Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2010, Article ID 403749, 13 pages
doi:10.1155/2010/403749

Research Article

A Nonlinear Projection Neural Network for Solving Interval Quadratic Programming Problems and Its Stability Analysis

Huaiqin Wu, Rui Shi, Leijie Qin, Feng Tao, and Lijun He

College of Science, Yanshan University, Qinhuangdao 066001, China

Correspondence should be addressed to Huaiqin Wu, huaiqinwu@ysu.edu.cn

Received 26 February 2010; Accepted 19 July 2010

Academic Editor: Jitao Sun

This paper presents a nonlinear projection neural network for solving interval quadratic programs subject to box-set constraints in engineering applications. Based on the Saddle point theorem, the equilibrium point of the proposed neural network is proved to be equivalent to the optimal solution of the interval quadratic optimization problems. By employing Lyapunov function approach, the global exponential stability of the proposed neural network is analyzed. Two illustrative examples are provided to show the feasibility and the efficiency of the proposed method in this paper.

1. Introduction

In lots of engineering applications including regression analysis, image and signal progressing, parameter estimation, filter design and robust control, and so forth [1], it is necessary to solve the following quadratic programming problem:

$$\begin{aligned} \min \quad & \frac{1}{2}x^TQx + c^Tx, \\ \text{subject to} \quad & x \in \Omega, \end{aligned} \tag{1.1}$$

where $Q \in R^{n\times n}$, $c \in R^n$, and Ω is a convex set. When Q is a positive definite matrix, the problem (1.1) is said to be the convex quadratic program. When Q is a semipositive definite matrix, the problem (1.1) is said to be the degenerate convex quadratic program. In general, the matrix Q is not precisely known, but can only be enclosed in intervals, that is, $\underline{Q} \le Q \le \overline{Q}$. Such quadratic program with interval data is named as interval quadratic program usually. In the recent years, there have been some project neural network approaches for solving the problem (1.1); see, for example, [2–15], and the references therein. In [2], Kennedy and Chua presented a primal network for solving the convex quadratic program. This network contains a finite penalty parameter, so it converges an approximate solution only. To overcome the penalty parameter, in [3, 4], Xia proposed several primal projection neural networks for solving the convex quadratic program and it dual, and analyzed the global asymptotic stability of the proposed neural networks when the constraint set Ω is a box set. In [5, 6], Xia et al. presented a recurrent projection neural network for solving the convex quadratic program and related linear piecewise equation, and gave some conditions of the exponential convergence. In [7, 8], Yang and Cao presented a delayed projection neural network for solving problem (1.1), and analyzed the global asymptotic stability and exponential stability of the proposed neural networks when the constraint set Ω is a unbounded box set. In order to solve the degenerate convex quadratic program, Tao et al. [9] and Xue and Bian [10, 11] proposed two projection neural networks, and proved that the equilibrium point of the proposed neural networks was equivalent to the KT point of the quadratic programming problem. Particularly, in [10], the proposed neural network was shown to have complete convergence and finite-time convergence, and the nonsingular part of the output trajectory with respect to Q has an exponentially convergent rate. In [12, 13], Hu and Wang designed a general projection neural network for solving monotone linear variational inequalities and extended linear-quadratic programming problems, and proved that the proposed network was exponentially convergent when the constraint set Ω is a polyhedral set.

In order to solve the interval quadratic program, in [14], Ding and Huang presented a new class of interval projection neural networks, and proved the equilibrium point of this neural networks is equivalent to the KT point of a class of interval quadratic program. Furthermore, some sufficient conditions to ensure the existence and global exponential stability for the unique equilibrium point of interval projection neural networks are given. To the best of the authors knowledge, the work in [14] is first to study solving the interval quadratic program by a projection neural network. However, the interval quadratic program discussed in [14] is only a quadratic program without constraints, thus has many limitations in practice. It is well known that the quadratic program with constraints is more popular.

Motivated by the above discussion, in the present paper, a new projection neural network for solving the interval quadratic programming problem with box-set constraints is presented. Based on the Saddle theorem, the equilibrium point of the proposed neural network is proved to be equivalent to the KT point of the interval quadratic program. By using the fixed point theorem, the existence and uniqueness of an equilibrium point of the proposed neural network are analyzed. By constructing a suitable Lyapunov function, a sufficient condition to ensure the existence and global exponential stability for the unique equilibrium point of interval projection neural network is obtained.

This paper is organized as follows. Section 2 describes the system model and gives some necessary preliminaries; Section 3 gives the proof of the existence of equilibrium point of the proposed neural network, and discusses the global exponential stability of the proposed neural network; Section 4 provides two numerical examples to demonstrate the validity of the obtained results. Some conclusions are drawn in Section 5.

2. A Projection Neural Network Model

Consider the following interval quadratic programming problem:

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Q x + c^T x \\ \text{subject to} \quad & g \le Dx \le h, \\ & \underline{Q} \le Q \le \overline{Q}, \end{aligned} \tag{2.1}$$

where $Q = (q_{ij})_{n\times n}$, $\underline{Q} = (\underline{q_{ij}})_{n\times n}$, $\overline{Q} = (\overline{q_{ij}})_{n\times n} \in R^{n\times n}$; $c, g, h \in R^n$, and $D = \operatorname{diag}(d_1, d_2, \ldots, d_n)$ is a positive definite diagonal matrix. $\underline{Q} \le Q \le \overline{Q}$ means $\underline{q_{ij}} \le q_{ij} \le \overline{q_{ij}}$, $i, j = 1, \ldots, n$. The Lagrangian function of the problem (2.1) is

$$L(x, u, \eta) = \frac{1}{2}x^T Q x + c^T x - u^T(Dx - \eta), \quad \underline{Q} \le Q \le \overline{Q}, \tag{2.2}$$

where $u \in R^n$ is referred to as the Lagrange multiplier and $\eta \in X = \{u \in R^n \mid g \le u \le h\}$. Based on the well-known Saddle point theorem [1], x^* is an optimal solution of (2.1) if and only if there exist u^* and η^*, satisfying $L(x^*, u, \eta^*) \le L(x^*, u^*, \eta^*) \le L(x, u^*, \eta)$, that is,

$$\begin{aligned} & \frac{1}{2}x^{*T} Q x^* + c^T x^* - u^T(Dx^* - \eta^*) \\ & \quad \le \frac{1}{2}x^{*T} Q x^* + c^T x^* - u^{*T}(Dx^* - \eta^*) \\ & \quad \le \frac{1}{2}x^T Q x + c^T x - u^{*T}(Dx - \eta). \end{aligned} \tag{2.3}$$

By the first inequality in (2.3), $(u - u^*)^T(Dx^* - \eta^*) \ge 0$, for all $u \in R^n$, hence $Dx^* = \eta^*$. Let $f(x) = (1/2)x^T Q x + c^T x - u^{*T} Dx$. By the second inequality in (2.3), $f(x^*) - f(x) \le u^{*T}(\eta - \eta^*)$, for all $x \in R^n$, $\eta \in X$. If there exists $\overline{x} \in R^n$ such that $f(x^*) - f(\overline{x}) > 0$, then $0 < u^{*T}(\eta - \eta^*)$, for all $\eta \in X$, which is contradictive when $\eta = \eta^*$. Thus, for any $x \in R^n$, it follows that $f(x^*) - f(x) \le 0$ and $u^{*T}(\eta - \eta^*) \ge 0$, for all $\eta \in X$.

By using the project formulation [16], the above inequality can be equivalently represented as $\eta^* = P_X(\eta^* - u^*)$, where $P_X(u) = [P_X(u_1), P_X(u_2), \ldots, P_X(u_n)]^T$ is a project function, and, for $i = 1, 2, \ldots, n$,

$$P_X(u_i) = \begin{cases} g_i, & u_i < g_i, \\ u_i, & g_i \le u_i \le h_i, \\ h_i, & u_i > h_i. \end{cases} \tag{2.4}$$

On the other hand, $f(x^*) \le f(x)$, for all $x \in R^n$. This implies that

$$\nabla f(x^*) = Qx^* + c - Du^* = 0. \tag{2.5}$$

Thus, x^* is an optimal solution of (2.1) if and only if there exist u^* and η^*, such that (x^*, u^*, η^*) satisfies

$$\begin{aligned} Dx &= \eta, \\ Qx + c - Du &= 0, \\ \eta &= P_X(\eta - u). \end{aligned} \tag{2.6}$$

From (2.6), it follows that $Dx = P_X(Dx - u)$. Hence, x^* is an optimal solution of (2.1) if and only if there exists u^* such that (x^*, u^*) satisfies

$$\begin{aligned} Qx + c - Du &= 0, \\ Dx &= P_X(Dx - u). \end{aligned} \tag{2.7}$$

Substituting $u = D^{-1}(Qx + c)$ into the equation $Dx = P_X(Dx - u)$, we have

$$Dx = P_X\left(Dx - D^{-1}Qx - D^{-1}c\right), \tag{2.8}$$

where

$$D^{-1} = \begin{pmatrix} \frac{1}{d_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{d_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{d_n} \end{pmatrix}, \qquad D^{-1}Q = \begin{pmatrix} \frac{1}{d_1}q_{11} & \frac{1}{d_1}q_{12} & \cdots & \frac{1}{d_1}q_{1n} \\ \frac{1}{d_2}q_{21} & \frac{1}{d_2}q_{22} & \cdots & \frac{1}{d_2}q_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{d_n}q_{n1} & \frac{1}{d_n}q_{n2} & \cdots & \frac{1}{d_n}q_{nn} \end{pmatrix}. \tag{2.9}$$

By the above discussion, we can obtain the following proposition.

Proposition 2.1. *Let x^* be a solution of the project equation*

$$Dx = P_X\left(Dx - D^{-1}Qx - D^{-1}c\right), \tag{2.10}$$

then, x^* is an optimal solution of the problem (2.1).

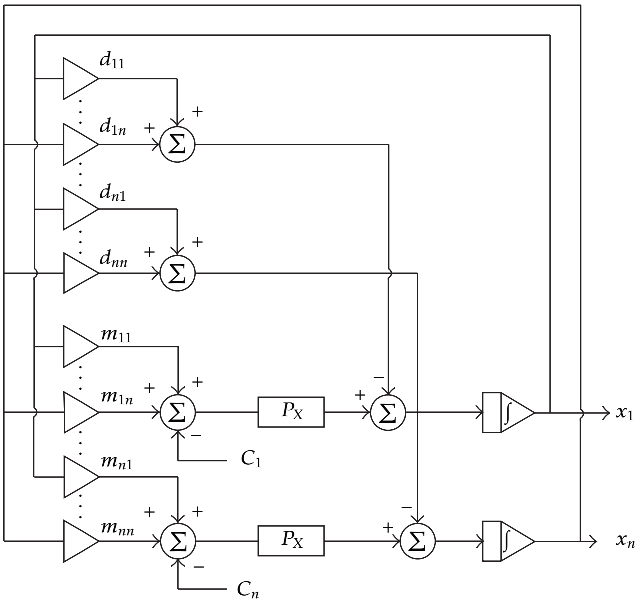

Figure 1: Architecture of the proposed neural network in (2.11).

In the following, we propose a neural network, which is said to be the interval projection neural network, for solving (2.1) and (2.10), whose dynamical equation is defined as follows:

$$\begin{gathered}\frac{dx(t)}{dt} = P_X\left(Dx - D^{-1}Qx - D^{-1}c\right) - Dx, \quad t > t_0,\\ x(t_0) = x_0,\\ \underline{Q} \le Q \le \overline{Q}.\end{gathered} \tag{2.11}$$

The neural networks (2.11) can be equivalently written as

$$\begin{gathered}\frac{dx_i(t)}{dt} = P_X\left(d_i x_i - \frac{1}{d_i}\sum_{j=1}^{n} q_{ij}x_j - \frac{c_i}{d_i}\right) - d_i x_i, \quad t > t_0,\\ x_i(t_0) = x_{i0},\\ \underline{q_{ij}} \le q_{ij} \le \overline{q_{ij}}, \quad i = 1,\dots,n, \ j = 1,\dots,n.\end{gathered} \tag{2.12}$$

Figure 1 shows the architecture of the neural network (2.11), where $M = (m_{ij})_{n\times n} = D - D^{-1}Q$, $C = D^{-1}c$, and $D = (d_{ij})_{n\times n}$.

Definition 2.2. The point x^* is said to be an equilibrium point of interval projection neural network (2.11), if x^* satisfies

$$0 = P_X\left(Dx^* - D^{-1}Qx^* - D^{-1}c\right) - Dx^*. \tag{2.13}$$

By Proposition 2.1 and Definition 2.2, we have the following theorem.

Theorem 2.3. *The point x^* is an equilibrium point of the interval projection neural network* (2.11) *if and only if it is an optimal solution of the interval quadratic program* (2.1)*.*

Definition 2.4. The equilibrium point x^* of the neural network (2.11) is said to be globally exponentially stable, if the trajectory $x(t)$ of the neural network (2.11) with the initial value x_0 satisfies

$$\|x(t) - x^*\| \leq c_0 \exp\left(-\beta(t - t_0)\right), \quad \forall t \geq t_0, \tag{2.14}$$

where $\beta > 0$ is a constant independent of the initial value x_0 and $c_0 > 0$ is a constant dependent on the initial value x_0. $\|\cdot\|$ denotes the 1-norm of R^n, that is, $\|x\| = \sum_{i=1}^{n}|x_i|$.

Lemma 2.5 (see [17]). *Let $\Omega \subset R^n$ be a closed convex set. Then,*

$$\begin{gathered}(v - P_\Omega(v))^T(P_\Omega(v) - u) \geq 0, \quad \forall u \in \Omega,\ v \in R^n, \\ \|P_\Omega(u) - P_\Omega(v)\| \leq \|u - v\|, \quad \forall u, v \in R^n,\end{gathered} \tag{2.15}$$

where $P_\Omega(u)$ is a project function on Ω, given by $P_\Omega(u) = \arg\min_{y\in\Omega}\|u - y\|$.

3. Stability Analysis

In order to obtain the results in this paper, we make the following assumption for the neural network (2.11):

$$H_1\colon q_{ii} \leq d_i^2, \qquad d_i^2 - \frac{d_i}{d^*} < \underline{q_{ii}} - \sum_{j=1, j\neq i}^{n} q_{ji}^* < d_i^2, \quad i = 1, 2, \ldots, n, \tag{3.1}$$

where $d^* = \max_{1\leq i\leq n} 1/d_i$, $q_{ji}^* = \max\{|\underline{q_{ji}}|, |\overline{q_{ji}}|\}$.

Theorem 3.1. *If the assumption H_1 is satisfied, then there exists a unique equilibrium point for the neural network* (2.11)*.*

Proof. Let $\mathcal{T}(x) = D^{-1}P_X(Dx - D^{-1}Qx - D^{-1}c),\ \ x \in R^n$. By Definition 2.2, it is obvious that the neural network (2.11) has a unique equilibrium point if and only if $\mathcal{T}$ has a unique fixed point in R^n. In the following, by using fixed point theorem, we prove that $\mathcal{T}$ has a unique fixed point in R^n. For any $x,\ \ y \in R^n$, by Lemma 2.5 and the assumption H_1, we can obtain that

$$\begin{aligned}
&\|\mathcal{T}(x) - \mathcal{T}(y)\| \\
&\quad = \left\| D^{-1}P_X\left(Dx - D^{-1}Qx - D^{-1}c\right) - D^{-1}P_X\left(Dy - D^{-1}Qy - D^{-1}c\right) \right\| \\
&\quad \leq \left\| D^{-1} \right\| \left\| P_X\left(Dx - D^{-1}Qx - D^{-1}c\right) - P_X\left(Dy - D^{-1}Qy - D^{-1}c\right) \right\| \\
&\quad \leq \left\| D^{-1} \right\| \left\| \left(Dx - D^{-1}Qx - D^{-1}c\right) - \left(Dy - D^{-1}Qy - D^{-1}c\right) \right\| \\
&\quad = \left\| D^{-1} \right\| \left\| \left(D - D^{-1}Q\right)(x - y) \right\| \\
&\quad \leq \left\| D^{-1} \right\| \left\| D - D^{-1}Q \right\| \|x - y\| \\
&\quad = \max_{1\leq i\leq n}\frac{1}{d_i} \cdot \max_{1\leq i\leq n}\left(d_i - \frac{1}{d_i}q_{ii} + \frac{1}{d_i}\sum_{j=1, j\neq i}^{n} |q_{ji}| \right) \cdot \|x - y\| \\
&\quad \leq d^* \max_{1\leq i\leq n}\left(d_i - \frac{1}{d_i}\underline{q_{ii}} + \frac{1}{d_i}\sum_{j=1, j\neq i}^{n} q_{ji}^* \right) \|x - y\| \\
&\quad = \max_{1\leq i\leq n} \ell_i \|x - y\|,
\end{aligned} \tag{3.2}$$

where $\ell_i = d^*(d_i - (1/d_i)\underline{q_{ii}} + (1/d_i)\sum_{j=1, j\neq i}^{n} q_{ji}^*)$. By the assumption H_1, $0 < d^*(d_i - (1/d_i)\underline{q_{ii}} + (1/d_i)\sum_{j=1, j\neq i}^{n} q_{ji}^*) < 1$, $i = 1, 2, \ldots, n$. This implies that $0 < \max_{1\leq i\leq n}\ell_i < 1$. Equation (3.2) shows that $\mathcal{T}$ is a contractive mapping, and hence $\mathcal{T}$ has a unique fixed point. This completes the proof. □

Proposition 3.2. *If the assumption H_1 holds, then for any $x_0 \in R^n$, there exists a solution with the initial value $x(0) = x_0$ for the neural network* (2.11).

Proof. Let $\mathcal{F} = D(\mathcal{T} - \mathcal{I})$, where $\mathcal{I}$ is an identity mapping, then $\mathcal{F}(x) = P_X(Dx - D^{-1}Qx - D^{-1}c) - Dx$. By (3.2), we have

$$\begin{aligned}
\|\mathcal{F}(x) - \mathcal{F}(y)\| &= \|D(\mathcal{T} - \mathcal{I})(x) - D(\mathcal{T} - \mathcal{I})(y)\| \\
&\leq \max_{1\leq i\leq n} d_i \left(1 + \max_{1\leq i\leq n} \ell_i\right) \|x - y\|, \quad \forall x, y \in R^n.
\end{aligned} \tag{3.3}$$

Equation (3.3) means that the mapping $\mathcal{F}$ is globally Lipschitz. Hence, for any $x_0 \in R^n$, there exists a solution with the initial value $x(0) = x_0$ for the neural network (2.11). This completes the proof. □

Proposition 3.2 shows the existence of the solution for the neural network (2.11).

Theorem 3.3. *If the assumption H_1 is satisfied, then the equilibrium point of the neural network* (2.11) *is globally exponentially stable.*

Proof. By Theorem 3.1, the neural network (2.11) has a unique equilibrium point. We denote the equilibrium point of the neural network (2.11) by x^*.

Consider Lyapunov function $\mathcal{V}(t) = \|x(t) - x^*\| = \sum_{i=1}^{n} |x_i(t) - x_i^*|$. Calculate the derivative of $\mathcal{V}(t)$ along the solution $x(t)$ of the neural network (2.11). When $t > t_0$, we have

$$\begin{aligned}
\frac{d\mathcal{V}(t)}{dt} &= \sum_{i=1}^{n} \frac{x_i(t) - x_i^*}{\left|x_i(t) - x_i^*\right|} \frac{d\left(x_i(t) - x_i^*\right)}{dt} \\
&= \sum_{i=1}^{n} \frac{x_i(t) - x_i^*}{\left|x_i(t) - x_i^*\right|} \left(P_X\left(d_i x_i - \frac{1}{d_i}\sum_{j=1}^{n} q_{ij} x_j - \frac{c_i}{d_i} \right) - d_i x_i \right) \\
&= \sum_{i=1}^{n} \frac{x_i(t) - x_i^*}{\left|x_i(t) - x_i^*\right|} \left(P_X\left(d_i x_i - \frac{1}{d_i}\sum_{j=1}^{n} q_{ij} x_j - \frac{c_i}{d_i} \right) - d_i x_i^* + d_i x_i^* - d_i x_i \right) \\
&= -\sum_{i=1}^{n} d_i \left|x_i(t) - x_i^*\right| + \sum_{i=1}^{n} \frac{x_i(t) - x_i^*}{\left|x_i(t) - x_i^*\right|} \left(P_X\left(d_i x_i - \frac{1}{d_i}\sum_{j=1}^{n} q_{ij} x_j - \frac{c_i}{d_i} \right) - d_i x_i^* \right) \\
&\le \left(-\min_{1\le i\le n} d_i \right) \sum_{i=1}^{n} \left|x_i(t) - x_i^*\right| + \sum_{i=1}^{n} \left| P_X\left(d_i x_i - \frac{1}{d_i}\sum_{j=1}^{n} q_{ij} x_j - \frac{c_i}{d_i} \right) - d_i x_i^* \right| \\
&= \left(-\min_{1\le i\le n} d_i \right) \|x - x^*\| + \left\| P_X\left(Dx - D^{-1}Qx - D^{-1}c \right) - Dx^* \right\|.
\end{aligned} \tag{3.4}$$

Noting $Dx^* = P_X(Dx^* - D^{-1}Qx^* - D^{-1}c)$, by Lemma 2.5, we have

$$\begin{aligned}
&\left\| P_X\left(Dx - D^{-1}Qx - D^{-1}c \right) - Dx^* \right\| \\
&\quad = \left\| P_X\left(Dx - D^{-1}Qx - D^{-1}c \right) - P_X\left(Dx^* - D^{-1}Qx^* - D^{-1}c \right) \right\| \\
&\quad \le \left\| \left(Dx - D^{-1}Qx - D^{-1}c \right) - \left(Dx^* - D^{-1}Qx^* - D^{-1}c \right) \right\| \\
&\quad \le \left\| \left(D - D^{-1}Q \right)(x - x^*) \right\| \\
&\quad \le \left\| D - D^{-1}Q \right\| \|x - x^*\|.
\end{aligned} \tag{3.5}$$

Hence,

$$\begin{aligned}
\frac{d\mathcal{V}(t)}{dt} &\leq \left(-\min_{1\leq i\leq n} d_i + \max_{1\leq i\leq n}\left(d_i - \frac{1}{d_i}\left|q_{ii}\right| + \frac{1}{d_i}\sum_{j=1,j\neq i}^{n}\left|q_{ji}\right|\right)\right)\cdot\|x-x^*\| \\
&\leq \left(-\frac{1}{d^*} + \max_{1\leq i\leq n}\left(d_i - \frac{1}{d_i}\underline{q_{ii}} + \frac{1}{d_i}\sum_{j=1,j\neq i}^{n} q_{ji}^*\right)\right)\cdot\|x-x^*\| \\
&= \max_{1\leq i\leq n}\ell_i'\|x-x^*\|,
\end{aligned}\tag{3.6}$$

where $\ell_i' = d_i - (1/d_i)\underline{q_{ii}} + (1/d_i)\sum_{j=1,j\neq i}^{n} q_{ji}^* - (1/d^*)$. By the assumption H_1, $\ell_i' < 0$. Hence, $\max_{1\leq i\leq n}\ell_i' < 0$. Let $\ell^* = \min_{1\leq i\leq n}|\ell_i'|$, then $\ell^* > 0$. Equation (3.6) can be rewritten as $d\mathcal{V}(t)/dt \leq -\ell^*\|x-x^*\|$. It follows easily that $\|x(t)-x^*\| \leq \|x_0-x^*\|\exp(-\ell^*(t-t_0))$, for all $t > t_0$. This shows that the equilibrium point x^* of the neural network (2.11) is globally exponentially stable. This completes the proof. □

4. Illustrative Examples

Example 4.1. Consider the interval quadratic program defined by $D = \mathrm{diag}(2,1)$, $g = (2,2)^T$, $h = (3,2)^T$, $c^T = (-1,-1)$, $\underline{Q} = \left(\begin{smallmatrix}3 & 0.1\\ 0.1 & 0.7\end{smallmatrix}\right)$, $\overline{Q} = \left(\begin{smallmatrix}3.1 & 0.2\\ 0.2 & 0.8\end{smallmatrix}\right)$, and $Q^* = (q_{ji}^*) = \left(\begin{smallmatrix}3.1 & 0.2\\ 0.2 & 0.8\end{smallmatrix}\right)$.

The optimal solution of this quadratic program is $(1,2)$ under $Q = \underline{Q}$ or $Q = \overline{Q}$. It is easy to check that

$$\begin{gathered}
3.0 < q_{11} < 3.1, \qquad q_{11} < d_1^2 = 4, \\
0.7 < q_{22} < 0.8, \qquad q_{22} < d_2^2 = 1, \\
d_1^2 - \frac{d_1}{d^*} = 2 < \underline{q_{11}} - q_{21}^* = 3.0 - 0.2 < d_1^2 = 4, \\
d_2^2 - \frac{d_2}{d^*} = 0 < \underline{q_{22}} - q_{12}^* = 0.7 - 0.2 < d_2^2 = 1.
\end{gathered}\tag{4.1}$$

The assumption H_1 holds. By Theorems 3.1 and 3.3, the neural network (2.11) has a unique equilibrium point which is globally exponentially stable, and the unique equilibrium point $(1,2)$ is the optimal solution of this quadratic programming problem.

In the case of $Q = \underline{Q}$, Figure 2 reveals that the projection neural network (2.11) with random initial value $(2.5,-0.5)$ has a unique equilibrium point $(1,2)$ which is globally exponentially stable. In the case of $Q = \overline{Q}$, Figure 3 reveals that the projection neural network (2.11) with random initial value $(-2.5,3)$ has the same unique equilibrium point $(1,2)$ which is globally exponentially stable. These are in accordance with the conclusion of Theorems 3.1 and 3.3.

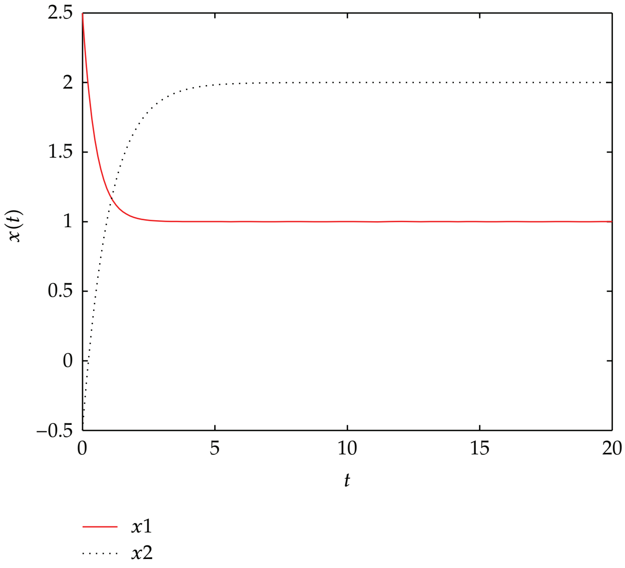

Figure 2: Convergence of the state trajectory of the neural network with random initial value (2.5,−0.5); $Q = \underline{Q}$ in this example.

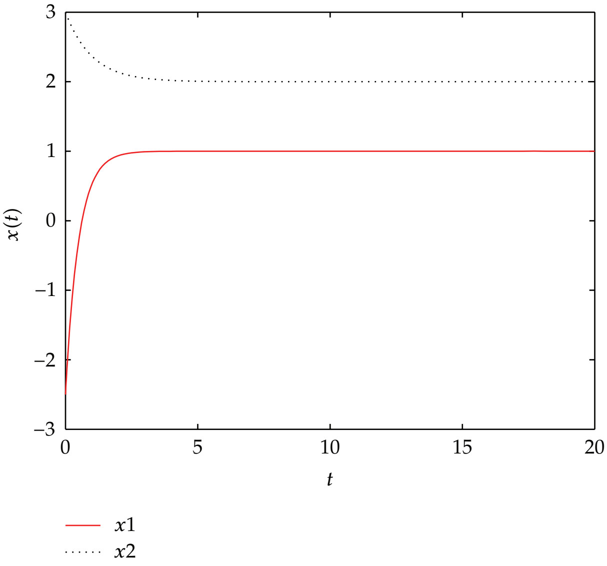

Figure 3: Convergence of the state trajectory of the neural network with random initial value (−2.5,3); $Q = \overline{Q}$ in this example.

Example 4.2. Consider the interval quadratic program defined by $D = \operatorname{diag}(1, 2, 2)$, $g = (1, 2, 3)^T$, $h = (2, 3, 4)^T$, $c^T = (1, 1, 1)$, $\underline{Q} = \begin{pmatrix} 0.8 & 0.2 & 0.3 \\ 0.2 & 3 & 0.1 \\ 0.3 & 0.1 & 3.5 \end{pmatrix}$, $\overline{Q} = \begin{pmatrix} 0.9 & 0.3 & 0.4 \\ 0.3 & 3.1 & 0.2 \\ 0.4 & 0.2 & 3.6 \end{pmatrix}$, and $Q^* = (q^*_{ji}) = \begin{pmatrix} 0.9 & 0.3 & 0.4 \\ 0.3 & 3.1 & 0.2 \\ 0.4 & 0.2 & 3.6 \end{pmatrix}$.

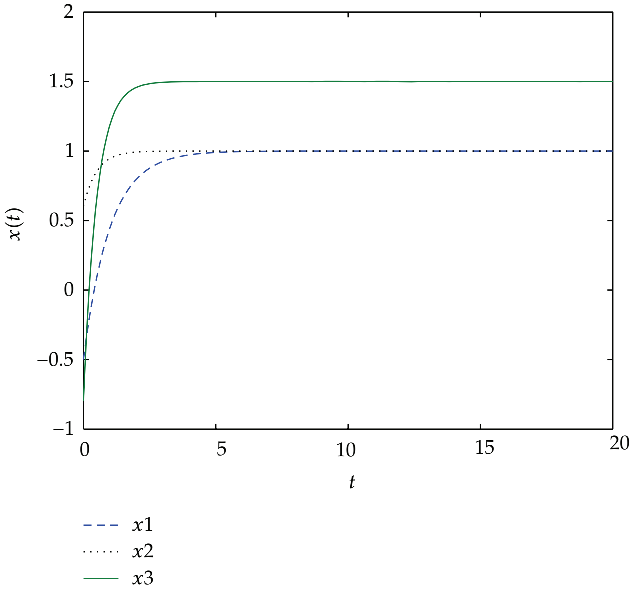

Figure 4: Convergence of the state trajectory of the neural network with random initial value (−0.5,0.6,−0.8); $Q = \underline{Q}$ in this example.

The optimal solution of this quadratic program is $(1, 1, 1.5)$ under $Q = \underline{Q}$ or $Q = \overline{Q}$. It is easy to check that

$$\begin{gathered} 0.8 < q_{11} < 0.9, \quad q_{11} < d_1^2 = 1, \\ 3.0 < q_{22} < 3.1, \quad q_{22} < d_2^2 = 4, \\ 3.5 < q_{33} < 3.6, \quad q_{33} < d_3^2 = 4, \\ d_1^2 - \frac{d_1}{d^*} = 0 < \underline{q_{11}} - \left(q_{21}^* + q_{31}^*\right) = 0.8 - (0.3 + 0.4) < d_1^2 = 1, \\ d_2^2 - \frac{d_2}{d^*} = 2 < \underline{q_{22}} - \left(q_{12}^* + q_{32}^*\right) = 3.0 - (0.3 + 0.2) < d_2^2 = 4, \\ d_3^2 - \frac{d_3}{d^*} = 0 < \underline{q_{33}} - \left(q_{13}^* + q_{23}^*\right) = 3.5 - (0.4 + 0.2) < d_3^2 = 4. \end{gathered} \tag{4.2}$$

The assumption H_1 holds. By Theorems 3.1 and 3.3, the neural network (2.11) has a unique equilibrium point which is globally exponentially stable, and the unique equilibrium point $(1, 1, 1.5)$ is the optimal solution of this quadratic programming problem.

In the case of $Q = \underline{Q}$, Figure 4 reveals that the projection neural network (2.11) with random initial value $(-0.5, 0.6, -0.8)$ has a unique equilibrium point $(1, 1, 1.5)$ which is globally exponentially stable. In the case of $Q = \overline{Q}$, Figure 5 reveals that the projection neural network (2.11) with random initial value $(0.8, -0.6, 0.3)$ has the same unique equilibrium point $(1, 1, 1.5)$ which is globally exponentially stable. These are in accordance with the conclusion of Theorems 3.1 and 3.3.

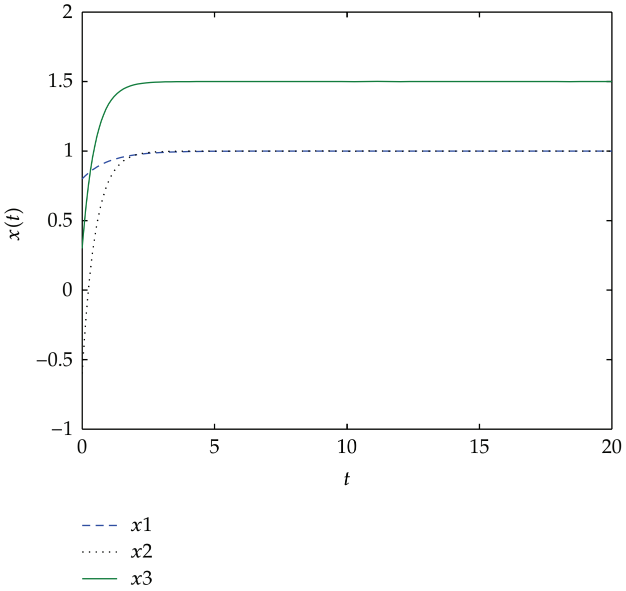

Figure 5: Convergence of the state trajectory of the neural network with random initial value (0.8,–0.6,0.3); $Q = \overline{Q}$ in this example.

5. Conclusion

In this paper, we have developed a new projection neural network for solving interval quadratic programs, the equilibrium point of the proposed neural network is equivalent to the solution of interval quadratic programs. A condition is derived which ensures the existence, uniqueness, and global exponential stability of the equilibrium point. The results obtained are highly valuable in both theory and practice for solving interval quadratic programs in engineering.

Acknowledgment

This paper was supported by the Hebei Province Education Foundation of China (2009157).

References

[1] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithm*, John Wiley & Sons, Hoboken, NJ, USA, 2nd edition, 1993.

[2] M. P. Kennedy and L. O. Chua, “Neural networks for nonlinear programming,” *IEEE Transactions on Circuits and Systems*, vol. 35, no. 5, pp. 554–562, 1988.

[3] Y. Xia, “A new neural network for solving linear and quadratic programming problems,” *IEEE Transactions on Neural Networks*, vol. 7, no. 6, pp. 1544–1547, 1996.

[4] Y. Xia and J. Wang, “A general projection neural network for solving monotone variational inequalities and related optimization problems,” *IEEE Transactions on Neural Networks*, vol. 15, no. 2, pp. 318–328, 2004.

[5] Y. Xia, G. Feng, and J. Wang, “A recurrent neural network with exponential convergence for solving convex quadratic program and related linear piecewise equations,” *Neural Networks*, vol. 17, no. 7, pp. 1003–1015, 2004.

[6] Y. Xia and G. Feng, "An improved neural network for convex quadratic optimization with application to real-time beamforming," *Neurocomputing*, vol. 64, no. 1–4, pp. 359–374, 2005.

[7] Y. Yang and J. Cao, "Solving quadratic programming problems by delayed projection neural network," *IEEE Transactions on Neural Networks*, vol. 17, no. 6, pp. 1630–1634, 2006.

[8] Y. Yang and J. Cao, "A delayed neural network method for solving convex optimization problems," *International Journal of Neural Systems*, vol. 16, no. 4, pp. 295–303, 2006.

[9] Q. Tao, J. Cao, and D. Sun, "A simple and high performance neural network for quadratic programming problems," *Applied Mathematics and Computation*, vol. 124, no. 2, pp. 251–260, 2001.

[10] X. Xue and W. Bian, "A project neural network for solving degenerate convex quadratic program," *Neurocomputing*, vol. 70, no. 13–15, pp. 2449–2459, 2007.

[11] X. Xue and W. Bian, "A project neural network for solving degenerate quadratic minimax problem with linear constraints," *Neurocomputing*, vol. 72, no. 7–9, pp. 1826–1838, 2009.

[12] X. Hu, "Applications of the general projection neural network in solving extended linear-quadratic programming problems with linear constraints," *Neurocomputing*, vol. 72, no. 4–6, pp. 1131–1137, 2009.

[13] X. Hu and J. Wang, "Design of general projection neural networks for solving monotone linear variational inequalities and linear and quadratic optimization problems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 37, no. 5, pp. 1414–1421, 2007.

[14] K. Ding and N.-J. Huang, "A new class of interval projection neural networks for solving interval quadratic program," *Chaos, Solitons and Fractals*, vol. 35, no. 4, pp. 718–725, 2008.

[15] Y. Yang and J. Cao, "A feedback neural network for solving convex constraint optimization problems," *Applied Mathematics and Computation*, vol. 201, no. 1-2, pp. 340–350, 2008.

[16] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational Inequalities and Their Applications*, vol. 88 of *Pure and Applied Mathematics*, Academic Press, New York, NY, USA, 1980.

[17] C. Baiocchi and A. Capelo, *Variational and Quasivariational Inequalities: Applications to Free Boundary Problems*, A Wiley-Interscience Publication, John Wiley & Sons, New York, NY, USA, 1984.