

Research Article

Robust Linear Neural Network for Constrained Quadratic Optimization

Zixin Liu,^{1,2} Yuanan Liu,¹ and Lianglin Xiong³

¹*School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 102209, China*

²*Department of Mathematics and Statistics, Guizhou University of Finance and Economics, Guiyang 550025, China*

³*School of Mathematics and Computer Science, Yunnan Minzu University, Kunming 650500, China*

Correspondence should be addressed to Zixin Liu; xinxin905@163.com

Received 11 April 2017; Revised 21 July 2017; Accepted 26 July 2017; Published 28 August 2017

Academic Editor: Manuel De la Sen

Copyright © 2017 Zixin Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Based on the feature of projection operator under box constraint, by using convex analysis method, this paper proposed three robust linear systems to solve a class of quadratic optimization problems. Utilizing linear matrix inequality (LMI) technique, eigenvalue perturbation theory, Lyapunov-Razumikhin method, and LaSalle's invariance principle, some stable criteria for the related models are also established. Compared with previous criteria derived in the literature cited herein, the stable criteria established in this paper are less conservative and more practicable. Finally, a numerical simulation example and an application example in compressed sensing problem are also given to illustrate the validity of the criteria established in this paper.

1. Introduction

Quadratic optimization problem is a simple but very important and basic problem in convex optimization theory. It is widely applied in many scientific and engineering applications, such as regression analysis, data fusion, system identification, filter design, and compressed sensing [1–5]. Among these applications, the real-time solutions of such quadratic optimization problems are often required. In order to solve quadratic optimization problem, many different algorithms are provided in previous reference, such as proximal point algorithm (PPA), extended PPA, and splitting methods [6–8]. However, in many practical optimization problems, the numbers of decision variables and constraints are usually very large. When a large-scale quadratic optimization problem has to be performed in practical problem, computation complexity becomes more challenging. For such applications, classical optimization techniques may not be competent due to the problem dimension and stringent requirement on computational time [9]. One promising method for solving these problems is to employ artificial recurrent neural networks, since neural network has parallel computing capacity [10].

Mathematically, the optimization problem to be solved is mapped into a dynamical system so that its state output can give the optimal solution and the optimal solution is then obtained by tracking the state trajectory of the designed dynamical system based on the numerical ordinary differential equation technique [11]. From the pioneering work of McCulloch and Pittes, numerous neural network models have been developed. Compared with conventional numerical optimization algorithm, neural network has a low model complexity and parallel computing capacity, it is more suitable for engineering applications, and it has a weaker global convergence condition.

In the recent decades, all kinds of different neural network models were established to solve variant constrained optimization problems. These optimization problems include game theory, linear programming problems, linear complement problems, projection equation problems, variational inequality problems, nonlinear optimization problems, general convex optimization problems, nonconvex optimization problems, and nonsmooth optimization problems.

Neural network for solving linear programming problem perhaps stemmed back from Pyne's work [12] and Tank and

Hopfield's work [13]. Their seminal work has inspired other researchers to develop recurrent neural networks for nonlinear optimization. Zhang and Constantinides derived a Lagrangian neural network for solving nonlinear convex optimization problems with linear equality constraints in [14]. In [15], Zhang researched the exponential stability of quadratic optimization problem on neural network and established a discrete-time neural network model to solve quadratic optimization problem with convex constraint only. Tan et al. studied the global exponential stability of discrete-time neural network for constrained quadratic optimization problem in [16]. In order to solve more general optimization problem, Yashtini and Malek investigated a discrete-time neural network model for solving nonlinear convex problems with hybrid constraints in [17]. Bouzerdoum and Pattison presented a neural network for solving quadratic convex optimization problems with bounded variables in [18]. Xia proposed primal-dual neural networks for solving linear and quadratic programming problems in [19], researched a dual neural network for solving strictly convex quadratic programming problems in [20], and proposed a Bi-projection neural network for solving constrained quadratic optimization problems in [21]. To solve quadratic minimax optimization problems, Liu and Wang [22] proposed a projection neural network (PNN) for constrained quadratic minimax optimization problem. To solve nonsmooth optimization problems, Liu and Wang [23] proposed a one-layer PNN for solving a class of pseudoconvex and nonsmooth nonlinear optimization problems.

It is worth pointing out that most of above established neural network models are nonlinear forms. And the stability criteria derived in the literature are based on Lyapunov stable theory. However, when the constrained conditions of quadratic optimization are box constrained, nonlinear projection operator satisfies section constraint; in this case, nonlinear projection operator can be expressed by a linear form with uncertain term. This means that nonlinear projection neural network can be rewritten as a robust linear system. Thus, utilizing eigenvalue perturbation theory, we can give out some new system stable criteria. This idea inspires this work. In this paper, using convex analysis tool, we first establish some new robust linear neural network for box constrained quadratic optimization; then, by using LMI technique and eigenvalue perturbation theory, some exponentially stable criteria are established. When time delays are considered, by using Lyapunov-Razumikhin method and LaSalle's invariance principle, we further derive some asymptotical stability criteria for the established time-delayed linear robust neural network. To illustrate the efficiencies and validity of the derived stability criteria in this paper, a numerical example and an application example in compressed sensing problem are also given.

The remainder of this paper is organized as follows. In Section 2, a constrained quadratic optimization problem and related neural network models are described. In Section 3, the global stability and convergence of the proposed neural network are analyzed. In Section 4, a simulation numerical example and an application to compressed sensing are given. Finally, conclusions are drawn in Section 5.

2. Problem and Neural Network Model

Consider the following constrained quadratic optimization problem:

$$\min E(x) = \min \left\{ \frac{1}{2} x^T M x + b^T x \mid x \in \Omega \right\}, \quad (1)$$

where $x = (x_1, x_2, \dots, x_n)^T \in \mathcal{R}^n$, M is a symmetric nonnegative definite matrix, b is a vector, and the superscript T denotes the transpose operator. Ω is defined by $\Omega = \{x \in \mathcal{R}^n \mid x_i \in [c_i, d_i], i = 1, 2, \dots, n\}$, where c_i, d_i are constants such that $c_i \leq d_i, i = 1, 2, \dots, n$. It is seen that (1) can contain the constrained least square problem as a special case. Many solution methods for solving (1) were presented, including neural networks and numerical optimization algorithms. They all have a solution space being greater than the solution space of (1). As a result, when n is large, these solution methods will have a slow convergence rate. However, fast computation of such large optimization problems is often required in practice. Due to low model complexity and parallel computing capacity, neural networks algorithm becomes a popular method to solve problem (1). In what follows, by using projection theory and equivalent transform, some neural network models for solving problem (1) will be first introduced.

2.1. Neural Network without Time Delays. As is well known, without considering time delays, the optimal solution x^* for problem (1) is equivalent to the equilibrium point of the following project neural network:

$$\begin{aligned} \frac{dx(t)}{dt} &= -x(t) + P_\Omega(x(t) - \rho M x(t) - \rho b), \quad t > t_0 \\ x(t_0) &= x_0, \quad t \leq t_0, \end{aligned} \quad (2)$$

where $\rho > 0$ is an arbitrary constant and $P_\Omega(\cdot)$ is a projection operator defined by

$$P_\Omega(x) = \arg \min_{y \in \Omega} \|x - y\|_2, \quad (3)$$

where $\|\cdot\|_2$ denotes l_2 norm of \mathcal{R}^n . Denote $P_\Omega(x) = (P_\Omega(x_1), P_\Omega(x_2), \dots, P_\Omega(x_n))^T$; since $\Omega = \{x \in \mathcal{R}^n \mid x_i \in [c_i, d_i], i = 1, 2, \dots, n\}$, from [2], the display expression of $P_\Omega(x_i)$ is as follows:

$$P_\Omega(x_i) = \begin{cases} c_i, & x_i < c_i \\ x_i, & c_i \leq x_i \leq d_i \\ d_i, & x_i > d_i. \end{cases} \quad (4)$$

Set $u(t) = x(t) - \rho M x(t) - \rho b$ and substitute $u(t)$ into (2); it yields that

$$\begin{aligned} \frac{du(t)}{dt} &= (I - \rho M) \frac{dx(t)}{dt} \\ &= -(I - \rho M) x(t) + (I - \rho M) P_\Omega(u(t)) \\ &= -u(t) + (I - \rho M) P_\Omega(u(t)) - \rho b. \end{aligned} \quad (5)$$

Denote u^* as an equilibrium point of system (5), and let $y(t) = u(t) - u^*$; system (5) can be rewritten in the following form:

$$\begin{aligned} \frac{dy(t)}{dt} &= -y(t) \\ &+ (I - \rho M) [P_{\Omega}(y(t) + u^*) - P_{\Omega}(u^*)] \quad (6) \\ &= -y(t) + (I - \rho M) \bar{P}_{\Omega}(y(t)), \end{aligned}$$

where $\bar{P}_{\Omega}(y(t)) = P_{\Omega}(y(t) + u^*) - P_{\Omega}(u^*)$. Notice the expression of $P_{\Omega}(\cdot)$ in (4); one can obtain that

$$0 \leq \frac{P_{\Omega}(u) - P_{\Omega}(v)}{u - v} \leq 1, \quad \forall u \neq v \in \mathcal{R}. \quad (7)$$

Denote $y(t) = (y_1(t), y_2(t), \dots, y_n(t))^T$; since $\bar{P}_{\Omega}(y(t)) = P_{\Omega}(y(t) + u^*) - P_{\Omega}(u^*)$, it follows that

$$0 \leq \frac{\bar{P}_{\Omega}(y_i(t))}{y_i(t)} \leq 1, \quad \forall u \neq v \in \mathcal{R}. \quad (8)$$

Set

$$\lambda_i(y_i(t)) = \frac{\bar{P}_{\Omega}(y_i(t))}{y_i(t)}, \quad (9)$$

$$\Delta(t) = \text{diag}(\lambda_1(y_1(t)), \lambda_2(y_2(t)), \dots, \lambda_n(y_n(t)));$$

it follows that $0 \leq \lambda_i(y_i(t)) \leq 1$, $\Delta(t)^T \Delta(t) \leq I$.

$$\begin{aligned} \bar{P}_{\Omega}(y_i(t)) \\ = [\lambda_i(y_i(t)) \cdot 1 + (1 - \lambda_i(y_i(t))) \cdot 0] y_i(t). \end{aligned} \quad (10)$$

And system (6) can be rewritten in the following linear robust neural network model form:

$$\frac{dy(t)}{dt} = [-I + (I - \rho M) \Delta(t)] y(t). \quad (11)$$

2.2. Neural Network with Time Delays

2.2.1. *Time-Delayed Neural Network Type I.* When the time delay is taken into account, a delayed projection neural network, which can be regarded as an improvement form for model (2), can be suggested for solving (1) as follows:

$$\begin{aligned} \frac{dx(t)}{dt} &= -x(t) + P_{\Omega}(x(t - \tau) - \rho Mx(t - \tau) - \rho b), \\ &t > t_0, \quad (12) \end{aligned}$$

$$x(t_0) = x_0, \quad t \leq t_0,$$

where time delay $\tau > 0$ is a constant. Set $u(t - \tau) = x(t - \tau) - \rho Mx(t - \tau) - \rho b$, and substitute $u(t - \tau)$ into (12); it yields that

$$\begin{aligned} \frac{du(t)}{dt} &= (I - \rho M) \frac{dx(t)}{dt} \\ &= -(I - \rho M)x(t) + (I - \rho M)P_{\Omega}(u(t - \tau)) \quad (13) \\ &= -u(t) + (I - \rho M)P_{\Omega}(u(t - \tau)) - \rho b. \end{aligned}$$

Denote u^* as an equilibrium point of system (13), and let $y(t) = u(t) - u^*$; system (13) can be rewritten in the following form:

$$\begin{aligned} \frac{dy(t)}{dt} &= -y(t) \\ &+ (I - \rho M) [P_{\Omega}(y(t - \tau) + u^*) - P_{\Omega}(u^*)] \quad (14) \\ &= -y(t) + (I - \rho M) \hat{P}_{\Omega}(y(t - \tau)), \end{aligned}$$

where $\hat{P}_{\Omega}(y(t - \tau)) = P_{\Omega}(y(t - \tau) + u^*) - P_{\Omega}(u^*)$. Notice the expression of $P_{\Omega}(\cdot)$ in (4); one can obtain that

$$0 \leq \frac{P_{\Omega}(u) - P_{\Omega}(v)}{u - v} \leq 1, \quad \forall u \neq v \in \mathcal{R}. \quad (15)$$

Denote $y(t - \tau) = (y_1(t - \tau), y_2(t - \tau), \dots, y_n(t - \tau))^T$; since $\hat{P}_{\Omega}(y(t - \tau)) = P_{\Omega}(y(t - \tau) + u^*) - P_{\Omega}(u^*)$, it follows that

$$0 \leq \frac{\hat{P}_{\Omega}(y_i(t - \tau))}{y_i(t - \tau)} \leq 1, \quad \forall u \neq v \in \mathcal{R}. \quad (16)$$

Set

$$\lambda_i(y_i(t - \tau)) = \frac{\hat{P}_{\Omega}(y_i(t - \tau))}{y_i(t - \tau)}, \quad (17)$$

$$\begin{aligned} \hat{\Delta}(t) &= \text{diag}(\lambda_1(y_1(t - \tau)), \lambda_2(y_2(t - \tau)), \dots, \\ &\lambda_n(y_n(t - \tau))); \end{aligned}$$

it follows that $0 \leq \lambda_i(y_i(t - \tau)) \leq 1$, $\hat{\Delta}(t)^T \hat{\Delta}(t) \leq I$.

$$\begin{aligned} \hat{P}_{\Omega}(y_i(t - \tau)) \\ = [\lambda_i(y_i(t - \tau)) \cdot 1 + (1 - \lambda_i(y_i(t - \tau))) \cdot 0] \\ \cdot y_i(t - \tau). \end{aligned} \quad (18)$$

And system (14) can be rewritten in the following time-delayed linear robust neural network form:

$$\frac{dy(t)}{dt} = -y(t) + (I - \rho M) \hat{\Delta}(t) y(t - \tau). \quad (19)$$

2.2.2. *Time-Delayed Neural Network Type II.* Another time-delayed neural network, which can be regarded as another improvement form for model (2), can be suggested for solving (1) as follows:

$$\begin{aligned} \frac{dx(t)}{dt} &= P_{\Omega}(x(t - \tau) - \rho Mx(t - \tau) - \rho b) \\ &- x(t - \tau), \quad t > t_0, \quad (20) \end{aligned}$$

$$x(t_0) = x_0, \quad t \leq t_0,$$

where time delay $\tau > 0$ is a constant. Set $u(t - \tau) = x(t - \tau) - \rho Mx(t - \tau) - \rho b$, and substitute $u(t - \tau)$ into (20); similar to the technique used above, system (20) can be transformed into the following equivalent time-delayed linear robust neural network form:

$$\frac{dy(t)}{dt} = [-I + (I - \rho M) \hat{\Delta}(t)] y(t - \tau). \quad (21)$$

3. Stability Analysis

Since systems (11), (19), and (21) are all linear differential equations with uncertain term, the stability criteria for these systems can be derived by using eigenvalue perturbation theory, Lyapunov-Razumikhin method, and LMI technique. To derive the stability criteria, we first introduce the following lemma.

Lemma 1 (see [24]). *Suppose that $\Delta(t) = [I - F(t)J]^{-1}F(t)$, where J is a constant matrix satisfying $I - JJ^T > 0$, and $F(t)$ is an uncertain matrix satisfying $F^T(t)F(t) \leq I$, $M^T = M$, S , and N of appropriate dimensions; the inequality*

$$M + S\Delta(t)N + N^T\Delta^T(t)S^T < 0 \quad (22)$$

holds if and only if, for some $\delta > 0$,

$$\begin{bmatrix} \delta M & S & \delta N^T \\ S^T & -I & J^T \\ \delta N & J & -I \end{bmatrix} < 0. \quad (23)$$

By using Lemma 1, the following stability criterion can be derived.

Theorem 2. *The equilibrium point y^* of system (2) is exponentially stable if there exists a positive constant δ such that the following linear matrix inequality holds:*

$$\begin{bmatrix} -2\delta I & I - \rho M & \delta I \\ I - \rho M & -I & 0 \\ \delta I & 0 & -I \end{bmatrix} < 0, \quad (24)$$

and the exponential convergence rate is $\min_i\{\mu_i\}$, where μ_i is the eigenvalue of matrix $B = (A + A^H)/2$, $A = -I + (I - \rho M)\Delta(t)$.

Proof. From Section 2.1, it follows that the equilibrium point's exponential stability of system (2) is equivalent to the exponential stability of the trivial solution of system (11). Notice that system (11) is a linear system structure; thus the exponential stability of the trivial solution of system (11) is equivalent to $\text{Re}(\lambda) < 0$, where λ is an arbitrary eigenvalue of matrix $-I + (I - \rho M)\Delta(t)$.

Set $B = (b)_{n \times n} = (A + A^H)/2$, $A = -I + (I - \rho M)\Delta(t)$. Denote $\{\mu_1, \mu_2, \dots, \mu_n\}$ as the eigenvalues of matrix B satisfying $\mu_1 \geq \mu_2 \geq \dots \geq \mu_n$ and $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ as the eigenvalues of matrix A satisfying $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. Let $X \neq 0$ be an identity eigenvector belonging to eigenvalue λ_i ; namely,

$$\begin{aligned} AX &= \lambda_i X, \\ \|X\|^2 &= X^H X = 1. \end{aligned} \quad (25)$$

It follows that

$$\begin{aligned} \langle X, AX \rangle &= \langle X, \lambda_i X \rangle = \lambda_i \langle X, X \rangle = \lambda_i, \\ X^H AX &= \lambda_i, \\ X^H A^H X &= \bar{\lambda}_i. \end{aligned} \quad (26)$$

Thus

$$\text{Re}(\lambda_i) = \left\langle X, \frac{(A + A^H)X}{2} \right\rangle = \langle X, BX \rangle. \quad (27)$$

Since B is a normal matrix, there exists a unitary matrix U such that

$$U^H B U = \text{diag}(\mu_1, \mu_2, \dots, \mu_n) \equiv D; \quad (28)$$

this means that

$$\text{Re}(\lambda_i) = \langle X, BX \rangle = \langle X, U D U^H X \rangle = X^H U D U^H X. \quad (29)$$

Set $Y = U^H X$; one can obtain that

$$Y^H Y = \sum_{i=1}^n |y_i|^2 = X^H U U^H X = X^H X = 1; \quad (30)$$

thus

$$\text{Re}(\lambda_i) = Y^H D Y = \sum_{i=1}^n \mu_i |y_i|^2. \quad (31)$$

Notice that $\mu_1 \geq \mu_2 \geq \dots \geq \mu_n$; it yields

$$\mu_n = \sum_{i=1}^n \mu_n |y_i|^2 \leq \sum_{i=1}^n \mu_i |y_i|^2 \leq \sum_{i=1}^n \mu_1 |y_i|^2 = \mu_1, \quad (32)$$

which means that $\mu_n \leq \text{Re}(\lambda_i) \leq \mu_1$. Obviously, if $\mu_i < 0$ ($i = 1, 2, \dots, n$) it means that $\text{Re}(\lambda_i) < 0$. Because μ_i is eigenvalue of matrix B and matrix B is a symmetric real matrix, this implies that if matrix $B < 0$, then $\text{Re}(\lambda_i) < 0$. On the other hand, $B < 0$ is equivalent to

$$-2I + (I - \rho M)\Delta(t)I + I\Delta(t)(I - \rho M)^T < 0. \quad (33)$$

Since $\Delta^T(t)\Delta(t) \leq I$, by Lemma 1, if there is a positive constant δ such that

$$\begin{bmatrix} -2\delta I & I - \rho M & \delta I \\ I - \rho M & -I & 0 \\ \delta I & 0 & -I \end{bmatrix} < 0, \quad (34)$$

then $\text{Re}(\lambda_i) < 0$, $i = 1, 2, \dots, n$, which means that the trivial solution is exponentially stable; this completes the proof. \square

Remark 3. Obviously, the stability condition established in Theorem 2 is a LMI form. Using Matlab LMI tool box, it can be easily solved. However, when a large-scale quadratic optimization problem has to be performed, the computation complexity becomes challenging. In order to overcome this flaw, by using eigenvalue perturbation theory, another more simple and practical result can be derived as follows. Before continuing, the following lemma is needed.

Lemma 4 (see [25]). *Let $A = P\Lambda P^{-1} \in C^{n \times n}$, $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_n\}$; $\delta \in C^{n \times n}$; $\mu_1, \mu_2, \dots, \mu_n$ denotes the eigenvalues of matrix $A + \delta$; then for arbitrary μ_j , there exists an eigenvalue λ_i such that*

$$|\lambda_i - \mu_j| \leq \|P^{-1}\delta P\|_{\infty}, \quad (35)$$

where $\|P^{-1}\delta P\|_{\infty} \equiv \max_i \sum_{j=1}^n |p_{ij}|$ and $P^{-1}\delta P \equiv (p_{ij})_{n \times n}$.

Theorem 5. *The equilibrium point y^* of system (2) is exponentially stable if $\|I - \rho M\|_\infty < 1$, and the exponential convergence rate is $\min_i \{\mu_i\}$.*

Proof. From the proof of Theorem 2, it follows that the equilibrium point's exponential stability of system (2) is equivalent to $\text{Re}(\lambda) < 0$, where λ is an arbitrary eigenvalue of matrix $-I + (I - \rho M)\Delta(t)$. Let μ_i be an arbitrary eigenvalue of matrix $-I + (I - \rho M)\Delta(t)$; by Lemma 4, it yields that

$$\begin{aligned} |-1 - \mu_i| &\leq \|(I - \rho M)\Delta(t)\|_\infty \\ &\leq \|I - \rho M\|_\infty \cdot \|\Delta(t)\|_\infty. \end{aligned} \quad (36)$$

By the definition of $\Delta(t)$, we have

$$\begin{aligned} \|\Delta(t)\|_\infty &= \|\text{diag}(\lambda_1(y_1(t)), \lambda_2(y_2(t)), \dots, \lambda_n(y_n(t)))\|_\infty \\ &= \max_i \sum_{j=1}^n |\Delta(t)_{ij}| = \max_i \{|\lambda_i(y_i(t))|\} \leq 1. \end{aligned} \quad (37)$$

From (36) and (37), it yields that

$$\begin{aligned} -1 - \|(I - \rho M)\|_\infty &\leq \mu_i \leq \|I - \rho M\|_\infty - 1, \\ &i = 1, 2, \dots, n. \end{aligned} \quad (38)$$

Obviously, $\|I - \rho M\|_\infty < 1$ means that arbitrary eigenvalue λ of matrix $-I + (I - \rho M)\Delta(t)$ is negative; this means that the equilibrium point of system (2) is exponentially stable, which completes the proof. \square

Remark 6. Usually, matrix infinite norm is larger than matrix 2-norm; thus the result established in Theorem 5 can be rewritten by matrix 2-norm form further. And notice the special property of identity matrix I ; by using improved eigenvalue perturbation theory, another exponentially stable criterion can be derived as follows. Before continuing, the following lemma is needed.

Lemma 7 (see [26]). *Let $A \in C^{n \times n}$ be normal matrix, $\lambda_1, \lambda_2, \dots, \lambda_n$ are eigenvalues of matrix A , $\delta \in C^{n \times n}$ is an arbitrary matrix, and μ denotes the eigenvalue of matrix $A + \delta$; then there exists an eigenvalue λ_i of matrix A such that*

$$|\mu - \lambda_i| \leq \|\delta\|_2, \quad (39)$$

where $\|\cdot\|_2$ denotes matrix 2-norm.

Theorem 8. *The equilibrium point y^* of system (2) is exponentially stable if $\|I - \rho M\|_2 < 1$, and the exponential convergence rate is $\min_i \{\mu_i\}$.*

Proof. From the proof of Theorem 2, it follows that the equilibrium point's exponential stability of system (2) is equivalent to $\text{Re}(\lambda) < 0$, where λ is an arbitrary eigenvalue of matrix $-I + (I - \rho M)\Delta(t)$. Let μ_i be an arbitrary eigenvalue of matrix $-I + (I - \rho M)\Delta(t)$; since $-I$ is a normal matrix and all

of the eigenvalues of matrix $-I$ are -1 , by Lemma 7, it yields that

$$|\mu_i + 1| \leq \|(I - \rho M)\Delta(t)\|_2 \leq \|I - \rho M\|_2 \cdot \|\Delta(t)\|_2. \quad (40)$$

By the definition of $\Delta(t)$, we have

$$\begin{aligned} \|\Delta(t)\|_2 &= \|\text{diag}(\lambda_1(y_1(t)), \lambda_2(y_2(t)), \dots, \lambda_n(y_n(t)))\|_2 \\ &= \sqrt{\max r(\Delta(t)^T \Delta(t))} = \max_i \{|\lambda_i(y_i(t))|\} \leq 1. \end{aligned} \quad (41)$$

From (40) and (41), it yields that

$$\begin{aligned} -1 - \|(I - \rho M)\|_2 &\leq \mu_i \leq \|I - \rho M\|_2 - 1, \\ &i = 1, 2, \dots, n. \end{aligned} \quad (42)$$

Obviously, $\|I - \rho M\|_2 < 1$ means that arbitrary eigenvalue λ of matrix $-I + (I - \rho M)\Delta(t)$ is negative; this means that the equilibrium point of system (2) is exponentially stable, which completes the proof. \square

Remark 9. As is well known, Lyapunov-Razumikhin method is a powerful stability analysis tool for linear time-delayed system. When time delays are considered, by using Lyapunov-Razumikhin method, a similar asymptotically stable result with Theorem 8 for systems (12) and (19) can be derived as follows. Before continuing, the following Razumikhin condition is needed.

Lemma 10 (see [27] (Razumikhin condition)). *The equilibrium point y^* of system (12) is asymptotically stable if there exists a positive definite Lyapunov function $V(t, y(t))$ satisfying*

$$\begin{aligned} V(t+s, y(t+s)) &\leq V(t, y(t)), \quad s \in [-\tau, 0] \\ &\implies \frac{d}{dt} V(t, y(t)) < 0. \end{aligned} \quad (43)$$

By using Razumikhin condition, the following asymptotically stable criterion for systems (12) and (19) can be obtained as follows.

Theorem 11. *The equilibrium point y^* of system (12) is asymptotically stable if $\|I - \rho M\|_2 < 1$.*

Proof. Construct a positive definite Lyapunov function $V(t, y(t)) = y(t)^T y(t)$; it follows that

$$\begin{aligned} V(t+s, y(t+s)) &\leq V(t, y(t)) \iff \\ &\|y(t+s)\|_2 \leq \|y(t)\|_2. \end{aligned} \quad (44)$$

When $V(t+s, y(t+s)) \leq V(t, y(t))$, $s \in [-\tau, 0]$, it yields $\|y(t+s)\|_2 \leq \|y(t)\|_2$, and the derivatives of Lyapunov function $V(t, y(t))$ along the trajectory of system (19) satisfy

$$\begin{aligned} \frac{dV(t, y(t))}{dt} &= 2y^T(t) [-y(t) + (I - \rho M) \widehat{\Delta}(t) y(t - \tau)] \\ &= 2y^T(t) (I - \rho M) \widehat{\Delta}(t) y(t - \tau) - 2\|y(t)\|_2^2 \\ &\leq 2\|y^T(t) (I - \rho M) \widehat{\Delta}(t) y(t - \tau)\|_2 - 2\|y(t)\|_2^2 \\ &\leq 2\|y^T(t)\|_2 \cdot \|(I - \rho M)\|_2 \|\widehat{\Delta}(t)\|_2 \cdot \|y(t - \tau)\|_2 \\ &\quad - 2\|y(t)\|_2^2. \end{aligned} \quad (45)$$

By the definition of $\widehat{\Delta}(t)$, we have

$$\begin{aligned} \|\widehat{\Delta}(t)\|_2 &= \|\text{diag}(\lambda_1(y_1(t - \tau)), \dots, \lambda_n(y_n(t - \tau)))\|_2 \\ &= \sqrt{\max r(\widehat{\Delta}^T(t) \widehat{\Delta}(t))} = \max_i \{|\lambda_i(y_i(t - \tau))|\} \\ &\leq 1. \end{aligned} \quad (46)$$

From (45), (46), and Lemma 10, we have

$$\begin{aligned} \frac{dV(t, y(t))}{dt} &\leq 2\|y^T(t)\|_2 \cdot \|(I - \rho M)\|_2 \\ &\quad \cdot \|y(t - \tau)\|_2 - 2\|y(t)\|_2^2 \\ &\leq 2\|y^T(t)\|_2 \cdot \|(I - \rho M)\|_2 \cdot \|y(t)\|_2 \\ &\quad - 2\|y(t)\|_2^2 \\ &= -2(1 - \|(I - \rho M)\|_2) \|y(t)\|_2^2. \end{aligned} \quad (47)$$

Obviously, from Razumikhin condition, if $\|I - \rho M\|_2 < 1$, then the equilibrium point y^* of system (12) is asymptotically stable, which completes the proof. \square

Remark 12. Similar to the proof of Theorem 11, by using Razumikhin condition, a delay-dependent stable result for systems (20) and (21) can be derived as follows.

Theorem 13. *The equilibrium point y^* of system (21) is asymptotically stable if $\|I - \rho M\|_2 + \tau[1 + 2\|I - \rho M\|_2 + \|I - \rho M\|_2^2] < 1$.*

Proof. Construct a positive definite Lyapunov function $V(t, y(t)) = y(t)^T y(t)$; it follows that

$$\begin{aligned} V(t+s, y(t+s)) \leq V(t, y(t)) &\iff \\ \|y(t+s)\|_2 \leq \|y(t)\|_2. & \end{aligned} \quad (48)$$

When $V(t+s, y(t+s)) \leq V(t, y(t))$, $s \in [-\tau, 0]$, it yields $\|y(t+s)\|_2 \leq \|y(t)\|_2$, and the derivatives of Lyapunov function $V(t, y(t))$ along the trajectory of system (21) satisfy

$$\begin{aligned} \frac{dV(t, y(t))}{dt} &= 2y^T(t) [-I + (I - \rho M) \widehat{\Delta}(t)] y(t - \tau) \\ &= 2y^T(t) [-I + (I - \rho M) \widehat{\Delta}(t)] \times \left[y(t) \right. \\ &\quad \left. - (-I + (I - \rho M) \widehat{\Delta}(t)) \int_{t-\tau}^t y(s - \tau) ds \right] \\ &\leq -2\|y(t)\|_2^2 + 2\|y^T(t) (I - \rho M) \widehat{\Delta}(t) y(t)\|_2 \\ &\quad + 2\|y^T(t) [-I + (I - \rho M) \widehat{\Delta}(t)]^2 \\ &\quad \cdot \int_{t-\tau}^t y(s - \tau) ds\|_2 \leq -2\|y(t)\|_2^2 + 2\|(I - \rho M)\|_2 \\ &\quad \cdot \|\widehat{\Delta}(t)\|_2 \|y(t)\|_2^2 + 2\tau \|[-I + (I - \rho M) \widehat{\Delta}(t)]^2\|_2 \\ &\quad \cdot \|y(t)\|_2^2. \end{aligned} \quad (49)$$

Since $\|\widehat{\Delta}(t)\|_2 < 1$, from (49), we have

$$\begin{aligned} \frac{dV(t, y(t))}{dt} &\leq -2\|y(t)\|_2^2 + 2\|(I - \rho M)\|_2 \|y(t)\|_2^2 \\ &\quad + 2\tau \|I - 2[I - \rho M] \widehat{\Delta}(t) \\ &\quad + [I - \rho M] \widehat{\Delta}(t) [I - \rho M] \widehat{\Delta}(t)\|_2 \|y(t)\|_2^2 \\ &\leq -2\|y(t)\|_2^2 + 2\|(I - \rho M)\|_2 \|y(t)\|_2^2 + 2\tau [1 \\ &\quad + 2\|I - \rho M\|_2 + \|I - \rho M\|_2^2] \|y(t)\|_2^2. \end{aligned} \quad (50)$$

Obviously, from Razumikhin condition, if $\|I - \rho M\|_2 + \tau[1 + 2\|I - \rho M\|_2 + \|I - \rho M\|_2^2] < 1$, then the equilibrium point y^* of system (21) is asymptotically stable, which completes the proof. \square

Remark 14. If $\tau = 0$, system (21) degenerates into system (11); Theorem 13 becomes the same as in Theorem 8; thus the criterion derived in Theorem 8 can be regarded as a special case of Theorem 13.

Theorem 15. *If the symbol $<$ in Theorems 2, 5, 8, and 11 becomes \leq , then, for any initial value $x_0 \in \Omega$, the solutions of systems (2) and (12) converge to a related equilibrium point y^* . In particular, these systems are asymptotically stable when their equilibrium set contains exactly one point.*

Proof. Since Ω is a bounded and closed convex set, as is well known, the state vectors of systems (2) and (12) are bounded, and Ω is an invariance set. Utilizing LaSalle's invariance principle (see [28]), similar to the proofs in [20–23], one

can easily obtain the results described in Theorem 15; this complete the proof. \square

Remark 16. Theorems 2, 5, 8, and 11 show that if the inequality signs are strict, then, for any initial x_0 whether it is in feasible set Ω or not, the state vectors are exponentially or asymptotically convergent. When the inequality signs are not strict, Theorem 15 shows that the sufficient condition ensuring system being asymptotically stable is $x_0 \in \Omega$. On the other hand, notice that if ρ is sufficiently small, nonstrict inequality sign in Theorems 2, 5, 8, and 11 can be guaranteed; this means that, for arbitrary initial value $x_0 \in \Omega$ with appropriate dimension, systems (2) and (12) are asymptotically stable.

4. Numerical Simulation

4.1. Numerical Example

Example 1. Consider the quadratic optimization problem (1) with

$$M = \begin{bmatrix} 0.180 & 0.648 & 0.288 \\ 0.648 & 2.880 & 0.720 \\ 0.288 & 0.720 & 0.720 \end{bmatrix}, \quad (51)$$

$$b = [0.4, 0.2, 0.3]^T,$$

and $c = -d = [20, -20, -20]^T$. For quadratic optimization (1), the author in [29] constructed a discrete-time neural network to solve this problem and gave out a globally convergent criterion. In order to further illustrate the globally exponential convergence of the neural networks constructed in [29], Tan et al. gave out some new improvement stable criteria in [16]. In [30], the authors constructed a continuous neural network to solve problem (1) and derived a globally exponential convergent criterion. In order to reduce the conservation of the criterion established in [30], literature [31] further gave out an improved stable result as $\|I - \rho M\|_{m_2} < 1$. If we construct model (11) to solve problem (1), by direct computation, it follows that $\|I - \rho M\|_2 = 0.9984 < 1$; from Theorem 8, it yields that the state vector of system (11) is exponentially convergent to the optimization value of problem (1). Simulation result is illustrated in Figure 1 with initial value $[2, -1, 1.5]^T$, from which one can see that the state vector of system (11) is convergent to the equilibrium point exponentially. Obviously, $\|I - \rho M\|_2 = 0.9984 < \|I - \rho M\|_\infty = 1.0696 < \|I - \rho M\|_{m_2} = 1.5338$; this means that the criterion established in Theorem 8 is less conservative than some earlier results.

If we adopt model (19) to solve problem (1), by direct computing, it follows that $\|I - \rho M\|_2 = 0.9984 < 1$; from Theorem 11, it means that the state vector of system (19) is exponentially stable. Simulation result is illustrated in Figure 2 with $\tau = 4$. If we use model (21) to solve problem (1), from the result established in Theorem 13, one can see that the state vector stable condition is stringently required for time delay. For example, in system (21), let $\tau = 0.01$; simulation

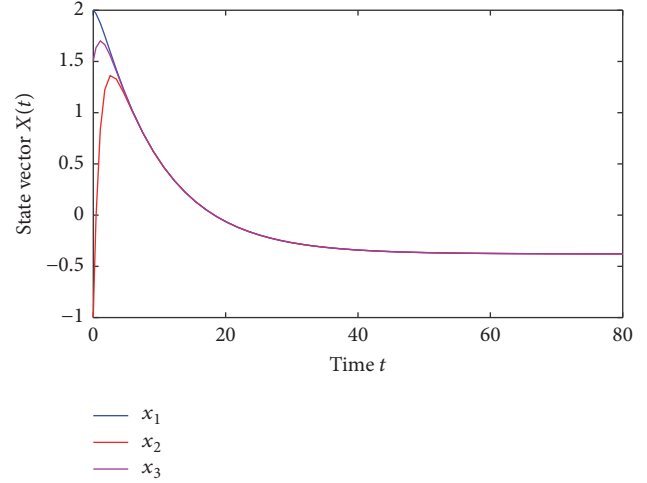


FIGURE 1: State vector of system (2) with parameters given in Example 1.

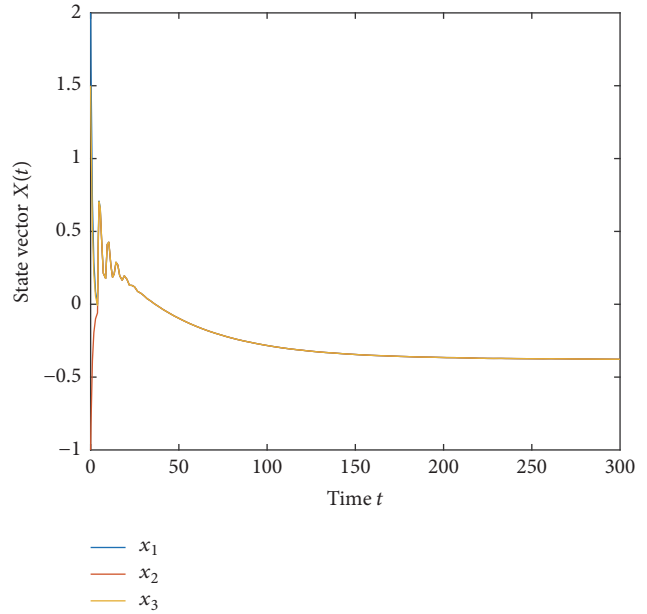


FIGURE 2: State vector of system (12) with $\tau = 4$.

result in Figure 3 shows that the state vector of system (21) is divergent. This means that even small time delay can cause the state vector of system (21) to be unstable. Hence, when time delays must be considered, model (19) has more superiority than model (21).

4.2. Application to Compressed Sensing. The sparsest solution of an undermined linear system of equations can be found by solving the so-called ℓ_0 -norm minimization problem; that is,

$$\min_{X \in \mathcal{R}^n} \|X\|_0, \quad \text{s.t. } AX = b, \quad (52)$$

where $A \in \mathcal{R}^{k \times n}$, $k < n$, $b \in \mathcal{R}^k$, and $\|X\|_0$ denotes the number of nonzero components in X . It is well known that

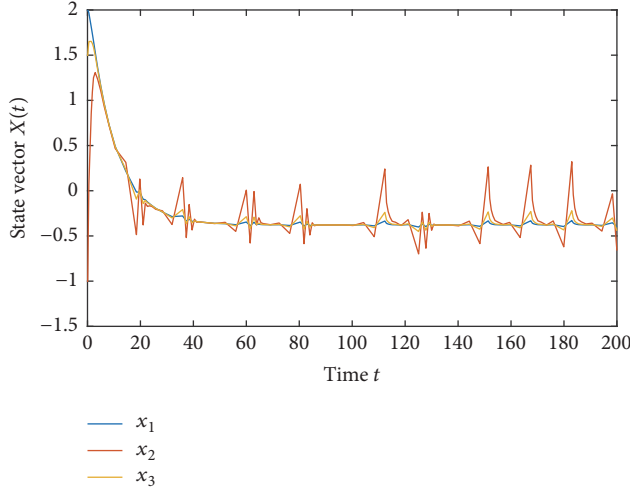


FIGURE 3: State vector of system (21) with $\tau = 0.01$.

problem (52) is NP-hard. For this problem, an approximation model named “Basic Pursuit Problem” was proposed as

$$\min_{X \in \mathcal{R}^n} \|X\|_1, \quad \text{s.t. } AX = b, \quad (53)$$

since the convex envelope of $\|X\|_0$ is $\|X\|_1$, where $\|X\|_1 = \sum_i |x_i|$ is the “ ℓ_1 ”-norm of X . Further, if b is contaminated by small dense noise, then problem (53) can be modified as the following form:

$$\min_{X \in \mathcal{R}^n} \|X\|_1, \quad \text{s.t. } \|AX - b\|_2 < \varepsilon, \quad (54)$$

where ε is nonnegative parameter and $\|\cdot\|_2$ is the Euclidean norm of X . Recently, convex analysis shows that (54) is equivalent to the following Lagrangian version:

$$\min_{X \in \mathcal{R}^n} \frac{1}{2} \|AX - b\|_2 + \tau \|X\|_1, \quad (55)$$

where $\tau > 0$ is a penalty parameter. Obviously, optimization problem (55) is convex but not differentiable. In order to solve problem (55), Liu and Hu [32] introduced a transform $u_i = \max\{x_i, 0\}$, $v_i = \max\{-x_i, 0\}$, and $X = U - V$. It follows that $\|X\|_1 = \mathbf{1}_n^T U + \mathbf{1}_n^T V$, where $\mathbf{1}_n = (1, 1, \dots, 1)^T$ is the vector consisting of n ones. Therefore, problem (55) can be rewritten as follows:

$$\begin{aligned} \min_{U, V \in \mathcal{R}^n} \quad & \frac{1}{2} \|b - A(U - V)\|_2^2 + \tau \mathbf{1}_n^T U + \tau \mathbf{1}_n^T V, \\ \text{s.t.} \quad & U > 0 \\ & V > 0. \end{aligned} \quad (56)$$

Namely,

$$\min_{Z \geq 0} F(Z) = \frac{1}{2} Z^T B Z + c^T Z, \quad (57)$$

where

$$\begin{aligned} Z &= \begin{bmatrix} U \\ V \end{bmatrix}, \\ c &= \tau \mathbf{1}_{2n}^T + \begin{bmatrix} -\bar{b} \\ \bar{b} \end{bmatrix}, \\ \bar{b} &= A^T b, \\ B &= \begin{bmatrix} A^T A & -A^T A \\ -A^T A & A^T A \end{bmatrix}. \end{aligned} \quad (58)$$

Obviously, problem (57) is a typical quadratic optimization problem. From (1) and (2), the optimal solutions of (57) are equivalent to the equilibrium points of the following projective neural network:

$$\frac{dZ}{dt} = -Z(t) + P_\Omega(Z(t) - \rho B Z(t) - \rho c), \quad (59)$$

$$Z(t_0) = Z_0 \in \Omega,$$

where Ω is the positive orthant, $\rho > 0$, and $P_\Omega(z_i) = \max\{z_i, 0\}$. By direct computing, it follows that

$$0 \leq \frac{P_\Omega(z_i) - P_\Omega(z_j)}{z_i - z_j} \leq 1, \quad z_i \neq z_j. \quad (60)$$

From the analysis in Section 2, system (59) can be rewritten in the following equivalent linear robust neural network form:

$$\frac{dy(t)}{dt} = [-I + (I - \rho B) \Delta(t)] y(t). \quad (61)$$

Let

$$A = \begin{bmatrix} 3 & 5 & 8 & 4 & 1 & 5 \\ 2 & 9 & 6 & 5 & 7 & 4 \\ 3 & 4 & 7 & 2 & 1 & 6 \\ 8 & 9 & 6 & 5 & 7 & 4 \end{bmatrix}, \quad (62)$$

$$\rho = 0.0008;$$

direct computing shows that $\|I - \rho B\|_2 = 1$; from Theorem 15, if we adopt initial x_0 in feasible set Ω , then the state vector of system (61) converges to optimizing solution of problem (55).

Remark 17. In addition to the application to compressed sensing, projection neural networks can also be applied to the motion generation and control of redundant robot manipulators. In [33], based on control perspective and projection neural networks, the authors researched distributed task allocation problem of multiple robots. Utilizing projection neural networks, literature [34] researches manipulability optimization problem of redundant manipulators. These new applications in robotics extend new application fields for projection neural network. How to use the technique derived in this paper to robotics field and solve related optimization problem will be a meaningful work, and this is also our future work direction.

5. Conclusions

This paper proposed three linear robust systems to solve a class of quadratic optimization problems. Utilizing LMI technique, eigenvalue perturbation theory, Lyapunov-Razumikhin method, and LaSalle's invariance principle, some stable criteria for related models are also established. Compared with previous criteria derived in the literature cited herein, the stable criteria established in this paper are less conservative and more practicable. Meanwhile, simulation results show that time-delay linear robust system type II is more sensitive to time delay than type I. This means that, in practical engineering problems, when time delay is needed to consider, model (19) has more superiority than model (21). Simulation example and application example in compressed sensing show that the results derived in this paper are valid.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grants 61472093 and 11461082) and Scientific Research Fund Project in Guizhou Provincial Department of Education (Grant KY[2014]243).

References

- [1] L. Bravi, V. Piccialli, and M. Sciarone, "An optimization-based method for feature ranking in nonlinear regression problems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 1005–1010, 2016.
- [2] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*, Wiley, New York, NY, USA, 1993, Neural Networks for Optimization and Signal Processing.
- [3] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*, John Wiley & Sons, Ltd, Chichester, UK, 2002.
- [4] Y. Cheung and L. Xu, "Dual multivariate auto-regressive modeling in state space for temporal signal separation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 33, no. 5, p. 835, 2003.
- [5] Q. Liu and J. Wang, "L1-minimization algorithms for sparse signal reconstruction based on a projection neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 3, pp. 698–707, 2016.
- [6] B. He, X. Yuan, and W. Zhang, "A customized proximal point algorithm for convex minimization with linear constraints," *Computational Optimization and Applications. An International Journal*, vol. 56, no. 3, pp. 559–572, 2013.
- [7] B. He and X. Yuan, "Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 119–149, 2012.
- [8] G. Gu, B. He, and X. Yuan, "Customized proximal point algorithms for linearly constrained convex minimization and saddle-point problems: a unified approach," *Computational Optimization and Applications. An International Journal*, vol. 59, no. 1-2, pp. 135–161, 2014.
- [9] M. P. Kennedy and L. O. Chua, "Neural networks for nonlinear programming," *Institute of Electrical and Electronics Engineers. Transactions on Circuits and Systems*, vol. 35, no. 5, pp. 554–562, 1988.
- [10] X. S. Zhang, *Neural Networks in Optimization*, Kluwer Academic Publishers, Norwell, Mass, USA, 2000.
- [11] C. Yin, S.-m. Zhong, X. Huang, and Y. Cheng, "Robust stability analysis of fractional-order uncertain singular nonlinear system with external disturbance," *Applied Mathematics and Computation*, vol. 269, article 21461, pp. 351–362, 2015.
- [12] I. B. Pyne, "Linear programming on an electronic analogue computer," *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, vol. 75, no. 2, pp. 139–143, 1956.
- [13] D. W. Tank and J. J. Hopfield, "Simple 'neural' optimization networks: an A/D converter, signal decision circuit, and a linear programming circuit," *IEEE Transactions on Circuits and Systems*, vol. 33, no. 5, pp. 533–541, 1986.
- [14] S. Zhang and A. G. Constantinides, "Lagrange programming neural networks," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 39, no. 7, pp. 441–452, 1992.
- [15] F. M. Zhang, "Exponential stability of quadratic optimization on neural networks," *J. Huazhong Univ. of Sci. Tech*, vol. 36, no. 5, pp. 57–59, 2008.
- [16] K. C. Tan, H. J. Tang, and Z. Yi, "Global exponential stability of discrete-time neural networks for constrained quadratic optimization," *Neurocomputing*, vol. 56, no. 1-4, pp. 399–406, 2004.
- [17] M. Yashtini and A. Malek, "A discrete-time neural network for solving nonlinear convex problems with hybrid constraints," *Applied Mathematics and Computation*, vol. 195, no. 2, pp. 576–584, 2008.
- [18] A. Bouzerdoum and T. R. Pattison, "Neural network for quadratic optimization with bound constraints," *IEEE Transactions on Neural Networks*, vol. 4, no. 2, pp. 293–304, 1993.
- [19] 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.
- [20] Y. Xia and J. Wang, "A dual neural network for kinematic control of redundant robot manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 31, no. 1, pp. 147–154, 2001.
- [21] Y. Xia and J. Wang, "A bi-projection neural network for solving constrained quadratic optimization problems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 2, pp. 214–224, 2016.
- [22] Q. Liu and J. Wang, "A projection neural network for constrained quadratic minimax optimization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 11, pp. 2891–2900, 2015.
- [23] Q. Liu and J. Wang, "A one-layer projection neural network for nonsmooth optimization subject to linear equalities and bound constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 5, pp. 812–824, 2013.
- [24] S. Zhou and J. Lam, "Robust stabilization of delayed singular systems with linear fractional parametric uncertainties," *Circuits, Systems, and Signal Processing*, vol. 22, no. 6, pp. 579–588, 2003.
- [25] J. G. Zhang, *Matrix Perturbation Analysis*, Science Press, Beijing, China, 2001.

- [26] T. Z. Huang, S. M. Zhong, and Z. L. Li, *Matrix Theory*, Higher Education Press, Beijing, China, 2007.
- [27] N. Toshiki et al., *Differential Equations with Time Lag: Introduction to Functional Differential Equations*, Makino Shoten, Tokyo, Japan, 2002.
- [28] X. X. Liao, *Theory Methods and Application of Stability*, Huazhong Univ. of Sci. Tech Press, Wuhan, China, 2005.
- [29] M. J. Pérez-Ilzarbe, "Convergence analysis of a discrete-time recurrent neural network to perform quadratic real optimization with bound constraints," *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1344–1351, 1998.
- [30] 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.
- [31] Z. Liu, S. Lv, and S. Zhong, "Novel exponential stability conditions for a class of interval projection neural networks," *International Journal of Biomathematics*, vol. 2, no. 3, pp. 287–297, 2009.
- [32] Y. Liu and J. Hu, "A neural network for ℓ_1 - ℓ_2 minimization based on scaled gradient projection: Application to compressed sensing," *Neurocomputing*, vol. 173, pp. 988–993, 2016.
- [33] L. Jin and S. Li, "Distributed task allocation of multiple robots: a control perspective," *IEEE Transactions on Systems Man & Cybernetics Systems*, 2016.
- [34] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4710–4720, 2017.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

