}^*$.

Example 2. We consider the similar CS scenario in [23] and reconstruct a length-n sparse signal by k observations, where $k = 1024$ and $n = 4096$. The original signal contains 160 randomly generated ± 1 and the positions of these ± 1 are randomly selected. Furthermore, the observation $\boldsymbol{b}$ is obtained according to $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n}_e$, where $\boldsymbol{n}_e$ is white Gaussian noise of variance $\sigma^2 = 10^{-4}$ and the matrix $\boldsymbol{A}$ is obtained by first independently sampling its entries from the standard normal distribution and then orthonormalizing its rows. We choose $\tau = 0.1 \| \boldsymbol{A}^T \boldsymbol{y} \|_\infty$ as in [23]. Figs. 3 and 4 display the transient behavior of the neural network (6). The original signal and the estimate obtained by solving (4) using the scaled gradient projection neural network (6) are shown in Fig. 5(a) and (b). In addition, since the solution obtained by solving (4) is an approximation of the original signal, a debiasing step should be performed as suggested in [23].Therefore, we plot the debiased result in Fig. 5(c), which shows that the original signal is exactly recovered. This means that our neural network (6) is feasible and efficient for compressed sensing.

4. Conclusions

In this paper, a new neural network is established for solving $\ell_1 - \ell_2$ minimization problem based on scaled gradient projection. It is shown that the proposed neural network (6) globally converges to the optimal solution of (4). Simulation examples are given to demonstrate the efficiency of (6). The proposed neural network includes the gradient projection neural network. Furthermore, it can be applied to solve general convex programming with the positive orthant constraint.

References

[1] B.K. Natarajan, Sparse approximation solutions to linear systems, SIAM J. Sci. Comput. 24 (2) (1995) 227–234.
[2] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, SIAM J. Sci. Comput. 20 (1998) 33–61.
[3] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (2006) 1289–1306.
[4] D.L. Donoho, For most large underdetermined systems of linear equations, the minimal ℓ_1-norm solution is also the sparsest solution, Commun. Pure Appl. Math. 59 (2006) 907–934.
[5] E.J. Candès, Compressive sampling, in: Proceedings of International Congress of mathematicians, European Mathematical Society Publishing House, Zürich, Switzerland, 2006, pp. 1433–1452.
[6] E.J. Candès, T. Tao, Near optional signal recovery from random projections: universal encoding strategies, IEEE Trans. Inf. Theory 52 (2006) 5406–5425.
[7] E.J. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete Fourier information, IEEE Trans. Inf. Theory 52 (2006) 489–509.
[8] E.J. Candès, J. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements, Commun. Pure Appl. Math. 59 (2006) 1207–1223.
[9] H. Jung, J. Ye, E. Kim, Improved k-t blask and k-t sense using focuss, Phys. Med. Biol. 52 (11) (2007) 3201–3226.
[10] M. Lustig, D. Donoho, J. Pauly, Application of compressed sensing for rapid MR imaging, Magn. Reson. Med. 58 (2007) 1182–1195.
[11] A. Yang, A. Ganesh, S. Sastry, Y. Ma, Fast L1-minimization algorithms and an application in robust face recognition: a review, in: Proceedings of the 17th IEEE International Conference on Image Processing (ICIP), 2010, pp. 1849–1852.
[12] A.M. Andrés, S. Padovani, M. Tepper, et al., Face recognition on partially occluded images using compressed sensing, Pattern Recognit. Lett. 36 (2014) 235–242.
[13] A.M. Bruckstein, D.L. Donoho, M. Elad, From sparse solutions of systems of equations to sparse modeling of signals and images, SIAM Rev. 51 (2009) 34–81.
[14] S. Mallat, A Wavelet Tour of Signal Processing—The Sparse Way, third ed., Academic, Burlington, MA, 2009.
[15] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, J. Am. Stat. Assoc. 96 (2001) 1348–1360.
[16] R. Tibshirani, Regression shrinkage and selection via the LASSO, J. R. Stat. Soc.: Ser. B 58 (1996) 267–288.

[17] S.J. Kim, K. Koh, M. Lustig, S. Boyd, D. Gorinevsky, An interior-point method for large-scale l1-regularized least squares, IEEE Sel. Top. Signal Process. 4 (2007) 606–617.
[18] B. Turlach, W.N. Venables, S.J. Wright, Simultaneous variable selection, Technometrics 27 (2005) 349–363.
[19] I. Daubechies, M. Defrise, C. Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, Commun. Pure Appl. Math. 57 (2004) 1413–1457.
[20] A. Beck, M. Teboulle, A. Fast, Iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (1) (2009) 183–202.
[21] G. Davis, S. Mallat, M. Avellaneda, Greedy adaptive approximation, Constr. Approx. 13 (1) (1997) 57–98.
[22] J. Tropp, A. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, IEEE Trans. Inf. Theory 53 (2007) 4655–4666.
[23] M.A.T. Figueiredo, R.D. Nowak, S.J. Wright, Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems, IEEE J. Sel. Top. Signal Process. 1 (2007) 586–597.
[24] E.T. Hale, W. Yin, Y. Zhang, Fixed-point continuation for ℓ_1-minimization: methodology and convergence, SIAM J. Optim. 19 (3) (2008) 1107–1130.
[25] J.F. Cai, S. Osher, Z. Shen, Linearized Bregman iterations for compressed sensing, Math. Comput. 78 (2009) 1515–1536.
[26] S. Osher, Y. Mao, B. Dong, W. Yin, Fast linearized Bregman iteration for compressive sensing and sparse denoising, Commun. Math. Sci. 8 (1) (2010) 93–111.
[27] J.F. Yang, Y. Zhang, Alternating direction algorithms for L1-problems in compressive sensing, SIAM J. Sci. Comput. 33 (1) (2011) 250–278.
[28] L.O. Chua, G.N. Lin, Nonlinear programming without computation, IEEE Trans. Circuits Syst. 31 (1984) 182–188.
[29] D.W. Tank, J.J. Hopfield, Simple neural optimization network: an A/D converter, signal decision circuit and a linear programming circuit, IEEE Trans. Circuits Syst. 33 (1986) 533–541.
[30] M.P. Kennedy, L.O. Chua, Neural networks for nonlinear programming, IEEE Trans. Circuits Syst. 35 (1988) 554–562.
[31] S. Zhang, A.G. Constantinides, Lagrange programming neural networks, IEEE Trans. Circuits Syst. 39 (1992) 441–452.
[32] S. Effati, M. Baymani, A new nonlinear neural network for solving convex nonlinear programming problems, Appl. Math. Comput. 168 (2005) 1370–1379.
[33] Y. Yang, J. Cao, A feedback neural network for solving convex constraint optimization problems, Appl. Math. Comput. 201 (2008) 340–350.
[34] Y. Zhang, J. Wang, A dual neural network for convex quadratic programming subject to linear equality and inequality constraints, Phys. Lett. 298 (2002) 271–278.
[35] Y. Xia, G. Feng, J. Wang, A recurrent neural networks with exponential convergence for solving convex quadratic program and related linear piecewise equations, Neural Netw. 17 (2004) 1003–1015.
[36] X. Liang, A recurrent neural networks for nonlinear continuously differentiable optimization over a compact convex subset, IEEE Trans. Neural Netw. 12 (2001) 1487–1490.
[37] X.B. Gao, A novel neural network for nonlinear convex programming, IEEE Trans. Neural Netw. 15 (2004) 613–621.
[38] Y. Yang, Y. Cao, A new neural network for solving nonlinear convex programs with linear constraints, Neurocomputing 74 (2011) 3079–3083.
[39] J.J. Fuchs, Multipath time-delay detection and estimation, IEEE Trans. Signal Process. 47 (1999) 237–243.
[40] D.P. Bertsekas, Nonlinear Programming, second ed., Athena Scientific, Nashua, 1999.
[41] H.K. Khalil, Nonlinear Systems, third ed., Prentice Hall, Upper Saddle River, NJ, 2002.

Yongwei Liu received the B.S. degree in Statistics from Hubei University of Science and Technology, Xianning, China, the M.S. degree in operations research from Hainan Normal University, Haikou, China in 2011 and 2014, respectively. Now, he works at School of Mathematics and Statistics, Hainan Normal University, Haikou, China. His current research interests include compressed sensing and operations research.

Jianfeng Hu received the B.S. degree in applied mathematics, the M.S. degree in operations research and the Ph.D. degree in applied mathematics from Southeast University, Nanjing, China in 2001, 2004, and 2008, respectively. From January 2008 to January 2011, he worked at the Department of Mathematics, Southeast University, as a Lecturer. He is currently an assistant professor at School of Mathematics and Statistics, Hainan Normal University, Haikou, China. His research interests include neural networks, operations research, and numerical optimization and its application.