Modern Applied Science
www.ccsenet.org/journal.html
Vol. 2, No. 2
March 2008

A Recurrent Neural Network for Solving Convex Quadratic Program

Caihong Shan (Corresponding author)
Department of Mathematics, Yanshan University
438, He Bei street, Qinghuangdao 066004, China
E-mail: caihongshan2006@126.com
Huaiqin Wu
Department of Mathematics, Yanshan University
438, He Bei street, Qinghuangdao 066004, China
E-mail: huaiqinwu@ysu.edu.cn

The research is supported by the National Natural Science Foundation of China (10571035) and the Educational Science Foundation of Hebei province (Z 2007431)

Abstract

In this paper, we present a recurrent neural network for solving convex quadratic programming problems, in the theoretical aspect, we prove that the proposed neural network can converge globally to the solution set of the problem when the matrix involved in the problem is positive semi-definite and can converge exponentially to a unique solution when the matrix is positive definite. Illustrative examples further show the good performance of the proposed neural network.

Keywords: Recurrent neural network, Convex quadratic program, Convergence

1. Introduction and model formulation

In this paper, we are concerned with the following quadratic optimization program:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}x^T Ax + c^T x \\ &\text{subject to} \quad Dx \le \mathrm{b},\ \mathrm{x} \ge 0 \end{aligned} \tag{1}$$

and its dual

$$\begin{aligned} &\text{maximize} \quad -\mathrm{b}^T y - \frac{1}{2}x^T Ax \\ &\text{subject to} \quad -D^T y - Ax \le \mathrm{c},\ \mathrm{y} \ge 0 \end{aligned} \tag{2}$$

where $A \in R^{n\times n}$ is symmetric and positive semi-definite, $D \in R^{m\times n}$, $b \in R^m$, and $c \in R^n$. It is well known that quadratic optimization problems arise in a wide variety of scientific and engineering applications including regression analysis, image and signal progressing, parameter estimation, filter design, robot control, etc. In many real-time applications these optimization problems have a time-varying nature, they have to be solved in real time. The main advantage of neural network approach to optimization is that the nature of the dynamic solution procedure is inherently parallel and distributed. Therefore, the neural network approach can solve optimization problems in running time at the orders of magnitude much faster than the most popular optimization algorithms executed on general-purpose digital computers. At present, there are several neural network approaches for solving quadratic programming problem. Next, we describe the proposed neural network.

By the duality theorem of convex programming, (x^*, y^*) is an optimal solution to Eq.(1) and (2), respectively, if and only if (x^*, y^*) satisfies the Karush-Kuhn-Tucker conditions

$$\begin{aligned} &u = c + A^T x + D^T y \ge 0, \quad x \ge 0, \quad x^T u = 0 \\ &v = b - Dx \ge 0, \quad y \ge 0, \quad y^T v = 0 \end{aligned} \tag{3}$$

We see that the above Eq. (3) may be transformed into the linear projection equation of the following form

$$u = P_\Omega[u - Mu - q] \tag{4}$$

Where $\Omega = \{u = (x, y) \in R^{n+m} \mid u \geq 0\}$, $M = \begin{bmatrix} A & D^T \\ -D & 0 \end{bmatrix}$, $q = \begin{bmatrix} c \\ b \end{bmatrix}$, notice that the matrix M is positive semi-definite

because $u^T Mu = x^T Ax \geq 0$ .and P_Ω is a projection operator which is defined by $P_\Omega(u) = [P_\Omega(u_1), P_\Omega(u_2), ..., P_\Omega(u_{n+m})]^T$

and for $i = 1, 2, \ldots, n+m$, $P_\Omega(u_i) = \begin{cases} d_i & u_i < d_i \\ u_i & d_i \leq u_i \leq h_i \\ h_i & u_i > h_i \end{cases}$

In particular, if $h_i = +\infty$ and $d_i = 0$, then $P_\Omega(u_i) = (u_i)^+ = \max\{0, u_i\}$.

We can see that the optimal solutions of (1) and its dual (2) can be obtained by solving the project equation (4).

We propose a neural network for solving (1) and (4), whose dynamical equation is defined as follow:

$$\frac{du}{dt} = (I + M^T)(P_\Omega(u - Mu - q) - u) \tag{5}$$

Theorem 1. If $u^* = (x^*, y^*) \in R^{n+m}$, is an equilibrium point of the proposed neural network, then x^*, y^* is optimal solution to Eq.(1) and Eq.(2), respectively. On the other hand, if x^*, y^* is optimal solution to Eq.(1) and Eq.(2), then $((x^*)^T, (y^*)^T)^T$ is an equilibrium point of the proposed neural network.

2. Preliminaries

This section, we introduce the related definitions and lemmas for later discussion.

Definition 1. If $g : \Omega_1 \in R^l \to R$, then any nonempty set of the form

$L(r) = \{u \in \Omega_1 \mid g(u) \leq r\}, \quad r \in R,$

is said to be a level set of g.

Definition 2. A system is said to have globally exponential convergence rate with degree η at u^* if every trajectory staring at any initial point $u(t_0) \in R^l$ satisfies the condition

$\| u - u^* \| \leq c_0 \| u(t_0) - u^* \| \exp(-\eta(t - t_0)) \quad \forall t \geq t_0$

where c_0 and η are positive constants independent of the initial points.

Lemma 1 (Gronwall). Let u and v be real-valued non-negative continuous functions with domain $\{t \mid t \geq t_0\}$, let $a(t) = a_0(|t - t_0|)$ where a_0 is a monotone increasing function. If for $t \geq t_0$ $\quad u(t) \leq a(t) + \int_{t_0}^{t} u(s)v(s)ds$, then

$u(t) \leq a(t) \exp\left\{\int_{t_0}^{t} v(s)ds\right\}.$

Lemma 2: let Ω be a closed convex set. Then

$$(v - P_\Omega(v))^T (P_\Omega(v) - u) \geq 0 \qquad \forall v \in R^n \quad \forall u \in \Omega$$

and

$$\| P_\Omega(u) - P_\Omega(v) \| \leq \| u - v \| \qquad \forall u, v \in R^n$$

Lemma 3. let $g : \Omega_1 \in R^l \to R$, where Ω_1 is unbounded. Then for all level sets of g are bounded if and only if

$\lim_{k \to \infty} g(u^k) = +\infty$ whenever $u^k \subset D$ and $\lim_{k \to \infty} \| u^k \| = +\infty$.

With the lemma 1 and 2, we can give the existence and uniqueness of the solution to Eq. (5).

Theorem 2. For each $u_0 \in R^{n+m}$ there exists a unique continuous solution $u(t)$ for (5) with $u(t_0) = u_0$ over $[t_0, \infty)$.

Proof. Let $T(u) = (I + M^T)(P_\Omega(u - Mu - q) - u)$ .then $T(u)$ is Lipschitz continuous in R^{n+m} since for any

$u, v \in R^{n+m}$

$$\|T(u)-T(v)\| \le \|I+M^T\| (\|P_\Omega(u-Mu-q)-P_\Omega(v-Mv-q)\|+\|u-v\|)$$
$$\le \|I+M^T\| (2\|u-v\|+\|M\|\|u-v\|)$$
$$\le \|I+M^T\| (2+\|M\|)\|u-v\|$$

Thus for any $u_0 \in R^{n+m}$, there exists a unique and continuous solution $u(t)$ of Eq. (5), defined in $t_0 \le t < T$, with the initial condition $u(t_0)=u_0$. Let $[t_0, T)$ be its maximal interval of existence, we next show that $T=\infty$. From lemma 2, it follows

$$\|T(u)\| = \|I+M^T\|\|P_\Omega(u-Mu-q)-u\|$$
$$\le \|I+M^T\| (\|Mu+q\|+\|P_\Omega(u)-P_\Omega(u^*)\|+\|P_\Omega(u^*)-u\|)$$
$$\le \|I+M^T\| ((2+\|M\|)\|u\|+\|q\|+\|u^*\|+\|P(u^*)\|)$$

then

$$\|u(t)\| \le \|u_0\| + \int_{t_0}^{t}\|T(u(s))\|ds$$

$$\le (\|u_0\|+k_1(t-t_0))+k_2\int_{t_0}^{t}\|u(s)\|ds$$

where $k_1=\|I+M^T\|(\|q\|+\|u^*\|+\|P_\Omega(u^*)\|)$ and $k_2=\|I+M^T\|(2+\|M\|)$. Therefore, using Lemma 1 we have $\|u(t)\| \le (\|u_0\|+k_1(t-t_0))e^{k_2(t-t_0)}, \quad t\in[t_0,T)$.

Hence the solution $u(t)$ is bounded on $[t_0, T)$. So $T=\infty$.

3. Convergence result

In the present section, under the assumption that $\Omega^* \ne \Phi$. We prove the convergence of the proposed neural network.

Theorem 3. Let M is positive semi-definite, then the neural network (5) is stable in the sense of Lyapunov and globally convergences to the solution subset of the problem (4).

Proof. First, definite

$$F(u)=(I+M^T)(P_\Omega(u-Mu-q)-u).$$

Clearly, $F(u)=0$ if and only if u is a solution to problem (4). Thus the equilibrium point of the system in Eq.(5) correspond to solutions to problem (4) because $I+M^T$ is non-singular. Next, by theorem 2 we see that there exists a unique and continuous solution $u(t)$ with any initial point $u_0 \in R^{n+m}$ for system (5).

Now let u_0 be any initial point taken in Ω, and let $u(t)=u(t,t_0;u_0)$ be the solution of the initial value problem associated with (3). We then consider the following Lyapunov function

$$V(u)=\frac{1}{2}\|u-u^*\|_2^2, \quad u\in R^{n+m},$$

where $u^* \in \Omega^*$. Clearly, $\lim_{k\to\infty} V(u^k)=+\infty$ whenever the sequence $\{u^k\}\subset\Omega$ and $\lim_{k\to\infty}\|u^k\|=+\infty$. Thus by Lemma 3 we see that all the level sets of V are bounded. On the other hand, using the technique of the proof from the literature, by the properties of the projection operator we have for all $u\in R^{n+m}$ and all $y\in\Omega$

$$[y-P_\Omega(u-Mu-q)]^T[Mu+q-u+P_\Omega(u-Mu-q)] \ge 0.$$

Since u^* is a solution of the problem (4), for all $y\in\Omega$

$$\{y-u^*\}^T\{Mu^*+q\}\ge 0$$

Taking $y=u^*$ in the first inequality and taking $y=P_\Omega(u-Mu-q)$ in the second inequality and then adding the two resulting inequalities yields

$$\{u^*-P_\Omega(u-Mu-q)\}^T\{M(u-u^*)-u+P_\Omega(u-Mu-q)\}\ge 0$$

Then

$$(u^*-u)^T M(u-u^*)+(u-u^*)(I+M^T)(u-P_\Omega(u-Mu-q)) \ge \|u-P_\Omega(u-Mu-q)\|_2^2$$

Since is M positive semi-definite, it follows that

$$(u-u^*)^T(I+M^T)(u-P_\Omega(u-Mu-q)) \ge \|u-P_\Omega(u-Mu-q)\|_2^2$$

Therefore, we have

$$\frac{d}{dt}V(u)=\frac{dV}{du}\frac{du}{dt}$$
$$=(u-u^*)^T(I+M^T)(P_\Omega(u-Mu-q)-u)$$
$$\le -\|u-P_\Omega(u-Mu-q)\|_2^2\le 0$$

Thus $V(u)$ is a global Lyapunov function for the system in (5) and the system (5) is stable in the sense of Lyapunov. Since $\{u(t)\mid t\ge t_0\}\subset\Omega_0$ where $\Omega_0=\{u\in\Omega\mid V(u)\le V(u_0)\}$ and the function $V(u)$ is continuously differentiable on the bounded and closed set Ω_0, it follows from Lasalle's invariance principal that trajectories $u(t)$ will converge to E, the largest invariant subset of the following set:

$$E=\left\{u\in\Omega_0\mid\frac{dV}{dt}=0\right\}$$

It is easy to see that $du/dt=0$ if and only if $dV/dt=0$. It follows that

$$E=\left\{u\in\Omega_0\mid\frac{dV}{dt}=0\right\}=\Omega_0\cap\Omega^*$$

Which is a nonempty, convex, and invariant set containing in the solution set Ω^*. So

$$\lim_{t\to\infty}dis(u(t),E)=0\cdot$$

Therefore, the proposed neural network converges globally to the solution set of the problem (4).

Remark 1. If A is positive definite then M is positive definite, too. Thus, form the proof of theorem 3 we can get the neural network (5) is globally exponentially convergent.

Since

$$(u^*-u)^TM(u-u^*)+(u-u^*)(I+M^T)(u-P_\Omega(u-Mu-q))\ge\|u-P_\Omega(u-Mu-q)\|_2^2$$

We have

$$(u^*-u)^TM(u-u^*)+(u-u^*)(I+M^T)(u-P_\Omega(u-Mu-q))\ge 0$$

by Schwarz inequality we obtain

$$\|I+M^T\|\|u-P_\Omega(u-Mu-q)\|\ge\mu\|u-u^*\|$$

where $\mu=\lambda_{\min}(\frac{M^T+M}{2})$, thus

$$\|u-P_\Omega(u-Mu-q)\|\ge\frac{\mu}{\|I+M^T\|}\|u-u^*\|$$

So

$$\frac{d}{dt}V(u)\le-\|u-P_\Omega(u-Mu-q)\|^2$$
$$\le-\frac{\mu^2}{\|I+M^T\|^2}\|u-u^*\|^2$$
$$\le-\frac{2\mu^2}{\|I+M^T\|^2}V(u)$$

Thus

$$V(u)\le V(u_0)e^{-2\delta(t-t_0)}$$

where $\delta=\frac{\mu^2}{\|I+M^T\|^2}>0$, and hence

$$\|u(t)-u^*\|\le\|u_0-u^*\|e^{-\delta(t-t_0)}$$

Therefore, the proposed neural network is globally exponentially converges to the solution subset of the problem (4) if M is positive definite.

4. Simulation example

In order to demonstrate the effectiveness and efficiency of the proposed neural network, in this section, we discuss the simulation results through an example. The simulation is conduct on Matlab, the ordinary differential equation solver engaged is ode45s.

Example 1. Consider the convex quadratic program

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2} x^T Ax + c^T x \\ &\text{subject to} \quad Dx \le \text{b}, \text{x} \ge 0 \end{aligned}$$

and its dual

$$\begin{aligned} &\text{maximize} \quad -\text{b}^T y - \frac{1}{2} x^T Ax \\ &\text{subject to} \quad -D^T y - Ax \le \text{c}, \text{y} \ge 0 \end{aligned}$$

where

$$D = \begin{bmatrix} 5/12 & -1 \\ 5/2 & 1 \\ -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 35/12 \\ 35/2 \\ 5 \\ 5 \end{bmatrix}, \quad c = \begin{bmatrix} -30 \\ -30 \end{bmatrix}.$$

Its exact solution is $(5, 5)^T$. we use the system (5) to solve the above problem . All simulation result show that the solution trajectory always converges to the unique point $u^* = (5.000, 5.000, 0, 6.000, 0, 9.000)^T$ which corresponds to the optimal solution $(5, 5)^T$ and its dual solution $(0, 6, 0, 9)^T$. Let the starting point be $(2, 4, 0, 0, 0, 0)^T$ and $(5, 6, 1, 0, 3, 0)^T$ respectively. Figure 1 (a) and (b) show the transient behavior of the neural network for those starting point, respectively.

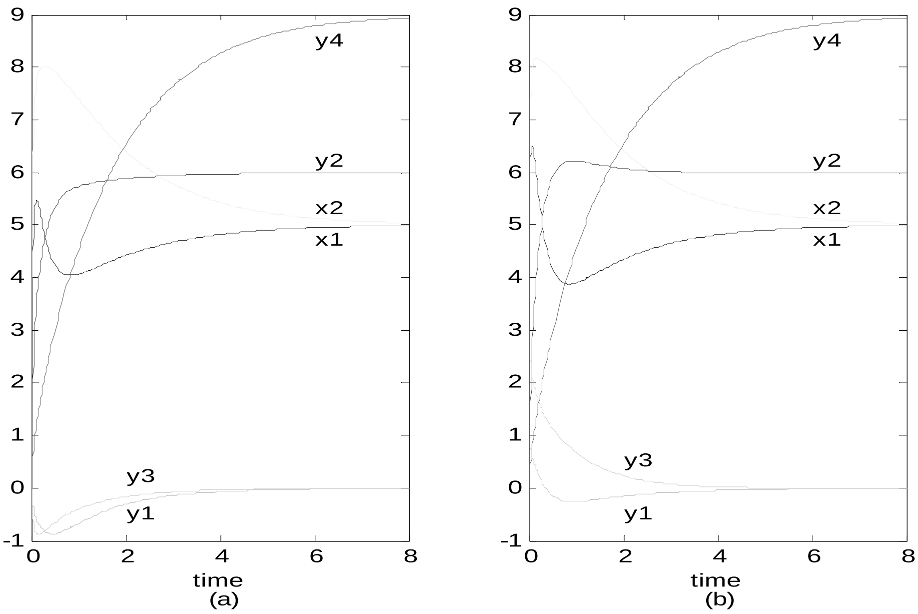

Figure 1. Transient behavior of the neural network (5) in example 1

(a) the initial point $(2, 4, 0, 0, 0, 0)^T$; (b) the initial point $(5, 6, 1, 0, 3, 0)^T$

5. Conclusion

In this paper, we have presented a recurrent neural network for solving convex quadratic programming problems, in the theoretical aspect, we have proved that the proposed neural network can converge globally to the solution set of the problem when the matrix involved in the problem is positive semi-definite and can converge exponentially to a unique solution when the matrix is positive definite. Illustrative examples further show the good performance of the proposed neural network.

References

Tao, Q., Cao, J. D., & Sun, D. M. (2001). A simple and high performance neural network for quadratic programming problems. *Appl. Math. Comput*. 124, 251-260.

Xia, Y. S. & Wang, G. (2005). An improved network for convex quadratic optimization with application to real-time beamforming. *Neurocomputing*, 64, 359-374

Xia, Y. S. & Wang, J. (2000). A recurrent neural network for solving linear projection equations. *Neural Networks*. 13, 337-350.

Xia, Y. S., Feng, G., & Wang, J. (2004). A recurrent neural network with exponential convergence for solving convex quadratic program and related linear piecewise equation. *Neural Networks*. 17, 1003-1015.

Xia, Y.S. (1996). A new neural network for solving linear and quadratic programming problems. *IEEE Transactions on Neural Networks*. 7,1544-1547.

Xue, X. P., & B, W. (2007). A project neural network for solving degenerate convex quadratic program. *Neurocomputing*. 70, 2449-2459.