A General Projection Neural Network for Solving Monotone Variational Inequalities and Related Optimization Problems

Youshen Xia, *Senior Member, IEEE,* and Jun Wang, *Senior Member, IEEE*

Abstract—**Recently, a projection neural network for solving monotone variational inequalities and constrained optimization problems was developed. In this paper, we propose a general projection neural network for solving a wider class of variational inequalities and related optimization problems. In addition to its simple structure and low complexity, the proposed neural network includes existing neural networks for optimization, such as the projection neural network, the primal-dual neural network, and the dual neural network, as special cases. Under various mild conditions, the proposed general projection neural network is shown to be globally convergent, globally asymptotically stable, and globally exponentially stable. Furthermore, several improved stability criteria on two special cases of the general projection neural network are obtained under weaker conditions. Simulation results demonstrate the effectiveness and characteristics of the proposed neural network.**

Index Terms—**Global stability, recurrent neural networks, variational inequalities optimization.**

I. Introduction

MANY PROBLEMS in mathematics, physics, and engineering can be formulated as variational inequalities and nonlinear optimization problems [1], [2]. Real-time solutions to these problems are often needed in engineering applications. These problems usually contain time-varying parameters, such as signal processing, system identification, and robot motion control [3], [4], and thus they have to be solved in real time to optimize the performance of dynamical systems. For such real-time applications, conventional numerical methods may not be effective due to stringent requirement on computational time. A promising approach to solving such problems in real time is to employ recurrent neural networks based on circuit implementation [5]–[8]. As parallel computational models, recurrent neural networks possess many desirable properties for real-time information processing. Therefore, recurrent neural networks for optimization, control, and signal processing received tremendous interests. In the past two decades, the theory, methodology, and applications of recurrent neural networks for optimization have been widely investigated (see [8]–[17] and references therein). Tank and Hopfield [5] first proposed a recurrent neural network for solving linear programming problems that was mapped into a closed-loop circuit. Kennedy and Chua [9] proposed a neural network for solving nonlinear convex programming problems by using the penalty function method. The equilibrium points of the Kennedy–Chua network fulfill the Kuhn–Tucker optimality conditions in terms of the penalty function [18]. However, this network can not converge an exact optimal solution and has an implementation problem when the penalty parameter is very large [19]. To avoid using finite penalty parameters, many other studies have been done. Rodríguez–Vázquez *et al.* proposed a switched-capacitor neural network for solving nonlinear convex programming problems, where the optimal solution is assumed to be inside of the bounded feasible region [10]. Zhang *et al.* proposed a second-order neural network for solving nonlinear convex programming problems with equality constraints [11]. The second-order neural network is complex in implementation due to the need for computing varying inverse matrices. Bouzerdoum and Pattison presented a neural network for solving quadratic convex optimization problems with bounded constraints [12]. Tao *et al.* proposed a two-layer neural network for solving a classes of convex optimization with linear equality constraints [13]. We developed several neural networks: primal-dual neural networks for solving linear and quadratic convex programming problems and monotone linear complementary problems [14], [15], a dual neural network for solving strictly convex quadratic programming problems [16], and a projection neural network for solving monotone finite variational inequalities and nonlinear convex optimization problems [17]. The primal-dual neural network have a two-layer structure, while the dual neural network and the projection neural network have one-layer structures and thus have a lower complexity for implementation than two-layer neural networks [13]–[15].

In this paper, we propose a general projection neural network, based on a generalized equation in [20], [21], for solving a wider class of monotone variational inequalities and related optimization problems. The proposed neural network has a one-layer structure with a low model complexity and contains existing neural networks for constrained optimization, such as the primal-dual neural networks, the dual neural network, and the projection neural network, as its special cases. The proposed neural network is shown to be stable in the sense of

Manuscript received October 7, 2002; revised October 13, 2003. This work was supported by the Hong Kong Research Grants Council under Grant CUHK4165/03E.

Y. Xia is with Department of Applied Mathematics, Nanjing University of Posts and Telecommunications, Nanjing, China (e-mail: ysxia2001@yahoo.com).

J. Wang is with the Department of Automation and Computer-Aided Engineering, The Chinese University of Hong Kong, New Territories, Hong Kong, China (e-mail: jwang@acae.cuhk.edu.hk).

Digital Object Identifier 10.1109/TNN.2004.824252

Lyapunov, globally asymptotically stable, and globally exponentially stable, respectively under different mild conditions. Furthermore, several improved stability conditions for two special cases of the general projection neural network are obtained under weaker conditions. Illustrative examples demonstrate the performance and effectiveness of the proposed neural network.

This paper is organized as follows. In the next section, a general projection neural network and its advantages are described. In Section III, the convergence properties of the proposed neural network, including global asymptotic stability and global exponential stability, are studied under different mild conditions. In Section IV, several illustrative examples are presented. Section V gives the conclusions of this paper.

II. Model Description

We propose a general projection neural network with its dynamical equation defined as

$$\frac{du}{dt} = \Lambda\{P_X(G(u) - F(u)) - G(u)\} \tag{1}$$

where $u \in R^n$ is the state vector, $\Lambda = \mathrm{diag}(\lambda_i)$ is a positive diagonal matrix, $F(u)$ and $G(u)$ are continuously differentiable vector-valued functions from R^n into R^n, $X = \{u \in R^n \,|\, l_i \le u_i \le h_i, i = 1, \ldots, n\}$, $P_X : R^n \to X$ is a projection operator defined by $P_X(u) = [P_X(u_1), \ldots, P_X(u_n)]^T$, and $P_X(u_i)$ is a piecewise activation function given by

$$P_X(u_i) = \begin{cases} l_i & u_i < l_i \\ u_i & l_i \le u_i \le h_i \\ h_i & u_i > h_i. \end{cases}$$

The dynamic equation described in (1) can be easily realized in a recurrent neural network with a single layer structure as shown in Fig. 1. From the Fig. 1 we can see that the circuit realizing the proposed neural network consists of $4n$ summers, n integrators, n piecewise linear activation functions, and n processors for $G(u)$ and $F(u)$. Therefore, the network complexity depends on the mapping $G(u)$ and $F(u)$.

In addition to its low complexity for realization, the general projection neural network in (1) has several advantages. First, it is a significant generalization of some existing neural networks for optimization. For example, let $G(u) = u$, then the proposed neural network model becomes the projection neural network model [17] given by

$$\frac{du}{dt} = \Lambda\{P_X(u - F(u)) - u\}. \tag{2}$$

In the affine case that $F(u) = Mu + p$ where $M \in R^{n\times n}$ is a positive semi-definite matrix and $p \in R^n$, the proposed neural network model becomes the primal-dual neural network model [14]

$$\frac{du}{dt} = \Lambda\{P_X(u - (Mu + p)) - u\}. \tag{3}$$

Let $F(u) = u$, then the proposed neural network model becomes

$$\frac{du}{dt} = \Lambda\{P_X(G(u) - u) - G(u)\}. \tag{4}$$

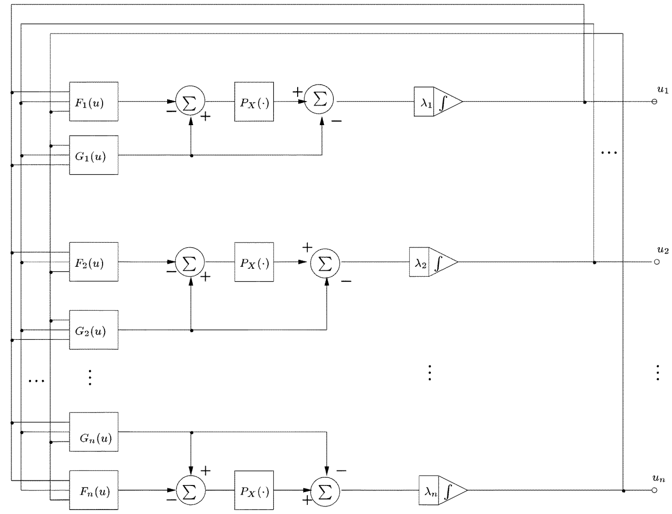


Fig. 1. Block diagram of the general projection neural network in (1).

In the affine case that $G(u) = Wu + q$ where $W \in R^{n\times n}$ is a positive semi-definite matrix and $q \in R^n$, the proposed neural network model becomes the dual neural network model [16]

$$\frac{du}{dt} = \Lambda\{P_X(Wu + q - u) - Wu - q\}. \tag{5}$$

Since $G(u)$ and $F(u)$ may be nonlinear for (1), the proposed general projection neural network extends the projection neural network and the dual neural network in term of the structure. As a result, the general projection neural network in (1) is useful for solving a wider class of variational inequalities and related optimization problems. This is because it is intimately related to the following general variational inequality (GVI) [20]: find $u^* \in X$ such that $G(u^*) \in X$ and

$$(u - G(u^*))^T F(u^*) \ge 0, \quad \forall u \in X. \tag{6}$$

From [21] it can be seen that solving GVI is equivalent to finding a zero of the generalized equation

$$P_X(G(u) - F(u)) - G(u) = 0. \tag{7}$$

Therefore, the equilibrium point of the general projection neural network in (1) solves GVI. This property shows that the existence of the equilibrium point of (1) is equivalent to the one of the solutions of GVI. As for the existence of the solutions of GVI, the reader is referred to related papers [20]–[22]. It is well known that GVI is viewed as the general framework of unifying the treatment of many optimization, economic, and engineering problems [24], [25]. For example, GVI includes two useful models: the variational inequalities and general complementarity problems. The variational inequality problem is to find an $u^* \in X$ such that

$$(u - u^*)^T F(u^*) \ge 0, \quad \forall u \in X. \tag{8}$$

The general complementarity problem is to find an $u^* \in R^n$ such that

$$G(u) \ge 0, \quad F(u) \ge 0, \quad G(u)^T F(u) = 0. \tag{9}$$

Other examples will be illustrated in Section V. Because the desirable solutions to GVI can be obtained by tracking the continuous trajectory of (1), the proposed neural network in (1) is attractive alternative as a real-time solver for many optimization and related problems.

III. Global Convergence

In this section, we show under mild conditions that the general projection neural network is globally convergent, globally asymptotically stable, and globally exponentially stable.

A. Preliminaries

For the convenience of later discussions, we first introduce to related definitions and a lemma.

Definition 1: A mapping F is said to be G-monotone at u^* if

$$(F(u) - F(u^*)^T(G(u) - G(u^*)) \geq 0, \quad \forall u \in R^n \tag{10}$$

where F is said to be strictly G-monotone at u^* if the strict inequality holds whenever $u \neq u^*$, and G-strongly monotone at u^* if there exists a constant $\beta > 0$ such that

$$(F(u) - F(u^*))^T(G(u) - G(u^*)) \geq \beta \|u - u^*\|^2, \quad \forall u \in R^n \tag{11}$$

where $\|\cdot\|$ denotes the l_2-norm of R^n. In particular, when $G(u) = u, F$ is said to be monotone, strictly monotone, and strongly monotone at u^*, respectively.

The above definitions of monotonicity are easily seen as listed in an order from weak to strong.

Definition 2: Let $u(t)$ be a solution trajectory of (1) with the initial point $u_0 \in R^n$. The general projection neural network in (1) is said to be globally convergent to a set Y if for each initial point u_0

$$\lim_{t\to\infty} \inf_{y\in Y} \|u(t) - y\| = 0.$$

In particular, the general projection neural network in (1) is globally asymptotically stable at u^* if it is stable at u^* in the sense of Lyapunov and $Y = \{u^*\}$. The general projection neural network in (1) is said to be globally exponentially stable at u^* if every trajectory $u(t)$ of (1) with the initial point $u(t_0) \in R^n$ satisfies

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

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

It is clear that the trajectory $u(t)$, starting from $u(t_0)$, approaches u^* at least as fast as the decaying exponential $\exp(-\eta(t - t_0))$ and the global exponential convergence is definitely the global convergence.

Throughout this paper we denote the Jacobian matrix of $G(u)$ and $F(u)$ as $\nabla G(u)$ and $\nabla F(u)$ respectively.

Lemma 1 [22]: Assume that the set $X \subset R^n$ is a closed convex set. Then

$$(v - P_X(v))^T(P_X(v) - x) \geq 0, \quad v \in R^n, \quad x \in X$$

and

$$\|P_X(v) - P_X(v)\| \leq \|u - v\|, \quad v, u \in R^n.$$

We now establish our main results on the global convergence of the proposed neural network in (1).

B. General Case

The following lemma shows the the existence of the solution trajectory of (1).

Lemma 2: For each initial point $u(t_0) \in R^n$, there exists a unique continuous solution $u(t)$ as (1) over $[t_0, \tau)$.

Proof: Let $T(u) = P_X(G(u) - F(u)) - G(u)$. Using Lemma 1 we have that for any $u, v \in R^n$

$$\begin{aligned}\|T(u) - T(v)\| &\leq \|P_X(G(u) - F(u)) - G(u) \\ &\quad - (P_X(G(v) - F(v)) - G(v)\| \\ &\leq \|F(u) - F(v)\| + 2\|G(u) - G(v)\|.\end{aligned}$$

Since $F(u)$ and $G(u)$ are continuously differentiable in R^n, they are locally Lipschitz continuous. Thus $T(u)$ is locally Lipschitz continuous also. By the existence theory of ordinary differential equations [26] we see that for any initial point $u(t_0)$ there is a unique solution $u(t)$ of (1) over $[t_0, \tau)$. □

Theorem 1: Assume that there exists an equilibrium point u^* of (1) so that

$$S^* = \{u \in R^n \mid F(u) + G(u) = F(u^*) + G(u^*)\}$$

is bounded and $F(u)$ is G-monotone at u^*. If $\nabla F(u) + \nabla G(u)$ is symmetric and positive semi-definite in R^n, then the general projection neural network in (1) is stable in the Lyapunov sense and is globally convergent to an equilibrium points of (1). Specially, the general projection neural network in (1) is globally asymptotically stable if it has a unique equilibrium point.

Proof: By Lemma 2 it can be seen that for any given initial point $u_0 \in R^n$ there exists a unique continuous solution $u(t)$ for (1) over $[t_0, \tau)$. Define the following Lyapunov function

$$\begin{aligned}V(u) = &\int_0^1 (u - u^*)^T \Lambda^{-1} \{F(u^* + s(u - u^*)) \\ &+ G(u^* + s(u - u^*))\}\, ds - (u - u^*)^T \Lambda^{-1}(F(u^*) + G(u^*))\end{aligned}$$

where u^* is an equilibrium point of (1). Since $\nabla F(u) + \nabla G(u)$ is symmetric and positive semi-definite in $R^n, V(u)$ is continuously differentiable and convex in R^n [26] and $\nabla V(u) = \Lambda^{-1}(F(u) + G(u) - F(u^*) - G(u^*))$, where $\nabla V(u)$ is the gradient of V. Thus $\nabla V(u^*) = 0$ if and only if u^* is a global minimizer of V. Furthermore, since u^* is the equilibrium point of (1), u^* satisfies

$$P_X(G(u) - F(u)) = G(u)$$

which is equivalent to the following

$$G(u^*) \in X, \quad (u - G(u^*))^T F(u^*) \geq 0, \quad \forall u \in X.$$

Then

$$\{P_X(G(u)-F(u))-G(u^*)\}^T F(u^*) \geq 0, \quad \forall u \in R^n.$$

In the projection inequality of Lemma 1, let $v = G(u) - F(u)$ and $x = G(u^*)$, we get

$$\begin{aligned}&[P_X(G(u)-F(u))-G(u^*)]^T \\ &\quad \times [G(u)-F(u)-P_X(G(u)-F(u))] \geq 0, \quad \forall u \in R^n.\end{aligned}$$

Adding the two resulting inequalities yields [21]

$$\begin{aligned}&\{P_X(G(u)-F(u))-G(u^*)\}^T \\ &\quad \{-F(u)+F(u^*)-P_X(G(u)-F(u))+G(u)\} \geq 0\end{aligned}$$

then

$$\begin{aligned}&\{F(u)-F(u^*)+G(u)-G(u^*)\}^T \\ &\{P_X(G(u)-F(u))-G(u)\} \\ &\leq -(G(u)-G(u^*))^T(F(u)-F(u^*)) \\ &\quad - \|G(u)-P_X(G(u)-F(u))\|^2.\end{aligned}$$

Note that for any $u \in R^n$

$$(G(u)-G(u^*))^T(F(u)-F(u^*)) \geq 0.$$

It follows that

$$\begin{aligned}\frac{dV}{dt} &= \nabla V(u)^T \frac{du}{dt} \\ &= \{F(u)-F(u^*)+G(u)-G(u^*)\}^T \Lambda^{-1} \frac{du}{dt} \\ &= \{F(u)-F(u^*)+G(u)-G(u^*)\}^T \{P_X(G(u) \\ &\quad - F(u)) - G(u)\} \\ &\leq -\|G(u)-P_X(G(u)-F(u))\|^2 \leq 0\end{aligned}$$

and hence

$$\{u(t)\} \subset X_0 = \{u \in X \mid V(u) \leq V(u_0)\}.$$

Note that the set of global minimizers of V is nonempty and bounded, then $V(u^k) \to +\infty$ as $\|u^k\| \to +\infty$ and $\{u^k\} \subset R^n$. It follows that all level sets of V are bounded [26]. Thus the solution trajectory $u(t)$ is bound and hence $\tau = +\infty$. On the other side, it can be seen that $dV/dt = 0$ if only and if $du/dt = 0$. From Lyapunov Theorem [26] it follows that the general projection neural network in (1) is stable in the Lyapunov sense and is globally convergent to the set of equilibrium points of (1). Specially, if $X^* = \{u^*\}$, then $\lim_{t\to\infty} u(t) = u^*$. Therefore, the general projection neural network in (1) is globally asymptotically stable. □

Remark 1: It is easy to see that S^*, defined in Theorem 1, is nonempty and $S^* = \{u^*\}$ when mapping $G + F$ is invertible.

Theorem 2: Assume that $F(u)$ is G-strongly monotone at u^* and $G(u)+F(u)$ is strongly monotone at u^*. If $\nabla F(u)+\nabla G(u)$ is symmetric and has an upper bound in R^n, then the general projection neural network in (1) is globally exponentially stable at u^*.

Proof: Let Lyapunov function $V(u)$ be defined in Theorem 1. Then

$$\begin{aligned}\frac{dV}{dt} &\leq \{F(u)-F(u^*)+G(u)-G(u^*)\}^T \\ &\quad \times \{P_X(G(u)-F(u))-G(u)\} \\ &\leq -(G(u)-G(u^*))^T(F(u)-F(u^*)).\end{aligned}$$

Since $\nabla F(u)+\nabla G(u)$ has an upper bound, there exists a $\delta_1 > 0$ such that for all $z \in R^n$

$$\|\nabla F(z)+\nabla G(z)\| \leq \delta_1.$$

Note that

$$\begin{aligned}V(u) &= \int_0^1 (u-u^*)^T \Lambda^{-1} \{F(u^*+s(u-u^*)) \\ &\quad + G(u^*+s(u-u^*)) - F(u^*) - G(u^*)\}\, ds \\ &= \int_0^1 s(u-u^*)^T \Lambda^{-1} \int_0^1 (\nabla F(z_t) \\ &\quad + \nabla G(z_t))(u-u^*)\, dt\, ds \\ &\leq \frac{1}{\lambda_{\min}} \int_0^1 \int_0^1 s\|\nabla F(z_t)+\nabla G(z_t)\|\|u-u^*\|^2\, dt\, ds\end{aligned}$$

where $\lambda_{\min} = \min_i\{\lambda_i\}$. Then

$$V(u) \leq \int_0^1 s \int_0^1 \delta\|u-u^*\|^2\, dt\, ds = \frac{\delta}{2}\|u-u^*\|^2$$

where $\delta = \delta_1/\lambda_{\min} > 0$. It follows:

$$\begin{aligned}\frac{d}{dt}V(u) &\leq -(G(u)-G(u^*))^T(F(u)-F(u^*)) \\ &\leq -\beta\|u-u^*\|^2 \\ &\leq -\frac{\delta}{2\beta}V(u)\end{aligned}$$

and thus

$$V(u) \leq V(u_0)\exp(-\mu_1(t-t_0)), \quad \forall t \geq t_0$$

where $\mu_1 = \delta/2\beta$. Note that the condition that $G(u)+F(u)$ is strongly monotone implies that $V(u)$ is uniformly convex, then there exists $\mu_2 > 0$ such that

$$V(u)-V(u^*) \geq \nabla V(u^*)(u-u^*) + \mu_2\|u-u^*\|^2.$$

Since $\nabla V(u^*) = 0$ and $V(u^*) = 0, V(u) \geq \mu_2\|u-u^*\|^2$. Therefore

$$\|u-u^*\| \leq \sqrt{\frac{V(u_0)}{\mu_2}} \exp\left(-\frac{\mu_1(t-t_0)}{2}\right) \quad \forall t \geq t_0.$$

Therefore, the general projection neural network in (1) is globally exponentially stable. □

As an immediate corollary of both Theorems 1 and 2, we obtain the following stability results of (2) and (4).

Corollary 1: Assume that $G(u) = u$ and $\nabla F(u)$ is symmetric. If $F(u)$ is monotone, then the projection neural network in (2) is stable in the Lyapunov sense and is globally convergent to an equilibrium point of (2). If $F(u)$ is strongly monotone and

$\nabla F(u)$ has an upper bound in R^n, then the projection neural network in (2) is globally exponentially stable.

The result of Corollary 1 is presented in [17].

Corollary 2: Assume that $F(u) = u$ and $\nabla G(u)$ is symmetric. If $G(u)$ is monotone, then the neural network in (4) is stable in the Lyapunov sense and is globally convergent to an equilibrium point of (4). If $G(u)$ is strongly monotone and $\nabla G(u)$ has an upper bound in R^n, then the neural network in (4) is globally exponentially stable.

As a special case the $G(u) = Wu + q$, where $W \in R^{n\times n}$ is symmetric and positive semidefinite and $q \in R^n$, the result of Corollary 2 is presented in [16].

C. Two Special Cases

In what follows, we further study the global stability of two special cases of the general projection neural network in (1).

We first consider the case of the projection neural network in (2), where $\nabla F(u)$ is asymmetric. To study the global asymptotical stability of (2) we first introduced a novel Lyapunov function in [17]

$$V(u) = \frac{1}{2}\|u - u^*\|^2 + V_0(u)$$

where

$$V_0(u) = (u - P_X(u - F(u)))^T F(u) - \frac{1}{2}\|P_X(u - F(u)) - u\|^2$$

called as the regularized gap function in [23]. Based on the above Lyapunov function we proved the following inequality

$$\frac{dV(u)}{dt} \le -F(u)^T(u - u^*) - (P_X(u - F(u)) - u)^T \times \nabla F(u)(P_X(u - F(u)) - u).$$

The global asymptotical stability of the projection neural network in (2) is thus obtained when $\nabla F(u)$ is positive definite. We here establish a result on the global convergence of (2) when $\nabla F(u)$ is positive semi-definite.

Theorem 3: Assume that $\nabla F(u)$ is positive semidefinite in R^n. If

$$\begin{cases} F(u)^T(u - u^*) = 0 \\ (P_X(u - F(u)) - u)^T \nabla F(u)(P_X(u - F(u)) - u) = 0 \end{cases}$$

implies that u is a solution to (8), then the projection neural network in (2) with any initial point $u(t_0) \in X$ is always convergent to an equilibrium point of (2).

Proof: Let $u(t)$ be the solution trajectory of (2) with the initial point $u(t_0) \in X$. Then

$$\frac{dV(u)}{dt} \le -F(u)^T(u - u^*) - (P_X(u - F(u)) - u)^T \times \nabla F(u)(P_X(u - F(u)) - u) \le 0.$$

It follows that $dV(u)/dt = 0$ if and only if:

$$\begin{cases} F(u)^T(u - u^*) = 0 \\ (P_X(u - F(u)) - u)^T \nabla F(u)(P_X(u - F(u)) - u) = 0. \end{cases}$$

By the given condition we see that $dV(u)/dt = 0$ implies that u is an equilibrium point of (2). On the other side, since $V(u^k) \to +\infty$ as $\|u^k\| \to +\infty$, for any initial point $u_0 \in R^n$ there exists a convergent subsequence $\{u(t_k)\}$ such that $\lim_{k\to\infty} u(t_k) = \hat{u}$ where $\hat{u}$ satisfies

$$\frac{dV(u)}{dt} = 0.$$

Then $\hat{u}$ is an equilibrium point of (2). Again define another Lyapunov function

$$\hat{V}(u) = \frac{1}{2}\|u - \hat{u}\|^2 + V_0(u).$$

It can be seen that $\hat{V}(\hat{u}) = 0$ and $d\hat{V}(u(t))/dt \le 0$. It follows that for $\forall \epsilon > 0$ there exists $q > 0$ such that for any $t_k \ge t_q$

$$\hat{V}(u(t_k)) < \epsilon.$$

It implies that

$$\|(u(t) - \hat{u})\| \le \hat{V}(u(t)) \le \epsilon, \quad \forall t \ge t_q.$$

Thus

$$\lim_{t\to\infty} \|(u(t) - \hat{u})\| = 0$$

and hence

$$\lim_{t\to\infty} u(t) = \hat{u}.$$

Therefore, the projection neural network in (2) is globally convergent to an equilibrium point of (2). □

There are partial results [17], [27], [28] on the global exponential stability of (2) under the condition that $\nabla F(u)$ is uniformly positive definite and other conditions. Also, paper [29] studied the exponential stability of (2) based on a similar Lyapunov function defined in paper [17]. However, the given proof is not complete (for example, see the inequality (6) in Lemma 4, in [19]). The following result in [29] removes all additional conditions and the proof is complete.

Theorem 4: If $F(u)$ is uniformly positive definite in R^n, then the projection neural network in (2) with any initial point $u(t_0) \in X$ is exponentially stable.

Proof: Let $u(t)$ be the solution of (2) with the initial point $u(t_0)$. Since $\nabla F(u)$ is uniformly positive definite, there exists a positive constant δ such that

$$x^T \nabla F(u) x \ge \delta\|x\|^2, \quad \forall x, u \in R^n.$$

We now consider the following Lyapunov function

$$V(u) = V_0(u) + \frac{1}{2}\|u - u^*\|^2$$

where u^* is a unique equilibrium point of (2) and

$$V_0(u) = (2\delta + 1)(F(u)^T(u - P_X(u - F(u))) - \frac{1}{2}\|u - P_X(u - F(u))\|^2|.$$

Similar to the analysis in [17], we have

$$\frac{dV_0(u)}{du} = (2\delta + 1)(F(u) + (\nabla F(u) - I)(u - P_X(u - F(u))).$$

Then

$$\begin{aligned}\frac{dV(u)}{dt} &= \nabla V(u)^T \frac{du}{dt} \\ &= (2\delta+1)\{F(u)+(\nabla F(u)-I)(u-P_X(u-F(u)))\}^T \\ &\quad \times (P_X(u-F(u))-u\} \\ &\quad + (u-u^*)^T(P_X(u-F(u))-u) \\ &= 2\delta F(u)^T(P_X(u-F(u))-u)-(2\delta+1) \\ &\quad \times (u-P_X(u-F(u)))^T\nabla F(u)(u-P_X(u-F(u))) \\ &\quad + (F(u)+u-u^*)^T(P_X(u-F(u))-u)+(2\delta+1) \\ &\quad \times \|u-P_X(u-F(u))\|^2.\end{aligned}$$

One one side, we can get

$$F(u)^T(P_X(u-F(u))-u) \le -\|P_X(u-F(u))-u\|^2.$$

It follows that $V_0(u) \ge 0$ and

$$V(u) \ge \frac{1}{2}\|u-u^*\|^2.$$

On the other side, we can obtain

$$\begin{aligned}&\{F(u)+u-u^*\}^T(P_X(u-F(u))-u) \\ &\quad + F(u)^T(u-u^*) \le -\|P_X(u-F(u))-u\|^2.\end{aligned}$$

Note that

$$\begin{aligned}&(u-P_X(u-F(u)))^T\nabla F(u)(u-P_X(u-F(u))) \\ &\qquad \ge \delta\|P_X(u-F(u))-u\|^2\end{aligned}$$

and

$$F(u)^T(u-u^*) \ge (F(u)-F(u^*))^T(u-u^*) \ge \delta\|u-u^*\|^2.$$

It follows

$$\begin{aligned}\frac{dV(u)}{dt} &\le -F(u)^T(u-u^*)-\|P_X(u-F(u))-u\|^2 \\ &\quad + 2\delta F(u)^T(P_X(u-F(u))-u) \\ &\quad - (2\delta+1)\delta\|P_X(u-F(u))-u\|^2 \\ &\quad + (2\delta+1)\|P_X(u-F(u))-u\|^2 \\ &\le -\delta\|P_X(u-F(u))-u\|^2-2\delta^2 \\ &\quad \times \|P_X(u-F(u))-u\|^2+2\delta\| \\ &\quad \times P_X(u-F(u))-u\|^2 \\ &\quad - 2\delta(F(u)^T(u-P_X(u-F(u)))-\delta\|u-u^*\|^2 \\ &\le -2\delta(F(u)^T(u-P_X(u-F(u))) \\ &\quad - \frac{1}{2}\|P_X(u-F(u))-u\|^2)-\delta\|u-u^*\|^2 \\ &\le -\frac{2\delta}{2\delta+1}\left\{\frac{1}{2}\|u-u^*\|^2+(2\delta+1)(F(u)^T\right. \\ &\quad \left.\times (u-P_X(u-F(u)))-\frac{1}{2}\|u-P_X(u-F(u))\|^2\right\} \\ &\le -\frac{2\delta}{2\delta+1}V(u).\end{aligned}$$

Then

$$V(u) \le V(u_0)\exp(-\beta(t-t_0))$$

where

$$\beta = \frac{2\delta}{2\delta+1} > 0.$$

It follows

$$\|u-u^*\| \le \sqrt{2V(u_0)}\exp\left(-\frac{\beta(t-t_0)}{2}\right).$$

Therefore, the projection neural network in (2) is exponentially stable. □

We next consider the case that both F and G are affine, i.e., $F(u)=Mu+q, G(u)=Nu+c, M,N \in R^{n\times n}$, and $q,c \in R^n$. The corresponding neural network model is then given by

$$\frac{du}{dt} = Q\{P_X(Wu+p)-Nu-c\} \tag{12}$$

where $p=c-q, W=N-M$, and Q is an $n\times n$ scaling matrix. As an immediate corollary of Theorems 1 and 2, we have the following convergence results.

Corollary 3: Assume that Q is symmetry and positive definite. If $N+M$ is symmetric and positive definite, then the neural network in (12) is globally convergent to an equilibrium point of (12) when M^TN is positive semi-definite, and is globally exponentially stable when M^TN is positive definite.

Proof: From the conclusion of Theorems 1 and 2 it follows the conclusion of Corollary 3. □

Corollary 4: Assume that matrix $Q=B(N^T+M^T)$, where B is an $n\times n$ symmetry and positive definite matrix. The neural network in (12) is globally convergent to an equilibrium point of (12) when M^TN is positive semi-definite, and is globally exponentially stable when M^TN is positive definite.

Proof: Consider the following function

$$V(u) = \frac{1}{2}\|C(u-u^*)\|^2$$

where u^* is an equilibrium point of (12) and C is an $n\times n$ symmetry and positive definite matrix satisfying $C^2=B^{-1}$. Then

$$\begin{aligned}\frac{d}{dt}V(u) &= (u-u^*)^TB^2\frac{du}{dt} \\ &= (u-u^*)^TB^2Q\{P_X(Wu+p)-Nu-c\}.\end{aligned}$$

By the proof of Theorem 1, we see that

$$\begin{aligned}&\{F(u)-F(u^*)+G(u)-G(u^*)\}^T \\ &\quad \times \{P_X(G(u)-F(u))-G(u)\} \\ &\quad \le -(G(u)-G(u^*))^T(F(u)-F(u^*)) \\ &\quad - \|G(u)-P_X(G(u)-F(u))\|^2.\end{aligned}$$

Substituting $G(u)=Nu+c$ and $F(u)=Mu+q$ into the above inequality, we have

$$\begin{aligned}&\{M(u-u^*)+N(u-u^*)\}^T \\ &\quad \times \{P_X(Nu+c-Mu-q)-Nu-c\} \\ &\quad \le -(M(u-u^*))^TN(u-u^*) \\ &\quad - \|Nu+c-P_X(Wu+p)\|^2.\end{aligned}$$

That is

$$\begin{aligned}&(u-u^*)^T(M^T+N^T)\{P_X(Wu+p)-Nu-c\} \\ &\quad \le -(u-u^*)^TM^TN(u-u^*) \\ &\quad - \|P_X(Wu+p)-Nu-c\|^2.\end{aligned}$$

It follows

$$\frac{d}{dt}V(u) = (u-u^*)^T B^2 Q\{P_X(Wu+p) - Nu - c\} \\ \leq -(u-u^*)^T M^T N(u-u^*) \\ - \|P_X(Wu+p) - Nu - q\|_2^2 \leq 0.$$

Similar to the analysis of Theorem 3 we can obtain the rest of the proof. □

Corollary 5: Assume that $N = I$ and $Q = I + M^T$. The neural network in (12) is globally convergent to an equilibrium point of (12) when M is positive semi-definite, and is globally exponentially stable when M is positive definite.

IV. Illustrative Examples

In order to demonstrate the effectiveness and performance of the general projection neural network, in this section, we discuss several illustrative examples. The simulation is conducted in MATLAB.

Example 1: Consider the implicit complementarity problem (ICP) [30]: find $u \in R^n$ such that

$$u - \Phi(u) \geq 0, \quad F(u) \geq 0, \quad (u-\Phi(u))^T F(u) = 0$$

where $\Phi(u) = \Theta(Mu+q)$

$$F(u) = Mu + q = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & 2 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix}_{n\times n} u + \begin{pmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{pmatrix}_{n\times 1}$$

and

$$\Theta(u) = \begin{pmatrix} -1.5u_1 + 0.25u_1^2 \\ -1.5u_2 + 0.25u_2^2 \\ \cdot \\ \cdot \\ \cdot \\ -1.5u_{n-1} + 0.25u_{n-1}^2 \\ -1.5u_n + 0.25u_n^2 \end{pmatrix}_{n\times 1}.$$

It is easy to see that the existing projection neural network in (2) can not be applied to solve the ICP. However, the proposed general projection neural network in (1) can be applied to solve the ICP since the ICP can be viewed as the GNCP where $G(u) = u - \Phi(u)$ and $F(u) = Mu + q$. Its dynamical equation is given by

$$\frac{du}{dt} = \Lambda\{(u - \Phi(u) - F(u))^+ - u + \Phi(u)\} \tag{13}$$

where $(u)^+ = [(u_1)^+, \ldots, (u_n)^+]^T$ and $(u_i)^+ = \max\{0, u_i\}$ for $i = 1, \ldots, n$. All simulation results show that the trajectory of (13) with any initial point is always convergent to an exact solution to ICP. For example, let $\Lambda = 2I$ and let the initial point be random. Fig. 2 shows the transient behaviors of the complementarity error $(u - \Phi(u))^T F(u)$ based on (13) with $n = 10, 30, 60, 90$. Fig. 3 shows the transient behavior of $u(t)$ based on (13) with $n = 10$.

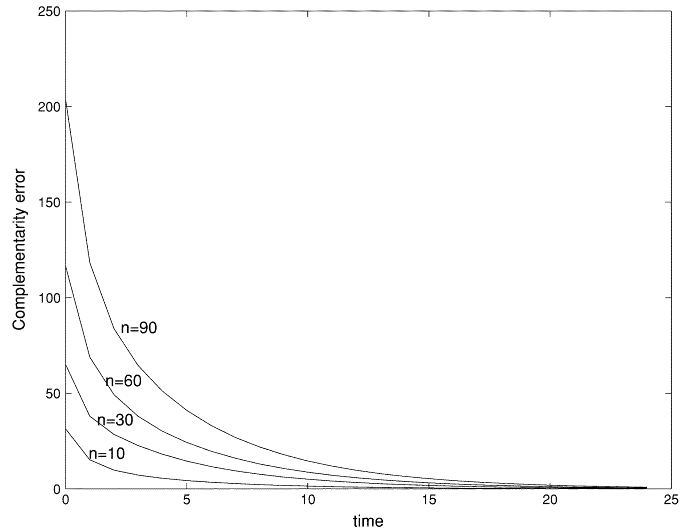


Fig. 2. Complementarity error based on the proposed neural network in (13) for solving ICP in Example 1.

Example 2: Consider the variational inequality problem (VIP) with nonlinear constraints: find $x \in R^{10}$ such that

$$(x - f(x^*))^T x^* \geq 0, \quad \forall x \in X \tag{14}$$

where $X = \{x \in R^{10} \mid h(x) \leq 0, x \geq 0\}$ and, see the equation at the bottom of the following page. This problem has an optimal solution given in [31] (see at the bottom of the page). By the Kuhn-Tucker condition [2], we see that there exists $y \in R^8$

$$f(x) = \begin{pmatrix} 2x_1 - 14 + x_2 \\ x_1 + 2x_2 - 16 \\ 2(x_3 - 10) \\ 8(x_4 - 5) \\ 2(x_5 - 3) \\ 4(x(6) - 1) \\ 10x_7 \\ 14(x_8 - 11) \\ 4(x_9 - 10) \\ 2(x_{10} - 7) \end{pmatrix}, \qquad h(x) = \begin{pmatrix} 3(x_1-2)^2 + 4(x_2-3)^2 + 2x_3^2 - 7x_4 - 120 \\ 5x_1^2 + (x_3-6)^2 + 8x_2 - 2x_4 - 40 \\ (x_1-8)^2/2 + 2(x_2-4)^2 + 3x_5^2 - x_6 - 30 \\ x_1^2 + 2(x_2-2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \\ 4x_1 + 5x_2 - 3x_7 + 9x_8 - 105 \\ 10x_1 - 8x_2 - 17x_7 + 2x_8 \\ 12(x_9-8)^2 - 3x_1 + 6x_2 - 7x_{10} \\ -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \end{pmatrix}$$

$$x^* = [2.172, 2.364, 8.774, 5.096, 0.991, 1.431, 1.321, 9.829, 8.280, 8.376]^T.$$

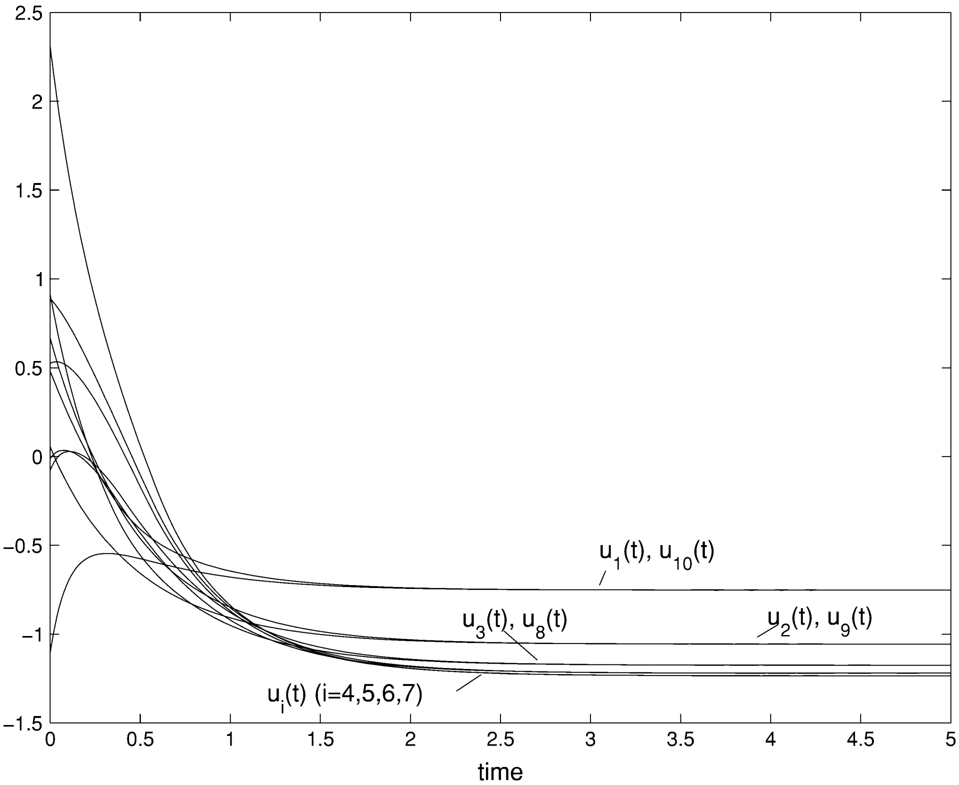


Fig. 3. Global convergence of the neural network in (13) for solving the ICP in Example 1.

such that x solves the above VIP if and only if $u=(x,y)\in R^{18}$ solves the GNCP where $F(u)=u$ and

$$G(u)=\begin{pmatrix} f(x)+\nabla h(x)y \\ -h(x)\end{pmatrix}.$$

It is easy to see that the existing dual neural network in (5) can not be applied to solve the GNCP. However, the general projection neural network in (1) can be applied to solve the GNCP, and its dynamical equation is given by

$$\frac{du}{dt}=\Lambda\{(G(u)-u)^+-G(u)\} \tag{15}$$

where $(u)^+=[(u_1)^+,\ldots,(u_3)^+]^T$ and $(u_i)^+=\max\{0,u_i\}$ for $i=1,\ldots,18$. All simulation results show the trajectory of (15) with any initial point always is convergent to $u^*=(x^*,y^*)$. For example, let $\Lambda=2I$ and let the initial point be zero. A solution to the GNCP is obtained as $x^1=[2.178, 2.368, 8.743, 5.09, 0.991, 1.430, 1.322, 9.823, 8.286, 8.371]^T$, where $t=6$. Fig. 4 shows the transient behavior of $x(t)$ based on (15) with a random initial point.

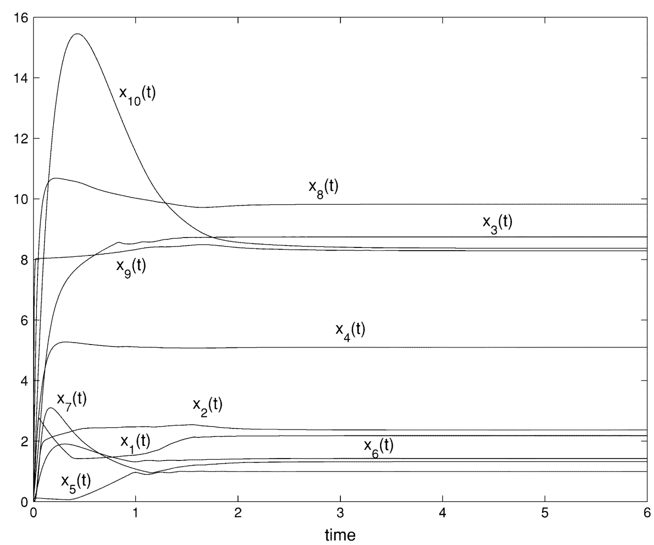


Fig. 4. Global convergence of the proposed neural network in (15) for solving GNCP in Example 2.

The following two examples will illustrate the result of Theorem 4.

Example 3: Consider the variational inequality problem (VIP): find $u\in R^4$ such that

$$(u-u^*)^TF(u^*)\geq 0, u\in X \tag{16}$$

where

$$F(u)=\begin{pmatrix} 3u_1-1/(u_1+0.1)+3u_2-2 \\ 3u_1+3u_2 \\ 4u_3+4u_4 \\ 4u_3+4u_4-1/(u_4+0.1)-3\end{pmatrix}$$

$X=\{u\in R^4\,|\,h\geq u\geq l\}, l=[1,0,0,1]^T$, and $h=[100,100,100,100]^T$. This problem has only one solution $u^*=[1,0,0,1]^T$ and $F(u^*)=[0,3,4,0]^T\neq 0$. Moreover

$$\nabla F(u)=\begin{pmatrix} 3+(u_1+0.1)^{-2} & 3 & 0 & 0 \\ 3 & 3 & 0 & 0 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 4 & 4+(u_4+0.1)^{-2}\end{pmatrix}$$

is uniformly positive definite on X. We use the projection neural network in (2) to solve the above VIP. Since $F(u)$ does

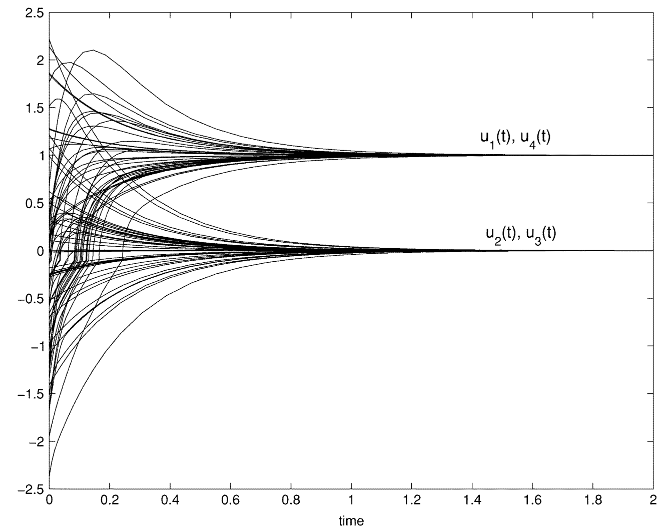


Fig. 5. Global Exponential stability of the projection neural network in (2) for solving the VIP in Example 3.

not satisfy the Lipschitz condition, the existing results [27], [28] cannot ascertain the exponential stability of the projection neural network in (2). However, from Theorem 4 it follows that the projection neural network in (2) is exponentially stable. All simulation results show that the corresponding neural network in (2) is always exponentially stable at u^*. For example, Fig. 5 displays the trajectory of (2) with 20 random initial points, where $\Lambda = 4I$.

Example 4: Consider the nonlinear complementarity problem (NCP)

$$u^T F(u) = 0, \quad F(u) \geq 0, \quad u \geq 0 \tag{17}$$

where

$$F(u) = \begin{pmatrix} 2u_1 \exp(u_1^2 + (u_2-1)^2) + u_1 - u_2 - u_3 + 1 \\ 2(u_2-1)\exp(u_1^2 + (u_2-1)^2) - u_1 + 2u_2 + 2u_3 + 3 \\ -u_1 + 2u_2 + 3u_3 \end{pmatrix}.$$

This problem has only one solution $u^* = [0, 0.167, 0]^T$ and $F(u^*) = [0.833, 0, 0.334]^T$. According to [22], u^* is a solution to the above NCP if and only if u^* satisfies the following equation

$$P_X(u - F(u)) = u \tag{18}$$

where $X = R_+^3, P_X(u) = [(u_1)^+, \ldots, (u_3)^+]^T$, and $(u_i)^+ = \max\{0, u_i\}$ for $i = 1, 2, 3$. We use the projection neural network in (2) to solve the above NCP. It can be seen that the existing results [27], [28] cannot ascertain the exponential stability of the projection neural network in (2) though $\nabla F(u)$ is uniformly positive definite in X. However, from Theorem 4 it follows that the neural network in (2) is globally exponentially stable. All simulation results show that the corresponding neural network in (2) is always exponentially stable at u^*. For example, Fig. 6 displays the trajectory of (2) with 20 random initial points, where $\Lambda = 4I$.

The final example will illustrate the result of Theorem 5.

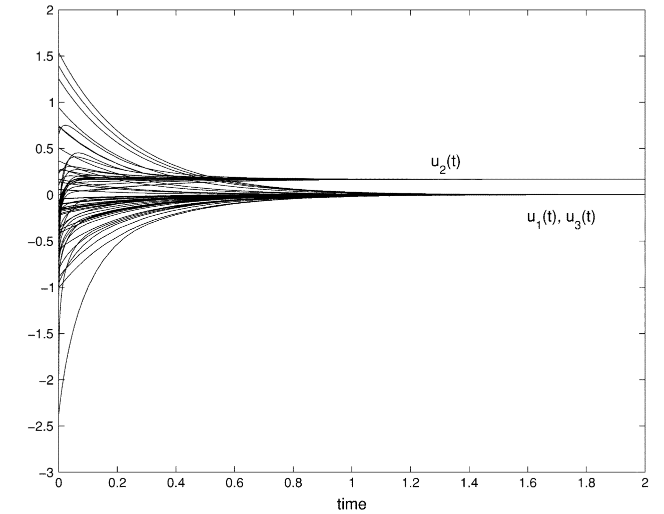


Fig. 6. Global Exponential stability of the projection neural network in (2) for solving the NCP in Example 4.

Example 5: Consider the general linear-quadratic optimization problem (GLQP) [32]

$$\begin{aligned} \min_{x\geq 0}\max_{y\geq 0} \quad f(x,y) =& q^T x - p^T y + \frac{1}{2}x^T \\ & Ax - x^T H y - \frac{1}{2} y^T Q y \\ \text{subject to} \quad 0 \leq Bx \leq& b, 0 \leq Cy \leq d \end{aligned} \tag{19}$$

where

$$A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \quad H = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

$$Q = \begin{pmatrix} 2 & -1 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix},$$

$$B = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 1 & -1 \\ -1 & 1 & -1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 2 & -2 \end{pmatrix}$$

and

$$q = \begin{pmatrix} -4 \\ -3 \\ -2 \end{pmatrix}, \quad p = \begin{pmatrix} -0.5 \\ -1.5 \\ 0.5 \\ -0.5 \end{pmatrix}, \quad b = \begin{pmatrix} 2 \\ 1 \\ 5 \end{pmatrix}, \quad d = \begin{pmatrix} 1 \\ 5 \\ 5 \\ 1 \end{pmatrix}.$$

This problem has an optimal solution $(x^*, y^*) = (0.5, 1.5, 1, 0, 0, 0)$. According to the well-known saddle point Theorem [1], it can be seen that the above GLQP can be converted into an general linear variational inequality (GLCP): find $z^* \in X$ such that

$$(z - Nz^*)^T (Mz^* + l) \geq 0, \quad \forall z \in X$$

where $z = (x, y, s, w) \in R^3 \times R^4 \times R^3 \times R^4, z^* = (x^*, y^*, s^*, w^*)^T$

$$X = \{z = (x, y, s, w) \in R^{14} \,|\, x \geq 0, y \geq 0, 0 \leq s \leq b, \\ 0 \leq w \leq d\},$$

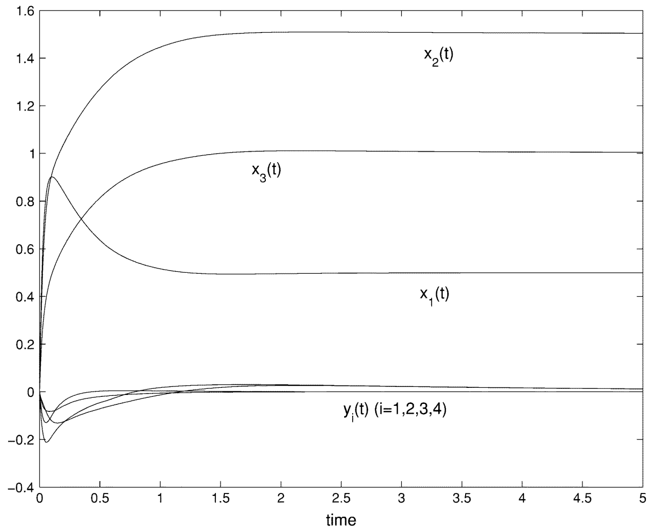


Fig. 7. Global convergence of the neural network in (20) for solving GLQP in Example 5.

and

$$N=\begin{pmatrix} I_1 & 0 & 0 & 0\\ 0 & I_2 & 0 & 0\\ B & 0 & 0 & 0\\ 0 & C & 0 & 0\end{pmatrix},\quad M=\begin{pmatrix} A & -H & B^T & 0\\ H^T & Q & 0 & C^T\\ 0 & 0 & -I_1 & 0\\ 0 & 0 & 0 & -I_2\end{pmatrix},$$

$$l=\begin{pmatrix} q\\ -p\\ 0\\ 0\end{pmatrix}.$$

The existing primal-dual neural network in (3) can not be applied to solve the GLCP. However, the proposed neural network in (12) can be applied to solve the above GLCP, and it becomes

$$\frac{dz}{dt}=Q\{P_X(Nz-Mz-l)-Nz\} \tag{20}$$

where $Q=B(M^T+N^T)$. All simulation results show the trajectory of (20) is always globally convergent to z^*. For example, let $B=2I$ and let the initial point be zero. A solution to GLCP is obtained as follows:

$$(x^1,y^1)=(0.499,1.504,1.004,0.012,0.012,0,0)$$

where $t=5$. Fig. 7 shows the transient behavior of $(x(t),y(t))$ based on (20) with an random initial point.

V. Conclusion

In this paper, we have proposed a general projection neural network for real-time solutions of such problems. The general projection neural network has a simple structure and low complexity for implementation and includes several existing neural networks for optimization, such as the primal-dual neural networks, the dual neural network, the projection neural network, as special cases. Moreover, its equilibrium points are able to solve a wide variety of optimization and related problems. Under mild conditions, we have shown the general projection neural network has properties of global convergence, global asymptotic stability, and a global exponential stability. Since the general projection neural network contains several existing neural networks as special cases, the obtained stability results naturally generalize the existing ones for special cases of neural networks. Furthermore, we have obtained several improved stability results on two special cases of the general projection neural network under weaker conditions. The obtained results are helpful for wide applications of the the general projection neural network. Illustrative examples with applications to optimization and related problems show that the proposed neural network is effective in solving these problems. Further investigations will be aimed at the improvement of the stability conditions and engineering applications of the general projection neural network to robot motion control and signal processing, etc.

References

[1] D. P. Bertsekas, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[2] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, 2nd ed. New York: Wiley, 1993.

[3] T. Yoshikawa, *Foundations of Robotics: Analysis and Control*. Cambridge, MA: MIT Press, 1990.

[4] B. Kosko, *Neural Networks for Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[5] D. W. Tank and J. J. Hopfield, "Simple neural optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit," *IEEE Trans. Circuits Syst. II*, vol. 33, pp. 533–541, May 1986.

[6] A. Cichocki and R. Unbehauen, "Switched-capacitor artificial neural networks for differential optimization," *J. Circuit Theory Applicat.*, vol. 19, pp. 161–187, 1991.

[7] G. L. Dempsey and E. S. McVey, "Circuit implementation of a peak detector neural network," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 585–591, Sept. 1993.

[8] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*. New York: Wiley, 1993.

[9] M. P. Kennedy and L. O. Chua, "Neural networks for nonlinear programming," *IEEE Trans. Circuits Syst. II*, vol. 35, pp. 554–562, May 1988.

[10] A. Rodríguez-Vázquez, R. Domínguez-Castro, A. Rueda, J. L. Huertas, and E. Sánchez-Sinencio, "Nonlinear switched-capacitor 'neural networks' for optimization problems," *IEEE Trans. Circuits Syst. II*, vol. 37, pp. 384–397, Mar. 1990.

[11] S. Zhang, X. Zhu, and L.-H. Zou, "Second-order neural networks for constrained optimization," *IEEE Trans. Neural Networks*, vol. 3, pp. 1021–1024, Nov. 1992.

[12] A. Bouzerdoum and T. R. Pattison, "Neural network for quadratic optimization with bound constraints," *IEEE Trans. Neural Networks*, vol. 4, pp. 293–304, Mar. 1993.

[13] Q. Tao, J. D. Cao, M. S. Xue, and H. Qiao, "A high performance neural network for solving nonlinear programming problems with hybrid constraints," *Phys. Lett. A*, vol. 288, no. 2, pp. 88–94, 2001.

[14] Y. S. Xia, "A new neural network for solving linear and quadratic programming problems," *IEEE Trans. Neural Networks*, vol. 7, pp. 1544–1547, July 1996.

[15] ——, "A new neural network for solving linear programming problems and its applications," *IEEE Trans. Neural Networks*, vol. 7, pp. 525–529, July 1996.

[16] Y. S. Xia and J. Wang, "A dual neural network for kinematic control of redundant robot manipulators," *IEEE Trans. Syst., Man Cybern. B*, vol. 31, pp. 147–154, Feb. 2001.

[17] Y. S. Xia, H. Leung, and J. Wang, "A projection neural network and its application to constrained optimization problems," *IEEE Trans. Circuits Syst. II*, vol. 49, pp. 447–458, Apr. 2002.

[18] C. Y. Maa and M. A. Shanblatt, "Linear and quadratic programming neural network analysis," *IEEE Trans. Neural Networks*, vol. 3, pp. 580–594, Nov. 1992.

[19] W. E. Lillo, M. H. Loh, S. Hui, and S. H. Žak, "On solving constrained optimization problems with neural networks: A penalty method approach," *IEEE Trans. Neural Networks*, vol. 4, pp. 931–939, Nov. 1993.

[20] M. A. Noor, "General nonlinear variational inequalities," *J. Math. Anal. Appl.*, vol. 126, no. 1, pp. 78–84, 1987.

[21] J. S. Pang and J. C. Yao, "On a generalization of a normal map and equation," *SIAM J. Control Optim.*, vol. 33, pp. 168–184, 1995.

[22] D. Kinderlehrer and G. Stampcchia, *An Introduction to Variational Inequalities and Their Applications*. New York: Academic, 1980.

[23] M. Fukushima, "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems," *Math. Programming*, vol. 53, pp. 99–110, 1992.

[24] M. C. Ferris and J. S. Pang, "Engineering and economic applications of complementarity problems," in *SIAM Rev.*, vol. 39, 1997, pp. 669–713.

[25] L. Vandenberghe, B. L. De Moor, and J. Vandewalle, "The generalized linear complementary problem applied to the complete analysis of resistive piecewise-linear circuits," *IEEE Trans. Circuits Syst. II*, vol. 36, pp. 1382–1391, Nov. 1989.

[26] R. K. Miller and A. N. Michel, *Ordinary Differential Equations*. New York: Academic, 1980.

[27] Y. S. Xia and J. Wang, "On the stability of globally projected dynamic systems," *J. Optim. Theory Appl.*, vol. 106, no. 1, pp. 129–150, July 2000.

[28] X. B. Liang, "On the analysis of a recurrent neural network for solving nonlinear monotone variational inequality problems," *IEEE Trans. Neural Networks*, vol. 13, pp. 481–486, Mar. 2002.

[29] X. Gao, "Exponential stability of globally projected dynamic systems," *IEEE Trans. Neural Networks*, vol. 14, pp. 426–431, Mar. 2003.

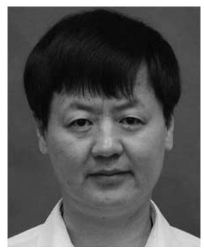
[30] R. Andreani, A. Friedlander, and S. A. Santos, "On the resolution of the generalized nonlinear complementarity problem," *SIAM J. Optim.*, vol. 12, no. 2, pp. 303–321, 2001.

[31] C. Charalambous, "Nonlinear least path optimization and nonlinear programming," *Math. Programming*, vol. 12, pp. 195–225, 1977.

[32] R. T. Rockafellar and R. J. B. Wets, "Linear-quadratic programming and optimal control," *SIAM J. Control Optim.*, vol. 25, pp. 781–814, 1987.

Youshen Xia (M'96–SM'01) received the B.S. and M.S. degrees both in computational mathematics and applied software from Nanjing University, Nanjing, China, in 1982 and 1989, respectively, and the Ph.D. degree from the Department of Automation and Computer-Aided Engineering, The Chinese University of Hong Kong, China, in 2000.

He was an Associate Professor in the Department of Applied Mathematics, Nanjing University of Posts and Telecommunications, and a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, University of Calgary, Canada, and in the Department of Automation and Computer-Aided Engineering, The Chinese University of Hong Kong, in 2001 and 2002, respectively. His present research interests include design and analysis of recurrent neural networks for constrained optimization and neural network applications to data mining, data fusion, and signal and image processing.



Jun Wang (S'89–M'90–SM'93) received the B.S. degree in electrical engineering and an M.S. degree in systems engineering from Dalian University of Technology, China, and the Ph.D. degree in systems engineering from Case Western Reserve University, Cleveland, OH.

Prior to 1995, he was an Associate Professor at the University of North Dakota, Grand Forks, ND. Currently, he is a Professor of automation and computer-aided engineering at the Chinese University of Hong Kong. His current research interests include neural networks and their engineering applications.

Dr. Wang is an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B CYBERNETICS, and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C APPLICATIONS AND REVIEWS.