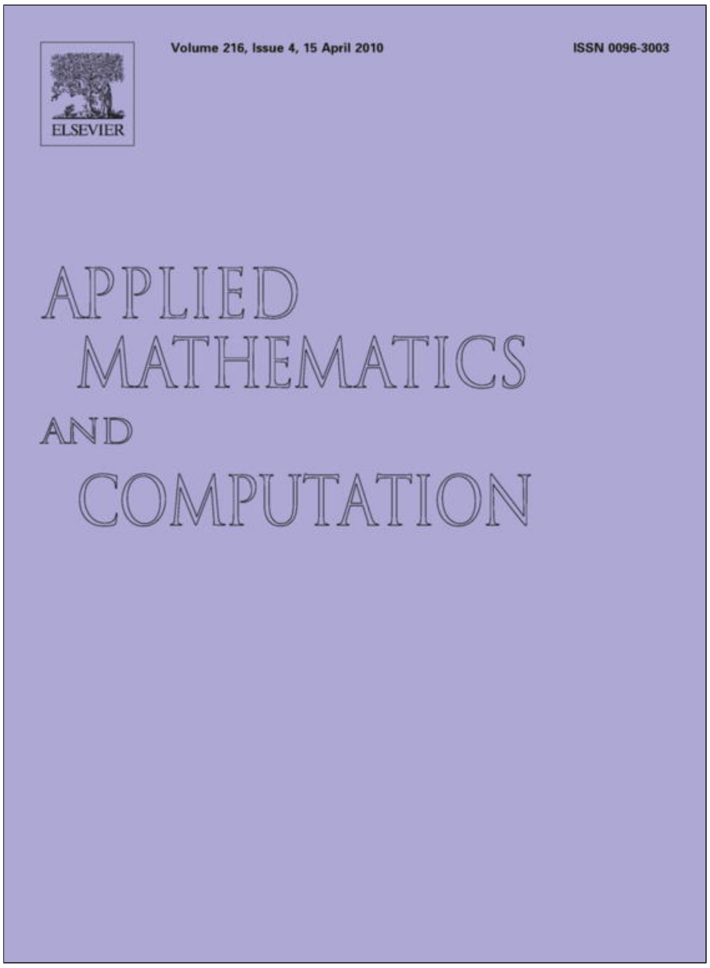
Volume 216, Issue 4, 15 April 2010
ISSN 0096-3003
ELSEVIER
APPLIED MATHEMATICS AND COMPUTATION

Applied Mathematics and Computation 216 (2010) 1181–1193

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Dynamic system methods for solving mixed linear matrix inequalities and linear vector inequalities and equalities

Xiaolin Hu

State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Keywords:
Dynamic system
Linear matrix inequality (LMI)
Semidefinite programming
Recurrent neural network
Stability analysis

ABSTRACT

A novel idea is proposed for solving a system of mixed linear matrix inequalities and linear vector inequalities and equalities. First, the problem is converted into an unconstrained minimization problem with a continuously differentiable convex objective function. Then, a continuous-time dynamic system and a discrete-time dynamic system are proposed for solving it. Under some mild conditions, the proposed dynamic systems are shown to be globally convergent to a solution of the problem. The merits of the methods refer to their simple numerical implementations and capability for handling nonstrict LMIs easily. In addition, the methods are promising in neural circuits realization, and therefore have potential applications in many online control problems. Several numerical examples are presented to illustrate the performance of the methods and substantiate the theoretical results.

1. Introduction

Linear matrix inequalities (LMIs) have been playing an increasingly important role in the field of optimization and control theory because a wide variety of problems (linear and convex quadratic inequalities, matrix norm inequalities, convex constraints etc.) can be written as LMIs [1–3]. In addition, LMIs have found many applications in exploring properties of recurrent neural networks, as their stability conditions are often expressed with the aid of LMIs (see, e.g., [4–7]).

LMIs are a class of convex optimization problems, and there exist many methods for solving them, e.g., the ellipsoid algorithm [8], the projective algorithm [9], and various interior-point methods (e.g., [10,11]). See [1,3] for excellent surveys of these methods. In the paper, two dynamic system methods are proposed for solving LMIs. The motivation is twofold. One reason is that analog neural computing defined by dynamic systems is regarded as a very promising approach for solving computationally expensive problems in real-time [12–14,22,15–17,19–21,23,26,25,27,24,18], and it is believed that the current research may pave a way for designing more powerful recurrent neural networks for solving LMIs. In 2000, a novel neural network method was proposed for solving LMIs [28]. The basic idea involves two steps. First, construct an energy function whose minima correspond to the solutions of the LMI. Second, construct a dynamic system with its right-hand-side being the negative gradient of the energy function. Solving this dynamic system will lead to a solution of the LMI. The limitation of this approach lies in its inability to handle nonstrict LMIs because it adopts the Cholesky factorization of positive definite matrices which cannot be generalized to positive semidefinite matrices. The other reason is that understanding the properties and features of such dynamic systems is helpful for developing new efficient iterative algorithms for solving LMIs. Some merits can be extracted from the dynamic systems. For example, it is known that many popular numerical algorithms such as the aforementioned interior-point algorithms [10,11,1,3] handle strict LMIs only; and in order to achieve a solution on the

E-mail address: xlhu@tsinghua.edu.cn

0096-3003/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2010.02.010

boundary of the feasible region, theoretically, some parameters in these algorithms should approach infinity.[1] It will be seen that the proposed methods in the paper do not require such parameters but can converge to exact solutions. For another example, the equivalence between LMIs and unconstrained optimization problems, which is disclosed in Section 2 can inspire some efficient iterative algorithms (e.g., some variants of the Newton method).

In the paper, based on the two steps similar to those in Lin et al. [28], two different dynamic systems are constructed to solve LMIs, both of which are capable of handling nonstrict LMIs. The idea of the proposed methods originates from two recurrent neural networks for solving linear vector equalities and inequalities [14]. Actually, the present paper addresses a natural, but interesting, extension of the idea in Xia et al. [14]. As this extension entails an operation of projecting a matrix onto a cone, which is not as easy as the projection operations in Xia et al. [14], the proposed dynamic systems can not be implemented in recurrent neural networks as easy as the counterparts in Xia et al. [14]. However, with the fast development of neural network theories, it is possible that this projecting operation can be realized by neural networks as well. On the other hand, if such neural networks are not available, some high-speed digital equipments specialized for this operation can be coupled with the neural networks whose state equations are defined by the proposed dynamic systems.

To make the proposed methods more versatile in practice, I consider solving LMIs together with conventional linear vector inequalities and equalities.

Throughout the paper, the following notations are used unless otherwise specified. $\mathbb{R}^n$ denotes the n dimensional real space, and $\mathbb{R}^n_+$ denotes the set of all n dimensional nonnegative vectors. $\mathbb{S}^n$ denotes the set of all $n \times n$ symmetric matrices, and $\mathbb{S}^n_+$ denotes the set of all $n \times n$ symmetric positive semidefinite matrices. If $A \in \mathbb{R}^{m\times n}$, $\mathrm{vec}(A) \in \mathbb{R}^{mn}$ is defined as a vector formed by stacking the columns of A one by one from left to right. If $x \in \mathbb{R}^m$, then $\|x\|_2$ denotes its l_2-norm, i.e., $\|x\|_2 = \sqrt{\sum_{i=1}^m x_i^2}$. The inner product equipped with the real matrix space $\mathbb{R}^{m\times n}$ is defined as $\langle A, B\rangle = \mathrm{tr}(A^TB) = \sum_{i=1}^m \sum_{j=1}^n a_{ij}b_{ij}, \forall A, B \in \mathbb{R}^{m\times n}$, where tr stands for the trace of a matrix. The induced norm in the matrix space is then defined as $\|A\|_{\mathscr{F}} = \sqrt{\langle A, A\rangle} = \|\mathrm{vec}(A)\|$, which is called the Frobenius norm. Moreover, $\|A\|_2$ denotes the l_2-norm of the matrix A, which is defined as $\|A\|_2 = \sqrt{\max(\lambda(A^TA))}$, where $\lambda(A^TA)$ stands for the eigenvalues of A^TA. If a, b are two vectors, then $a \geqslant (>) b$ is interpreted as $a_i \geqslant (>) b_i$ for each i. Similar definitions apply to $\leqslant$ and $<$ between two vectors. A square symmetric matrix $A \geqslant (>) 0$ is interpreted as A is positive semidefinite (positive definite). On the contrary, $A \leqslant (<) 0$ is interpreted as -A is positive semidefinite (positive definite).

2. Problem formulation

The standard linear matrix inequality (LMI) problem is to find a vector $x \in \mathbb{R}^m$ such that

$$N(x) \triangleq N_0 + \sum_{i=1}^m N_i x_i \geqslant 0, \tag{1}$$

where $N_i \in \mathbb{S}^n, i = 0, 1, \ldots, m$. The equation is called a nonstrict LMI; correspondingly, $N(x) > 0$ is called a strict LMI. Note that most LMIs commonly encountered in control applications are not expressed in the canonical form (1), for instance, the generalized Lyapunov inequality

$$BPA + A^TPB^T + D < 0, \quad P > 0 \tag{2}$$

studied in Lin et al. [28] and the following inequality

$$\begin{pmatrix} -A^TP - PA - Q & PB \\ B^TP & R \end{pmatrix} \geqslant 0$$

studied in Boyd et al. [1], where $A, B, D = D^T > 0, Q = Q^T, R = R^T > 0$ are given matrices and $P^T = P$ is the matrix variable. It is easy to rewrite such LMIs in the canonical form (1) by using the following results, which can be deduced easily.

Proposition 1. *Let $U \in \mathbb{R}^{n\times m}, P \in \mathbb{R}^{m\times m}, V \in \mathbb{R}^{m\times r}$ be three matrices whose scalar forms are, respectively, defined as $\{u_{ij}\}_{n\times m}, \{p_{ij}\}_{m\times m}, \{v_{ij}\}_{m\times r}$. Then the following hold:*

(1) $UP = \sum_{s=1}^m \sum_{t=1}^m p_{st}K(p_{st})$, where $K(p_{st}) = \{k_{ij}\}_{n\times m}$ with $k_{ij} = u_{is}$ if $j = t$ and 0 otherwise,
(2) $PV = \sum_{s=1}^m \sum_{t=1}^m p_{st}K(p_{st})$, where $K(p_{st}) = \{k_{ij}\}_{m\times r}$ with $k_{ij} = v_{tj}$ if $i = s$ and 0 otherwise,
(3) $UPV = \sum_{s=1}^m \sum_{t=1}^m p_{st}K(p_{st})$, where $K(p_{st}) = \{k_{ij}\}_{n\times r}$ with $k_{ij} = u_{is}v_{tj}$.

Many popular methods, e.g., [9,28], solve strict LMIs; or treat nonstrict LMIs as strict LMIs with some degree of approximation (e.g., any interior-point method). However, such methods may fail on some nonstrict LMIs. A simple example is to solve $\mathrm{diag}(x, -x) \geqslant 0$ where x is a scalar. The only solution is $x = 0$ and any small perturbation to it will lead to an infeasible

[1] In practice, however, the parameters only need to increase to some values to achieve specified solution accuracy.

solution. In fact, every nonstrict LMI involves an implicite equality constraint [1]. Theoretically, any feasible nonstrict LMI (including the above example) can be reduced to an equivalent feasible strict LMI by eliminating implicit equality constraints and then reducing the resulting LMI by removing any constant nullspace. But in practice this is a tough task (see [1, pp. 20–21] for an impression of how tough the task is, where a simple nonstrict Lyapunov inequality is converted to a strict LMI), which turns out not to be a wise idea for solving nonstrict LMIs.

In the paper, I consider solving a more general problem: find a vector $x \in \mathbb{R}^m$ such that the following hold

$$Ax \geqslant b, \quad Cx = d, \quad N(x) \geqslant 0, \tag{3}$$

where $A \in \mathbb{R}^{p\times m}, b \in \mathbb{R}^p, C \in \mathbb{R}^{q\times m}, d \in \mathbb{R}^q$ and $N(x)$ is defined in (1). Note that the linear vector inequality $Ax \geqslant b$ can be written in the form of LMI (1) as well. But in general LMIs are harder to deal with than linear vector inequalities, therefore $Ax \geqslant b$ is not put into the LMI form. In what follows, the solution set of (3) is denoted by $\mathbb{X}^*$.

Problem (3) includes many problems as special cases. For example, consider the linear semidefinite programming problem:

$$\begin{aligned} &\text{minimize} \quad h^T y \\ &\text{subject to } \; F(y) \triangleq F_0 + \sum_{i=1}^{m_1} F_i y_i \geqslant 0, \end{aligned} \tag{4}$$

where $h, y \in \mathbb{R}^{m_1}$ and $F_i \in \mathbb{S}^{n_1} (i = 0, 1, \ldots, m_1)$. The dual problem is [2]

$$\begin{aligned} &\text{maximize} \quad -\operatorname{tr}(F_0 W) \\ &\text{subject to } \; \operatorname{tr}(F_i W) = h_i, \quad i = 1, \ldots, m_1, \\ &\qquad\qquad\quad W \in \mathbb{S}^{n_1}_+. \end{aligned}$$

In view that W is symmetric, only $n_1(n_1+1)/2$ components are independent. Let $W = \{w_{ij}\}_{n_1\times n_1}$. By introducing a new variable

$$z = ((w_{11}, \ldots, w_{n_1 1}), (w_{22}, \ldots, w_{n_1 2}), \ldots, w_{n_1 n_1})^T \in \mathbb{R}^r \tag{5}$$

where $r = n_1(n_1+1)/2$, and an operator $\overline{\operatorname{vec}} : \mathbb{S}^{n_1} \to \mathbb{R}^r$,

$$\overline{\operatorname{vec}}(K) = ((k_{11}, 2k_{21}, \ldots, 2k_{n_1 1}), (k_{22}, 2k_{32}, \ldots, 2k_{n_1 2}), \ldots, k_{n_1 n_1})^T$$

where $K = \{k_{ij}\}_{n_1\times n_1}$, the dual problem can be rewritten as

$$\begin{aligned} &\text{maximize} \quad -\overline{\operatorname{vec}}(F_0)^T z \\ &\text{subject to } \; Qz = h, \\ &\qquad\qquad\quad \sum_{i=1}^{r} G_i z_i \geqslant 0, \end{aligned} \tag{6}$$

where $Q = (\overline{\operatorname{vec}}(F_1), \ldots, \overline{\operatorname{vec}}(F_{m_1}))^T$ and $G_i = \{g^{(i)}_{jk}\} \in \mathbb{S}^{n_1}$ with

$$g^{(i)}_{jk} = \begin{cases} 1 & \text{if } z_i \text{ corresponds to } w_{jk} \text{ in (5)}, \\ 0 & \text{otherwise}, \end{cases}$$

and $g^{(i)}_{kj} = g^{(i)}_{jk}$, for all $j, k = 1, \ldots, n_1$ and $j \geqslant k$. According to the primal-dual theorem [3], under some standard constraint qualifications, a point y solves the primal problem (4) if and only if there exists $z \in \mathbb{R}^r$ such that the following holds:

$$\begin{cases} h^T y = -\overline{\operatorname{vec}}(F_0)^T z, \quad Qz = h, \\ F_0 + \sum\limits_{i=1}^{m_1} F_i y_i \geqslant 0, \quad \sum\limits_{i=1}^{r} G_i z_i \geqslant 0. \end{cases}$$

This system can be put into the form of (3) with

$$x = \begin{pmatrix} y \\ z \end{pmatrix}, \quad C = \begin{pmatrix} h^T & \overline{\operatorname{vec}}(F_0)^T \\ 0_{m_1\times m_1} & Q_{m_1\times r} \end{pmatrix}, \quad d = \begin{pmatrix} 0 \\ h \end{pmatrix}, N_i = \begin{pmatrix} F_i & \\ & 0_{n_1\times n_1} \end{pmatrix}, \quad \forall i = 0, 1, \ldots, m_1,$$

$$N_i = \begin{pmatrix} 0_{n_1\times n_1} & \\ & G_{i-m_1} \end{pmatrix}, \quad \forall i = m_1+1, \ldots, m_1+r,$$

and $A = 0, b = 0$.

For the convenience of later discussion, the following chain rule for computing the derivatives of compound functions is introduced.

Lemma 1. *Suppose that there are two continuously differentiable functions $h : \mathbb{R}^{n\times n} \to \mathbb{R}$ and $g : \mathbb{R}^m \to \mathbb{R}^{n\times n}$. Define a compound function $f = h \circ g : \mathbb{R}^m \to \mathbb{R}$ as*

$$f(x) = h(g(x)), \quad \operatorname{dom} f = \{x \in \operatorname{dom} g | g(x) \in \operatorname{dom} h\}.$$

Then

$$\frac{\partial f(x)}{\partial x} = \begin{pmatrix} \langle \partial h/\partial g, \partial g/\partial x_1 \rangle \\ \vdots \\ \langle \partial h/\partial g, \partial g/\partial x_m \rangle \end{pmatrix},$$

where the derivative of the scalar valued-function f with respect to a matrix g is defined by

$$\frac{\partial f}{\partial g} = \left\{ \frac{\partial f}{\partial g_{ij}} \right\}_{n\times n}, \quad i,j = 1,\ldots,n,$$

and the derivative of g with respect to a scalar x_k is defined by

$$\frac{\partial g}{\partial x_k} = \left\{ \frac{\partial g_{ij}}{\partial x_k} \right\}_{n\times n}, \quad k = 1,\ldots,m.$$

Proof. The result follows from

$$\frac{\partial f(x)}{\partial x_k} = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial f}{\partial g_{ij}} \frac{\partial g_{ij}}{\partial x_k} = \langle \partial h/\partial g, \partial g/\partial x_k \rangle, \quad \forall k = 1,\ldots,m,$$

immediately by considering the definition of inner product of two matrices. □

Similar to [14], an energy function associated with problem (1) is defined first:

$$E(x) = \frac{1}{2}\{\|Ax - b - (Ax - b)^+\|_2^2 + \|Cx - d\|_2^2 + \|N(x) - N(x)^+\|_{\mathscr{F}}^2\}, \tag{7}$$

where $(\cdot)^+$ stands for a projection operator. Specifically, if $x \in \mathbb{R}^n$, x^+ denotes its projection onto $\mathbb{R}_+^n$; if $M \in \mathbb{S}^n$, then M^+ denotes its projection onto $\mathbb{S}_+^n$. The following two lemmas disclose some basic properties of these projections.

Lemma 2 [30]. *Let Ω be a closed convex set in a Euclidean space $\mathbb{L}$ endowed with any inner product $\langle\cdot,\cdot\rangle$ and norm $\|\cdot\|$, and let x^+ denote the projection of a point $x \in \mathbb{L}$ onto Ω, i.e., $x^+ = \arg\min_{v\in\Omega}\|x - v\|$. Then for any $x, y \in \mathbb{L}$ and any $v \in \Omega$, we have*

$$\langle x^+ - x, v - x^+ \rangle \geqslant 0, \tag{8}$$

and

$$\|x^+ - y^+\| \leqslant \|x - y\|. \tag{9}$$

Lemma 3. *Let $\Omega, \mathbb{L}, \langle\cdot,\cdot\rangle, \|\cdot\|$ be defined the same as in Lemma 2. Then the function $f : \mathbb{L} \to \mathbb{R}$ defined as $f(x) = \|x - x^+\|$ is convex in x, i.e.,*

$$f(\theta x + (1-\theta)y) \leqslant \theta f(x) + (1-\theta)f(y) \quad \forall x,y \in \mathbb{L} \text{ and } 0 \leqslant \theta \leqslant 1.$$

Moreover, the function $g : \mathbb{L} \to \mathbb{R}$ defined as $g(x) = \|x - x^+\|^2$ is convex in x, too, and continuously differentiable with gradient $\nabla g(x) = 2(x - x^+)$.

Proof. Since Ω is convex and $x^+, y^+ \in \Omega$, we have $\theta x^+ + (1-\theta)y^+ \in \Omega$ for $0 \leqslant \theta \leqslant 1$. Let $u = \theta x + (1-\theta)y$ and $v = \theta x^+ + (1-\theta)y^+$. According to the definition of the projection operator we have

$$f(\theta x + (1-\theta)y) = \|u - u^+\| \leqslant \|u - v\|.$$

The convexity of f then follows from the fact

$$\|u - v\| = \|\theta x + (1-\theta)y - \theta x^+ - (1-\theta)y^+\| \leqslant \|\theta x - \theta x^+\| + \|(1-\theta)y - (1-\theta)y^+\| = \theta\|x - x^+\| + (1-\theta)\|y - y^+\|.$$

Similarly,

$$g(\theta x + (1-\theta)y) = \|u - u^+\|^2 \leqslant \|u - v\|^2.$$

By noticing that

$$\begin{aligned}\theta(1-\theta)\|(x - x^+) - (y - y^+)\|^2 &= (\theta - \theta^2)\|x - x^+\|^2 + (\theta - \theta^2)\|y - y^+\|^2 - 2\theta(1-\theta)\langle x - x^+, y - y^+\rangle \\ &= \theta\|x - x^+\|^2 + (1-\theta)\|y - y^+\|^2 - \theta^2\|x - x^+\|^2 - (1-\theta)^2\|y - y^+\|^2 \\ &\quad - 2\theta(1-\theta)\langle x - x^+, y - y^+\rangle \\ &= \theta\|x - x^+\|^2 + (1-\theta)\|y - y^+\|^2 - \|\theta x - \theta x^+ + (1-\theta)y - (1-\theta)y^+\|^2 \\ &= \theta g(x) + (1-\theta)g(y) - \|u - v\|^2 \geqslant 0,\end{aligned}$$

we can conclude the convexity of $g(x)$. Define a function $\tilde{g}:\mathbb{L}\times\mathbb{L}\to\mathbb{R}$ as $\tilde{g}(x,y)=\|x-y\|^2$, then

$$g(x)=\min_{y\in\Omega}\tilde{g}(x,y).$$

Notice that $\tilde{g}$ is continuously differentiable in both x and y, and the minimum solution of the right-hand-side of above equation is uniquely attained at $y^*=x^+$ for any fixed $x\in\mathbb{L}$, then it follows from [31, Chapter 4, Theorem 1.7] that $g(x)$ is differentiable and $\nabla g(x)=\nabla_x\tilde{g}(x,y^*)|_{y^*=x^+}=2(x-x^+)$. □

Lemma 3 shows that the distance (in any norm) to the nearest point in a closed convex set Ω, defined as $f(x)$, is a convex function. I would like to mention that the distance (in any norm) to the farthest point in an arbitrary set $\mathbb{C}\subseteq\mathbb{L}$, defined as $\tilde{f}(x)=\sup_{y\in\mathbb{C}}\|x-y\|$, is also a convex function [29, p.81].

Lemma 4. [32] *Let $X\in\mathbb{S}^n$ be a given symmetric matrix and let $X=QDQ^T$ be the eigenvalue–eigenvector decomposition of X, where D is a diagonal matrix of eigenvalues and Q is an orthogonal matrix of normalized eigenvectors. Then $X^+=QD^+Q^T$, where D^+ denotes the diagonal matrix obtained by replacing the negative components of D by zeros.*

Clearly, for any scalar α we have $\alpha-\alpha^+=-(-\alpha)^+$. Then for any vector a we have $a-a^+=-(-a)^+$. Moreover, from Lemma 4, it is easy to see that $M-M^+=-(-M)^+$ where $M\in\mathbb{S}^n$. Let $E_i(x), i=1,\cdots,3$ denote the three terms in $E(x)$ in (7) from left to right. Then, from Lemmas 1 and 3 we can deduce the following theorem easily.

Theorem 1. *The function $E(x)=\sum_{i=1}^3E_i(x)$ is convex and continuously differentiable with gradient $\nabla E(x)=\sum_{i=1}^3\nabla E_i(x)$, where*

$$\nabla E_1(x)=A^T[(Ax-b)-(Ax-b)^+]=-A^T(b-Ax)^+,$$

$$\nabla E_2(x)=C^T(Cx-d),$$

$$\nabla E_3(x)=\begin{pmatrix}\langle N(x)-N(x)^+,N_1\rangle\\ \vdots\\ \langle N(x)-N(x)^+,N_m\rangle\end{pmatrix}=-\begin{pmatrix}\langle(-N(x))^+,N_1\rangle\\ \vdots\\ \langle(-N(x))^+,N_m\rangle\end{pmatrix}.$$

Moreover, $E(x)\geqslant 0$, and the equality holds if and only if $x\in\mathbb{X}^*$.

In what follows, let $\mathbb{X}^\dagger=\{x\in\mathbb{R}^m|E(x)\leqslant E(y),\forall y\in\mathbb{R}^m\}$. Clearly, $\mathbb{X}^\dagger=\mathbb{X}^*$ if $\mathbb{X}^*\neq\emptyset$, i.e., system (3) is solvable.

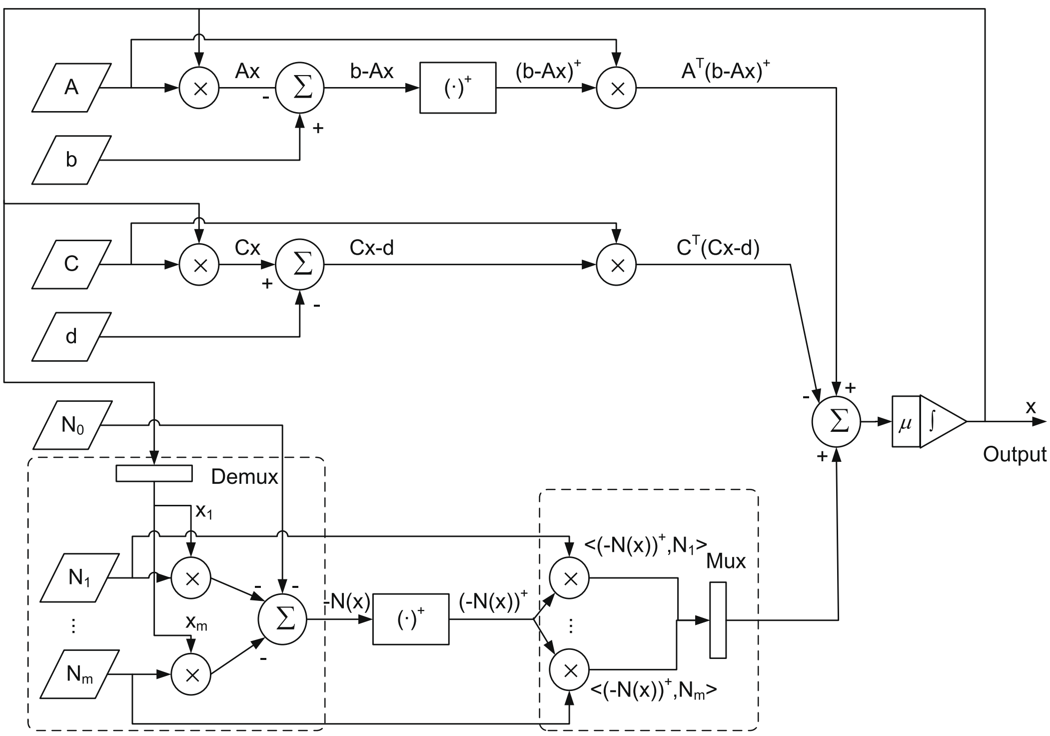

Fig. 1. Block diagram of the neural network for realizing continuous-time dynamic system (10). In the left dashed rectangle, the vector x is decomposed to components, while in the right dashed rectangle, some scalars join together to constitute a vector.

3. Dynamic system methods

3.1. Description of dynamic systems

Using the standard gradient descent method for the minimization of the function $E(x)$ in (7) we can derive the following continuous-time dynamic system

$$\frac{dx}{dt} = -\mu \nabla E(x) \tag{10}$$

and the corresponding discrete-time dynamic system

$$x^{k+1} = x^k - \eta \nabla E(x^k) \tag{11}$$

to solve (3), where ∇E is given in Theorem 1 and $\mu, \eta > 0$ are user-defined parameters.

It is seen that both of the two dynamic systems involve computing of the projections of vectors and matrices onto convex cones $\mathbb{R}^p_+$ and $\mathbb{S}^n_+$, respectively. The former can be readily calculated as $x^+ = (x_1^+, \ldots, x_p^+)$ with $x_i^+ = \max\{x_i, 0\}$, the latter, however, can not be calculated directly. In the paper, the method suggested by Lemma 4 is used.

It is seen that the dynamic systems (10) and (11) share similar forms to those in Xia et al. [14], in which the dynamic systems are shown to be realizable by neural circuits. Here arises a question concerning about the possibility of realizing (10) and (11) by neural circuits. Fig. 1 illustrates the block diagram of the possible neural network for realizing (10). Similarly, we can draw the block diagram for realizing (11). Obviously, the bottleneck of the potential neural network is located in the matrix projection block, and a high-speed computing equipment should be placed in it to accomplish the task. This computing unit can be either a digital processing unit – which results in a hybrid analog-digital approach, or another neural network – which results in a hierarchical neural network approach. Regarding the latter approach, currently, there do not exist such neural networks. Some clues can be found in [23,24,26,25] and references therein, all of which are about the eigenvalue and eigenvector problem, matrix diagonalization, matrix factorization, and so on from the perspective of neural networks, in view that the projection can be accomplished mainly by solving the eigenvalue-eigenvector decomposition problem (see Lemma 4).

3.2. Global convergence

In order to prove the convergence result of the continuous-time dynamic system (10), the following lemma is needed.

Lemma 5.

(1) $\mathbb{X}^e = \mathbb{X}^\dagger \neq \emptyset$, where $\mathbb{X}^e$ stands for the equilibrium set of *(10)*.
(2) *For any initial point* $x(t_0) = x^0 \in \mathbb{R}^m$, *there exists a unique solution* $x(t)$ *to* (10) *for* $t \in [t_0, \tau)$.

Proof. Since $E(x)$ defined in (7) is lower bounded, $\mathbb{X}^\dagger \neq \emptyset$. The fact $\mathbb{X}^\dagger = \mathbb{X}^e$ follows from the equivalence between $x \in \mathbb{X}^\dagger$ and $\nabla E(x) = 0$ as $E(x)$ is convex and continuously differentiable. Now I prove the second part of the lemma by showing that the right-hand-side of (10), denoted by $H(x)$, is Lipschitz continuous in $\mathbb{R}^m$. Actually, for any $x, y \in \mathbb{R}^m$, we can deduce

$$\begin{aligned}\|H(x) - H(y)\|_2 &= \left\| \sum_{k=1}^{3} \nabla E_k(x) - \nabla E_k(y) \right\|_2 \\ &\leqslant \|A\|_2 \|(b - Ax)^+ - (b - Ay)^+\|_2 + \|C^T C\|_2 \|x - y\|_2 + \sum_{i=1}^{m} |\langle (-N(x))^+ - (-N(y))^+, N_i \rangle| \\ &\leqslant \|A\|_{\mathscr{F}} \|(b - Ax) - (b - Ay)\|_2 + \|C^T C\|_{\mathscr{F}} \|x - y\|_2 + \sum_{i=1}^{m} \|(-N(x))^+ - (-N(y))^+\|_{\mathscr{F}} \|N_i\|_{\mathscr{F}} \\ &\leqslant (\|A\|_{\mathscr{F}}^2 + \|C^T C\|_{\mathscr{F}}) \|x - y\|_2 + \sum_{i=1}^{m} \|N(x) - N(y)\|_{\mathscr{F}} \|N_i\|_{\mathscr{F}},\end{aligned}$$

where Lemma 2, the Cauchy–Schwarz Inequality and the fact that the l_2-norm of a matrix is equal to or smaller than its Frobenius norm are used. Note that

$$\mathrm{vec}(N(x) - N(y)) = \sum_{i=1}^{m} \mathrm{vec}(N_i)(x_i - y_i) = (\,\mathrm{vec}(N_1) \quad \cdots \quad \mathrm{vec}(N_m)\,) \begin{pmatrix} x_1 - y_1 \\ \vdots \\ x_m - y_m \end{pmatrix} = \overline{N}(x - y),$$

where $\overline{N} = (\mathrm{vec}(N_1), \ldots, \mathrm{vec}(N_m)) \in \mathbb{R}^{n^2 \times m}$. Then $\|N(x) - N(y)\|_{\mathscr{F}} = \|\,\mathrm{vec}(N(x) - N(y))\|_2 = \|\overline{N}(x - y)\|_2$. Using these facts, we can obtain

$$\|H(x) - H(y)\|_2 \leqslant \left(\|A\|_{\mathscr{F}}^2 + \|C^T C\|_{\mathscr{F}} + \|\overline{N}\|_{\mathscr{F}} \sum_{i=1}^{m} \|N_i\|_{\mathscr{F}} \right) \|x - y\|_2.$$

Hence $H(x)$ is Lipschitz continuous in $\mathbb{R}^m$. According to the existence theorem of ordinary differential equations, there exists a unique solution to the dynamic system (10) for $t \in [t_0, \tau)$. The proof is completed. □

Theorem 2. *The continuous-time dynamic system (10) is stable in the sense of Lyapunov and globally convergent to a point in* $\mathbb{X}^\dagger$.

In view of Lemma 5 and the fact that there always exists at least one finite point $x^* \in \mathbb{X}^\dagger$, Theorem 2 can be established without much difficulty by following similar lines in the proof of Theorem 1 in Xia et al. [14]. The only difference is that one should choose a reference point x^* in $\mathbb{X}^\dagger$ instead of $\mathbb{X}^*$ in the proof. For brevity, the details are omitted here. The following important result is a direct consequence of Theorem 2.

Corollary 1. *If system (3) is solvable, i.e.,* $\mathbb{X}^* \neq \emptyset$, *then the continuous-time dynamic system (10) globally converges to a point in* $\mathbb{X}^*$.

Remark 1. If the LMI in (3) is absent, Corollary 1 reduces to Theorem 1 in Xia et al. [14]. In other words, [14] does not ascertain the global convergence of the corresponding dynamic system when (3) without LMI is unsolvable; while this result can be ascertained according to Theorem 2.

In view of the above results, if the dynamic system (10) converges to a point which does not solve (3), then (3) has no solution. This statement is useful in checking stability of many control systems and recurrent neural networks.

In what follows the convergence of the discrete-time dynamic system (11) is investigated. First, two lemmas are introduced.

Lemma 6. *For any* $x, y \in R^m$,

$$E(x) \leqslant E(y) + \nabla E(y)^T (x-y) + \frac{1}{2}\left(\|A^T A + C^T C\|_2 + \|\overline{N}\|_2^2\right)\|x-y\|_2^2,$$

where $\overline{N} = (\mathrm{vec}(N_1), \ldots, \mathrm{vec}(N_m))$.

Proof. Let $E_{12}(x) = E_1(x) + E_2(x)$. According to Proposition 3 in Xia et al. [14], we have

$$E_{12}(x) \leqslant E_{12}(y) + \nabla E_{12}(y)^T (x-y) + \frac{1}{2}\|A^T A + C^T C\|_2 \|x-y\|_2^2. \tag{12}$$

Regarding $E_3(x)$, the following holds

$$\begin{aligned} &E_3(y) - E_3(x) + \langle N(y) - N(y)^+, N(x) - N(y)\rangle + \frac{1}{2}\|N(x) - N(y)\|_{\mathscr{F}}^2 \\ &= \frac{1}{2}\|N(y) - N(y)^+\|_{\mathscr{F}}^2 + \langle N(y) - N(y)^+, N(x) - N(y)\rangle + \frac{1}{2}\|N(x) - N(y)\|_{\mathscr{F}}^2 - \frac{1}{2}\|N(x) - N(x)^+\|_{\mathscr{F}}^2 \\ &= \frac{1}{2}\|N(x) - N(y)^+\|_{\mathscr{F}}^2 - \frac{1}{2}\|N(x) - N(x)^+\|_{\mathscr{F}}^2 \geqslant 0. \end{aligned}$$

The last inequality above follows from the fact that $N(x)^+$ is nearest to $N(x)$ among all points in $\mathbb{S}_+^n$. By noticing

$$\begin{aligned} \langle N(y) - N(y)^+, N(x) - N(y)\rangle &= \left\langle N(y) - N(y)^+, \sum_{i=1}^m N_i(x_i - y_i)\right\rangle = \sum_{i=1}^m \langle N(y) - N(y)^+, N_i(x_i - y_i)\rangle \\ &= \begin{pmatrix} \langle N(y) - N(y)^+, N_1\rangle \\ \vdots \\ \langle N(y) - N(y)^+, N_m\rangle \end{pmatrix}^T (x-y) = \nabla E_3(y)^T (x-y) \end{aligned}$$

and

$$\|N(x) - N(y)\|_{\mathscr{F}}^2 = \|\overline{N}(x-y)\|_2^2 \leqslant \|\overline{N}\|_2^2 \|x-y\|_2^2$$

we have

$$E_3(x) \leqslant E_3(y) + \nabla E_3(y)^T (x-y) + \frac{1}{2}\|\overline{N}\|_2^2 \|x-y\|_2^2. \tag{13}$$

Adding (12) and (13) yields the desired result. □

Lemma 7.

(1) For any $x^* \in \mathbb{X}^\dagger$,

$$(x - x^*)^T \nabla E(x) \geqslant E(x) - E(x^*).$$

(2) For any $x^* \in \mathbb{X}^*$,

$$(x - x^*)^T \nabla E(x) \geqslant 2E(x).$$

Proof. The first part follows from a basic property of any continuously differentiable convex function [29]. I now prove the second part. Regarding $E_1(x)$, we have

$$\begin{aligned}(x - x^*)^T \nabla E_1(x) &= (x - x^*)^T A^T[(Ax - b) - (Ax - b)^+] \\ &= [(Ax - b) - (Ax - b)^+]^T[(Ax - b) - (Ax - b)^+ + (Ax - b)^+ - (Ax^* - b)] \\ &= \|(Ax - b) - (Ax - b)^+\|_2^2 + [(Ax - b) - (Ax - b)^+]^T[(Ax - b)^+ - (Ax^* - b)] \geqslant 2E_1(x)\end{aligned}$$

where Lemma 2 is used. Similarly we can show that

$$(x - x^*)^T \nabla E_3(x) = \langle N(x) - N(x)^+, N(x) - N(x^*)\rangle \geqslant 2E_3(x).$$

Regarding $E_2(x)$, we have

$$(x - x^*)^T \nabla E_2(x) = (Cx - Cx^*)^T(Cx - d) = 2E_2(x).$$

Adding the above three equations yields the desired result. □

The following theorem can be established similarly to Theorem 2 in Xia et al. [14].

Theorem 3. *The sequence* $\{x^k\}$ *generated by* (11) *is globally convergent to a point in* $\mathbb{X}^\dagger$ *if* $0 < \eta \leqslant 1/\rho$ *where*

$$\rho = \|A^TA + C^TC\|_2 + \|\overline{N}\|_2^2$$

with $\overline{N}$ *defined in* Lemma 6.

Proof. From Lemma 6 we have

$$E(x^{k+1}) \leqslant E(x^k) + (x^{k+1} - x^k)^T \nabla E(x^k) + \frac{1}{2}\left(\|A^TA + C^TC\|_2 + \|\overline{N}\|_2^2\right)\|x^{k+1} - x^k\|_2^2.$$

Substituting (11) into above yields

$$E(x^{k+1}) \leqslant E(x^k) - \eta\|\nabla E(x^k)\|_2^2 + \frac{\eta^2}{2}\left(\|A^TA + C^TC\|_2 + \|\overline{N}\|_2^2\right)\|\nabla E(x^k)\|_2^2.$$

Therefore the sequence $\{E(x^k)\}$ is monotonically deceasing and bounded above. Moreover,

$$\|\nabla E(x^k)\|_2^2 \leqslant \frac{2}{\eta(2 - \rho\eta)}(E(x^k) - E(x^{k+1})).$$

Then

$$\sum_{k=1}^m \|\nabla E(x^k)\|_2^2 \leqslant \frac{2}{\eta(2 - \rho\eta)}\sum_{k=1}^m (E(x^k) - E(x^{k+1})) = \frac{2}{\eta(2 - \rho\eta)}(E(x^1) - E(x^{m+1})),$$

which implies $\sum_{k=1}^\infty \|\nabla E(x^k)\|_2^2 \leqslant +\infty$ and

$$\lim_{k\to\infty} \|\nabla E(x^k)\|_2 = 0.$$

Then, for any $x^* \in \mathbb{X}^\dagger$,

$$\begin{aligned}\|x^{k+1} - x^*\|_2^2 &= \|x^k - x^*\|_2^2 + \eta^2\|\nabla E(x^k)\|_2^2 - 2\eta(x^k - x^*)^T\nabla E(x^k) \\ &\leqslant \|x^k - x^*\|_2^2 + \frac{2\eta}{2 - \rho\eta}(E(x^k) - E(x^{k+1})) - 2\eta(x^k - x^*)^T\nabla E(x^k).\end{aligned}$$

It follows from the first part of Lemma 7 that

$$\begin{aligned}\|x^{k+1} - x^*\|_2^2 &\leqslant \|x^k - x^*\|_2^2 + \frac{2\eta}{2 - \rho\eta}(E(x^k) - E(x^{k+1})) - 2\eta(E(x^k) - E(x^*)) \\ &\leqslant \|x^k - x^*\|_2^2 - 2\eta\left(1 - \frac{1}{2 - \rho\eta}\right)(E(x^k) - E(x^*)) - \frac{2\eta}{2 - \rho\eta}(E(x^{k+1}) - E(x^*)).\end{aligned}$$

Since $1 - 1/(2 - \rho\eta) \geqslant 0$, $\|x^{k+1} - x^*\|_2 \leqslant \|x^k - x^*\|_2$ thus $\{x^k\}$ is bounded. Then there exists a subsequence $\{x^{k_i}\}$ such that $\lim_{i\to\infty} x^{k_i} = \hat{x}$. If follows that

$$\lim_{i\to\infty} \|\nabla E(x^{k_i})\|_2 = \|\nabla E(\hat{x})\|_2 = 0$$

since $\|\nabla E(x)\|_2$ is continuous. Therefore $\hat{x} \in \mathbb{X}^\dagger$. Finally, because $\|x^{k+1} - \hat{x}\|_2 \leqslant \|x^k - \hat{x}\|_2$, the sequence $\{x^{k_i}\}$ has only one accumulation point and thus $\lim_{k\to\infty} x^k = x^*$. □

Regarding the proof of Theorem 3, I should remark that, to ensure $\lim_{k\to\infty}\|\nabla E(x^k)\|_2 = 0$, the step size η can be chosen as $\eta < 2/\rho$. The tighter upper bound $1/\rho$ is used to ensure the boundedness of x^k. If the boundedness of x^k can be ensured by some other conditions, then a looser upperbound $2/\rho$ can be used. See the following result, which follows from Theorem 3 directly.

Corollary 2. *If $\mathbb{X}^\dagger$ is bounded, then the sequence $\{x^k\}$ generated by (11) is globally convergent to a point in $\mathbb{X}^\dagger$ by choosing $0 < \eta < 2/\rho$ where ρ is defined in Theorem 3.*

According to Theorem 3 and Corollary 2 we have the following results about the discrete-time system (11) for solving (3).

Corollary 3.

(1) *If (3) is solvable, i.e., $\mathbb{X}^* \neq \emptyset$, then the sequence $\{x^k\}$ generated by (11) is globally convergent to a point in $\mathbb{X}^*$ by choosing $0 < \eta \leqslant 3/(2\rho)$, where ρ is defined in Theorem 3.*
(2) *If $\mathbb{X}^* \neq \emptyset$ is bounded, then the sequence $\{x^k\}$ generated by (11) is globally convergent to a point in $\mathbb{X}^*$ by choosing $0 < \eta < 2/\rho$, where ρ is defined in Theorem 3.*

Proof. In both cases, there obviously exists at least one finite solution $x^* \in \mathbb{X}^*$. The first part of the corollary can be proved in the same way as proving Theorem 3 by using the second part instead of the first part of Lemma 7. The second part of the corollary is an immediate consequence of Corollary 2. □

Remark 2. If the LMI is absent in (3), Corollary 3 reduces to Theorem 2 and Corollary 1 in Xia et al. [14]. In other words, [14] does not ascertain the global convergence of the corresponding dynamic system when (3) without LMI is unsolvable; while this result can be ascertained according to Theorem 3 and Corollary 2.

According to Corollary 3, if the discrete-time dynamic system (11) does not converge to a solution of (3) with the step size $0 < \eta \leqslant 3/(2\rho)$, it implies that (3) admits no feasible solutions. If the step size is chosen smaller, such as $\eta \leqslant 1/\rho$, according to Theorem 3, the dynamic system can always converge to a point that corresponds to the least square error of (3), i.e., a minimum of E defined in (7).

Remark 3. In order to guarantee the global convergence of the dynamic system, the choice of the step size η in discrete-time dynamic system (11) should depend on the problem parameters A, C, N_i, while there is no such a restriction on the scaling factor μ in the continuous-time dynamic system (10). It is worth mentioning that, similar to Xia et al. [14], some globally convergent variants of (11) can be figured out with step size independent of the problem parameters. As there are no obvious advantages with such variants over (11) in terms of convergence conditions as well as convergence rates, those results are not discussed here. Interested readers may refer to Section 4 of Xia et al. [14] for details.

4. Simulation examples

Example 1. Consider a problem (3) with

$$A = \begin{pmatrix} 1 & 1 & 0 & 3 \\ -2 & 1 & 1 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ -10 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 2 & -3 \\ -2 & 1 & 0 & 2 \\ 0 & 0 & -1 & -3 \end{pmatrix}, \quad d = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \quad N_0 = \begin{pmatrix} -5 & 0 \\ 0 & -5 \end{pmatrix},$$
$$N_1 = \begin{pmatrix} 2 & -2 \\ -2 & 0 \end{pmatrix}, \quad N_2 = \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}, \quad N_3 = \begin{pmatrix} 0 & -6 \\ -6 & 2 \end{pmatrix}, \quad N_4 = \begin{pmatrix} 0 & -1 \\ -1 & 3 \end{pmatrix}.$$

I first used the continuous-time dynamic system (10) to solve the problem. The function "ode45" in MATLAB was adopted as the ODE solver. Experiments showed that (10) was always convergent to a solution with any choice of $\mu > 0$ from any initial point $x_0 \in \mathbb{R}^4$. For example, Fig. 2 displays the results of one of these experiments, where $\mu = 10^4$ and the trajectories converged to a solution $x^* = (3.4665, 6.1628, -1.1555, 0.3852)^T$. The eigenvalues of the resulting LMI, i.e., $N(x^*)$, are 20.429 and 0.000, which indicates that x^* is a nonstrict LMI solution.

Then, I used the discrete-time dynamic system (11) to solve the problem. According to Corollary 3, the step size should be chosen as $\eta = 3/(2\rho) = 0.012$. Simulations showed that with this η (11) was always convergent to a solution of the problem from any initial point $x_0 \in \mathbb{R}^4$. For example, Fig. 3 displays the trajectories of (11) (continuous lines) from a random initial point, which converged to a solution $x^* = (4.8279, 8.5829, -1.6093, 0.5364)^T$. It is worth mentioning that Corollary 3 just gives a sufficient condition for ensuring the convergence of the system. It is possible that the system is still convergent if $\eta > 3/(2\rho)$. For instance, the dashed lines in Fig. 3 depict the trajectories of (11) with $\eta = 0.07$ from the same initial point as

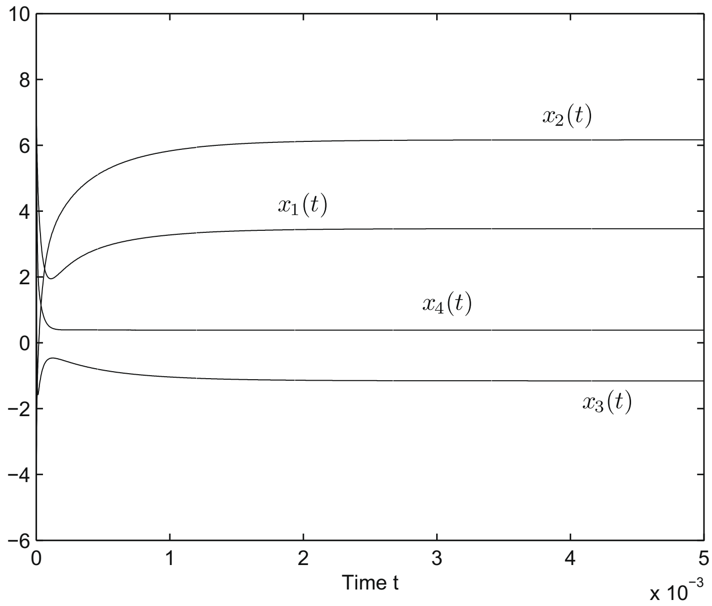

Fig. 2. Transient behavior of the continuous-time dynamic system (10) for solving the problem in Example 1.

that for the continuous lines. Clearly, the trajectories converged to the same solution x^*, but with a higher speed. However, when η was increased further, e.g., $\eta = 0.08$, the trajectories diverged to infinity.

Example 2. Consider solving the generalized Lyapunov inequality (2) with

$$A = \begin{pmatrix} -1 & 0 & 0 \\ 2 & -1 & 0 \\ 0 & 0 & -4 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 0 & -2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix}, \quad D = \begin{pmatrix} 3 & 0 & 2 \\ 0 & 3 & 0 \\ 2 & 0 & 1 \end{pmatrix}.$$

The strict LMI (2) can be converted to the following nonstrict LMI and then solved by using the proposed dynamic system methods,

$$BPA + A^T PB^T + D + \epsilon I \leqslant 0, \quad P - \epsilon I \geqslant 0, \tag{14}$$

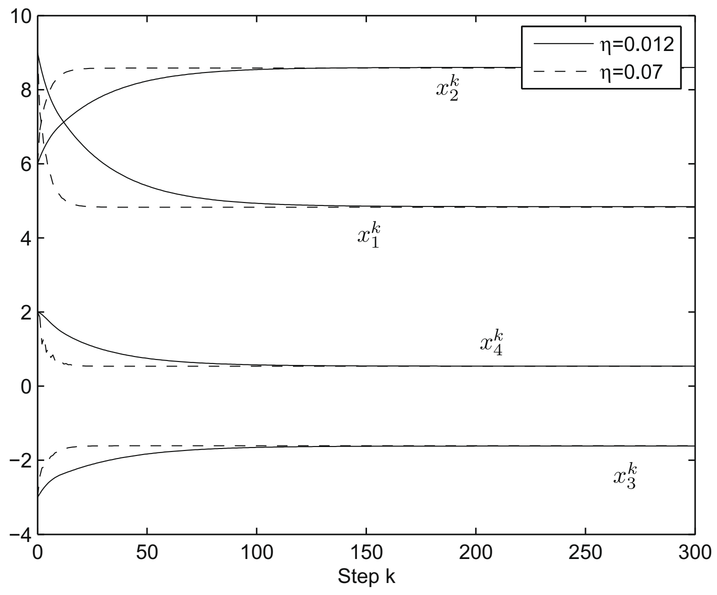

Fig. 3. Transient behavior of the discrete-time dynamic system (11) for solving the problem in Example 1 with different step sizes from the same initial point.

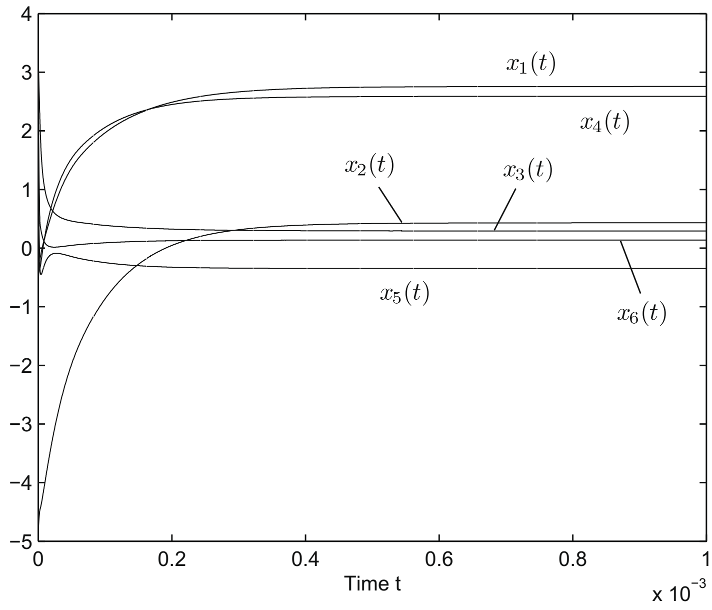

Fig. 4. Transient behavior of the continuous-time dynamic system (10) for solving the first problem in Example 2.

where I denotes the identity matrix and ϵ is a sufficiently small positive number. Based on Proposition 1, the new LMI can be easily converted into the canonical form (1). Note that because $P^T = P$, there are six independent variables in P, which can be represented by $x_i, i = 1, \ldots, 6$. I have simulated the continuous-time dynamic system (10) to solve the resulting nonstrict LMI with $\epsilon = 0.01$. All simulations showed that the system with any choice of μ was globally convergent to a solution of the LMI. For instance, Fig. 4 depicts the trajectories of the system with $\mu = 10^4$ from a random initial point, and the trajectories converged to $x^* = (2.7557, 0.4308, 0.2924, 2.5878, -0.3456, 0.1367)^T$, i.e.,

$$P^* = \begin{pmatrix} 2.7557 & 0.4308 & 0.2924 \\ 0.4308 & 2.5878 & -0.3456 \\ 0.2924 & -0.3456 & 0.1367 \end{pmatrix}.$$

It was calculated $\text{eig}(P^*) = \{0.0431, 2.3264, 3.1107\}$ and $\text{eig}(BP^*A + A^TPP^*B^T + D) = \{-0.8172, -0.0104, -0.0099\}$, which indicates that the LMI (1) is strictly feasible.

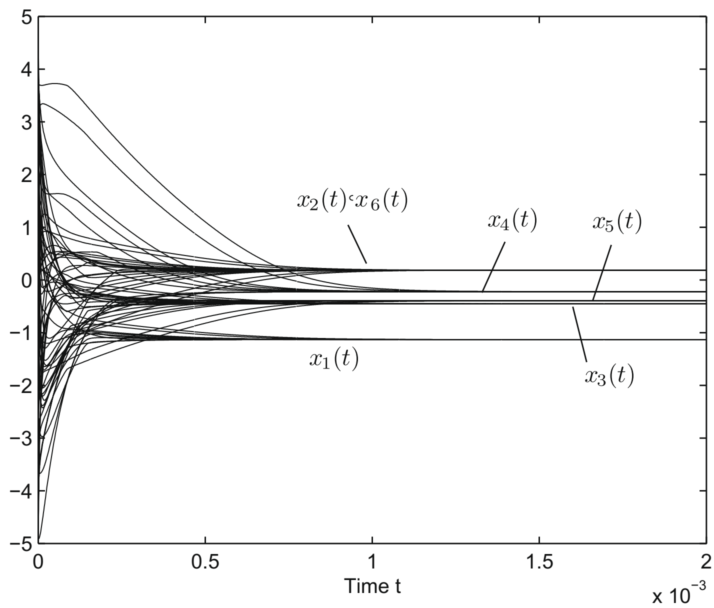

Fig. 5. Transient behavior of the continuous-time dynamic system (10) from 10 random initial points for solving the second problem in Example 2.

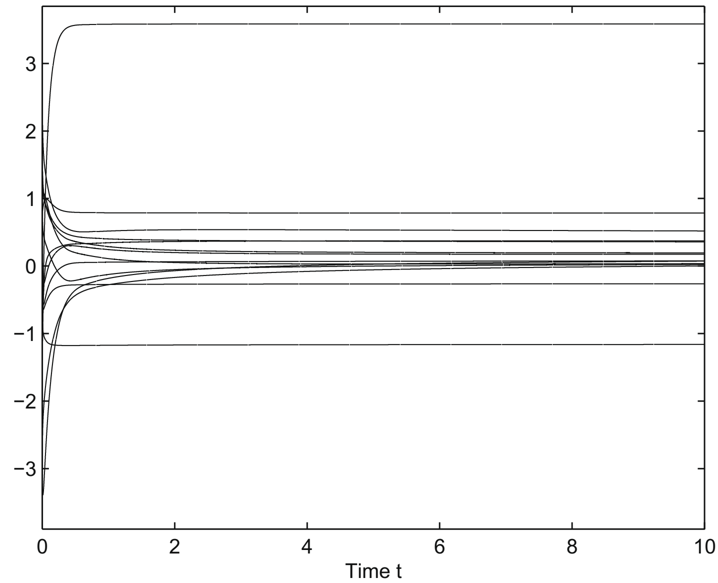

Fig. 6. Transient behavior of the continuous-time dynamic system (10) for solving the problem in Example 3.

Then, A was changed to the identity matrix, while B and D were kept the same as before. The continuous-time dynamic system (10) was simulated to solve the LMI (14) again. It was shown that the dynamic system always converged to a point which does not solve the LMI no matter how small ϵ was (even when $\epsilon = 0$). Fig. 5 depicts the transient behavior of the system in solving (14) with $\epsilon = 0$ from 10 random initial points in $\mathbb{R}^6$, where μ was set to 10^4. The trajectories converged to $x^* = (-1.1317, 0.1850, -0.4551, -0.2236, -0.3959, 0.1832)^T$, which is the unique minimum of the corresponding energy function $E(x)$ defined in (7), but is not a solution of the LMI. This implies that the LMI (2) does not have a solution.

Example 3. Finally, let us consider solving a maximum eigenvalue minimization problem [2] by using the proposed methods. Suppose that the symmetric matrix $W(x)$ depends affinely on $x \in \mathbb{R}^m : W(x) = W_0 + \sum_{i=1}^m W_i x_i$, where $W_i \in \mathbb{S}^n$. The problem of minimizing the maximum eigenvalue of the matrix $W(x)$ can be cast into a semidefinite programming problem

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & sI - W(x) \geqslant 0 \end{array} \tag{15}$$

with variables $s \in \mathbb{R}$ and $x \in \mathbb{R}^m$. According to the discussion in Section 2, the problem can be further cast into problem (3) with $m + 1 + n(n+1)/2$ variables. For example, let

$$W_0 = \begin{pmatrix} 5 & 0 & 3 & -1 \\ 0 & -5 & 2 & 0 \\ 3 & 2 & -1 & 4 \\ -1 & 0 & 4 & 0 \end{pmatrix}, \quad W_1 = \begin{pmatrix} -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad W_2 = \begin{pmatrix} 2 & -2 & 0 & 0 \\ -2 & -5 & 1 & 1 \\ 0 & 1 & 6 & 2 \\ 0 & 1 & 2 & -3 \end{pmatrix}, \quad W_3 = \begin{pmatrix} 1 & 0 & 2 & -5 \\ 0 & 2 & 0 & 0 \\ 2 & 0 & -2 & 0 \\ -5 & 0 & 0 & 5 \end{pmatrix}.$$

First, the continuous-time dynamic system was simulated to solve the problem. All simulations showed that the system was convergent to the unique solution $x^* = (0.7822, -1.1538, -0.2579)^T$ of (15). Fig. 6 demonstrates the trajectories of the $m + 1 + n(n+1)/2 = 14$ variables starting from a random initial point, which converged to x^*. The same solution was obtained by simulating the discrete-time dynamic system (11).

5. Concluding remarks

Two dynamic system methods are presented in the paper for solving linear matrix inequalities (LMIs) and linear vector equalities and inequalities. One is of continuous-time type and the other is of discrete-time type, both of which are advantageous for their simple numerical implementation characteristics and capability for handling nonstrict LMIs easily. A more promising potential of the methods is the recurrent neural network realization of them, hybridized with some high performance computing units for matrix factorization, while such units can be either digital equipments or some other neural networks. The global convergence and stability of both dynamic systems are analyzed rigorously, and then substantiated by several numerical examples.

However, the paper just lays some theoretical foundations and presents some preliminary simulation results. The value of this work is worth further investigation. For instance, based on Theorem 1, many efficient unconstrained optimization algorithms can be specialized to solve LMIs; but what are their characteristics? For another instance, based on the projection techniques utilized in the paper and the techniques in Refs. [15,17], it seems quite possible to design some dynamic system to solve nonlinear semidefinite programming problems. These are my future researches.

Acknowledgements

The work was supported by the National Natural Science Foundation of China under Grants 60805023, 60621062 and 60605003, National Key Foundation R&D Project under Grants 2003CB317007, 2004CB318108 and 2007CB311003, China Postdoctoral Science Foundation under Grants 20080430032 and 200801072, and Basic Research Foundation of Tsinghua National Laboratory for Information Science and Technology (TNList).

References

[1] S. Boyd, L.E. Ghaoui, E. Feron, V. Balakrishnan, Linear Matrix Inequalities in System and Control Theory, SIAM, Philadelphia, PA, 1994.
[2] L. Vandenberghe, S. Boyd, Semidefinite Programming, SIAM Review 38 (1) (1996) 49–65.
[3] L. Vandenberghe, V. Balakrishnan, Algorithms and software for LMI problems in control, IEEE Control Syst. Mag. 17 (1997) 89–95.
[4] Q. Liu, J. Cao, Y. Xia, A delayed neural network for solving linear projection equations and its analysis, IEEE Trans. Neural Network 16 (4) (2005) 834–843.
[5] W.L. Lu, T.P. Chen, Dynamical behaviors of delayed neural network systems with discontinuous activation functions, Neural Comput. 18 (3) (2006) 683–708.
[6] T. Li, L. Guo, C. Sun, C. Lin, Further results on delay-dependent stability criteria of neural networks with time-varying delays, IEEE Trans. Neural Network 19 (4) (2008) 726–730.
[7] T. Li, C. Sun, X. Zhao, C. Lin, LMI-based asymptotic stability analysis of neural networks with time-varying delays, Int. J. Neural Syst. 18 (3) (2008) 257–265.
[8] N.Z. Shor, Minimization Methods for Non-Differentiable Functions, Springer-Verlag, Berlin, 1985.
[9] A. Nemirovskii, P. Gahinet, The projective method for solving linear matrix inequalities, in: Proceedings of the American Control Conference, Baltlmore, Maryland, June 1994, pp. 840–844.
[10] Y. Nesterov, A. Nemirovsky, Interior-point polynomial methods in convex programming, Studies in Applied Mathematics, vol. 13, SIAM, Philadelphia, PA, 1994.
[11] F. Jarre, An interior-point method for minimizing the maximum eigenvalue of a linear combination of matrices, SIAM J. Control Optim. 31 (5) (1993) 1360–1377.
[12] J.J. Hopfield, D.W. Tank, Computing with neural circuits: a model, Science 233 (4764) (1986) 625–633.
[13] M.P. Kennedy, L.O. Chua, Neural networks for nonlinear programming, IEEE Trans. Circuits Syst. 35 (1988) 554–562.
[14] Y. Xia, J. Wang, D.L. Hung, Recurrent neural networks for solving linear inequalities and equations, IEEE Trans. Circuits Systems II 46 (4) (1999) 452–462.
[15] Y. Xia, J. Wang, Recurrent neural networks for solving nonlinear convex programs with linear constraints, IEEE Trans. Neural Network 16 (2) (2005) 379–386.
[16] X. Gao, L. Liao, L. Qi, A novel neural network for variational inequalities with linear and nonlinear constraints, IEEE Trans. Neural Network 16 (6) (2005) 1305–1317.
[17] X. Hu, J. Wang, Solving pseudomonotone variational inequalities and pseudoconvex optimization problems using the projection neural network, IEEE Trans. Neural Network 17 (6) (2006) 1487–1499.
[18] X. Hu, J. Wang, A recurrent neural network for solving a class of general variational inequalities, IEEE Trans. Syst. Man Cybern. B 37 (3) (2007) 528–539.
[19] X. Hu, J. Wang, An improved dual neural network for solving a class of quadratic programming problems and its *k*-winners-take-all application, IEEE Trans. Neural Network 19 (12) (2008) 2022–2031.
[20] X. Hu, B. Zhang, A new recurrent neural network for solving convex quadratic programming problems with an application to the *k*-winners-take-all problem, IEEE Trans. Neural Network 20 (4) (2009) 654–664.
[21] Q. Liu, J. Wang, A one-layer recurrent neural network with a discontinuous hard-limiting activation function for quadratic programming, IEEE Trans. Neural Network 19 (4) (2008) 558–570.
[22] Y. Zhang, S.S. Ge, T.H. Lee, A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators, IEEE Trans. Syst. Man Cybern. B 34 (5) (2004) 2126–2132.
[23] D.-Z. Feng, Z. Bao, X.-D. Zhang, A cross-associative neural network for SVD of nonsquared data matrix in signal processing, IEEE Trans. Neural Network 12 (5) (2001) 1215–1221.
[24] Z. Yi, Y. Fu, Neural networks based approach for computing eigenvectors and eigenvalues of symmetric matrix, Comput. Math. Appl. 47 (8–9) (2004) 1155–1164.
[25] Y. Liu, Z. You, L. Cao, A simple functional neural network for computing the largest and smallest eigenvalues and corresponding eigenvectors of a real symmetric matrix, Neurocomputing 67 (2005) 369–383.
[26] Y. Liu, Z. You, L. Cao, A functional neural network computing some eigenvalues and eigenvectors of a special real matrix, Neural Networks 18 (10) (2005) 1293–1300.
[27] Y. Liu, Z. You, L. Cao, A recurrent neural network computing the largest imaginary or real part of eigenvalues of real matrices, Comput. Math. Appl. 53 (1) (2007) 41–53.
[28] C.-L. Lin, C.-C. Lai, T.-H. Huang, A neural network for linear matrix inequality problems, IEEE Trans. Neural Network 11 (5) (2000) 1078–1092.
[29] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, 2004.
[30] D. Kinderlehrer, G. Stampcchia, An Introduction to Variational Inequalities and Their Applications, Academic, New York, 1980.
[31] A. Auslender, Optimisation: Méthodes Numériques, Masson, Paris, 1976.
[32] N. Higham, Computing a nearest symmetric positive semidefinite matrix, Linear Algebra Appl. 103 (1988) 103–118.