Noname manuscript No.
(will be inserted by the editor)

A data-driven quadratic stability condition and its application for stabilizing unknown nonlinear systems

Fan Zhang · Dirk Söffker

Received: date / Accepted: date

Abstract This paper proposes a data-driven stability criterion for quadratic stabilization of unknown nonlinear discrete-time systems. The novelty of this quadratic stability criterion lies in the direct use of the time series of system states, instead of using mathematical models. The data-driven stability criterion is utilized to design a control for stabilizing unknown nonlinear systems using recurrent neural networks. The effectiveness and the adaptability of the proposed approach is compared with the adaptive feedback linearization method with an example of stabilizing a nonlinear aeroelastic system.

Keywords Data-driven · Quadratic Lyapunov function · Quadratic stabilization · Unknown system · Recurrent neural network

1 Introduction

Quadratic stability and stabilization is an important topic in stability analysis and control of dynamical systems. A system is said to be quadratic stable if there exists a Quadratic Lyapunov Function (QLF) whose derivative (or first-order difference in discrete time) along the system trajectories is negative [3]. While in the beginning of 1990s the study of quadratic stability concentrated on systems with uncertainties [5,14,27,19,24], in the past decade this topic has

F. Zhang
Chair of Dynamics and Control, University of Duisburg-Essen, Duisburg, Germany
Tel.: +49-203-3793416
Fax: +49-203-3793027
E-mail: fan.zhang@uni-due.de

D. Söffker
Chair of Dynamics and Control, University of Duisburg-Essen, Duisburg, Germany
Tel.: +49-203-3793429
Fax: +49-203-3793027
E-mail: soeffker@uni-due.de

been more intensively investigated in switched systems [12,33,2,25] or fuzzy dynamical systems [15,9,18,20].

The main issue in the research of quadratic stability is the sufficient and necessary condition of the existence of a QLF. Nevertheless, the existing results about quadratic stability conditions, such as those mentioned above, are model-based. These approaches rely on the accuracy of the mathematical model describing system dynamics. If a mathematical model is difficult to be established precisely enough or unavailable, controllers that are designed based on model-based stability conditions may lead to bad control performance or even failure of stabilization.

Consider a system trajectory starting from the initial time to an arbitrary time instant. According to the definition of quadratic stability, a system is quadratic stable if a QLF exists for arbitrary trajectories. If it can be known under which condition a QLF exist for the concerned trajectory, the system can be controlled to be quadratic stable, as long as at any time instant the trajectory of system under control satisfy this condition of QLF existence. Under this circumstance, the mathematical system model is not necessarily known and the model-related stabilization problems mentioned above can be avoided.

Motivated by these concerns, this paper concentrates on the data-driven realization of quadratic stability judgment and the corresponding control design of stabilizing nonlinear discrete-time systems, i.e., establishing the data-driven condition for QLF existence of a single system trajectory and applying the data-driven condition for designing control of stabilization problem.

It should be remarked that two assumptions: i) all system states are measurable, and ii) the measurements are free of noise, have to be made at present stage of this contribution.

The paper is organized as follows: at first, the problem definition of quadratic stability and stabilization is introduced in section 2; secondly, the proposed data-driven quadratic stability criterion is presented in section 3; after that, the stabilization using the proposed stability condition is introduced in section 4; in section 5 a numerical example of stabilizing a nonlinear aeroelastic system is presented; finally, this contribution is concluded in the last section of the paper.

2 Problem definition

2.1 Data-driven quadratic stability

The discrete-time nonlinear system considered for stability analysis is described by

$$\boldsymbol{x}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k)), \tag{1}$$

where $\boldsymbol{f}(\cdot) : \Omega \to \mathbb{R}^n$ is a smooth mapping from a compact set $\Omega \subset \mathbb{R}^n$ into $\mathbb{R}^n$ and x the state vector defined in Ω. According to [3], the definition of quadratic stability for system (1) can be stated as

Definition 1 *The system (1) is quadratic stable if there exists a positive definite matrix* $\boldsymbol{P}$ *such that the first-order difference of the function* $\boldsymbol{V}(\boldsymbol{x}(k)) = \boldsymbol{x}(k)^T \boldsymbol{P}\boldsymbol{x}(k)$ *along the solution of system (1) satisfies*

$$\begin{aligned} \Delta V(\boldsymbol{x}(k)) &= V(\boldsymbol{x}(k+1)) - V(\boldsymbol{x}(k)) \\ &= V(\boldsymbol{f}(\boldsymbol{x}(k))) - V(\boldsymbol{x}(k)) < 0. \end{aligned}$$

Correspondingly, the function $V(\boldsymbol{x}(k))$ is called Quadratic Lyapunov Function (QLF). If in addition $\boldsymbol{P}$ is diagonal, $V(\boldsymbol{x}(k))$ is called Diagonal Quadratic Lyapunov Function (DQLF) and the system (1) is diagonally quadratic stable.

In the data-driven context, the existence of a QLF cannot be determined by using the analytical form of $\boldsymbol{f}(\boldsymbol{x})$ because it is assumed as unknown. Assume that the system (1) be fully observable and the system states be measured without noise. At the time instant $t = r$, the data set containing r consecutive measurements of system states can be denoted as

$$\mathcal{X}_r = \{\boldsymbol{x}(1), \ldots, \boldsymbol{x}(r)\} \ . \tag{2}$$

As mentioned in the introduction section, the data-driven quadratic stability judgment needs only to judge the stability of the currently running motion of the concerned system, i.e., the measured states of the running trajectory, but not the existence of the QLF for the global state space of the system. Therefore, the objective of this paper is defined as to determine the existence of a QLF for the complete system trajectory under initial condition $\boldsymbol{x}(1) = \boldsymbol{x}_0$. The system is judged as quadratic stable if and only if a QLF can be found based on the measured data.

2.2 Stabilization using data-driven stability condition

Consider the nonlinear discrete-time system

$$\boldsymbol{x}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k)), \tag{3}$$

where $\boldsymbol{x}$ denotes the state vector and $\boldsymbol{u}$ the control input. The quadratic stabilization problem related to system (3) is to design a control input

$$\boldsymbol{u}(k) = \boldsymbol{u}(\boldsymbol{x}(k), \boldsymbol{x}(k-1), \ldots, \boldsymbol{x}(k-l)), \tag{4}$$

where l is an integer and $l < k$, such that the origin $\boldsymbol{x} = \boldsymbol{0}$ in the state space of the system (3) with control input (4) is a uniformly asymptotic stable equilibrium point [13].

In this paper, the known information at an arbitrary instant $t = r$ is only the measurements of historical states, i.e., the vector set $\mathcal{X}_r$ defined in equation (2). The structural and physical information of the nonlinear function $\boldsymbol{f}(\cdot)$ described in (3) is assumed unknown. The objective is to determine the suitable control input $\boldsymbol{u}(k)$ online to stabilize the system (3), with the measured system trajectory being the only utilizable information.

3 Data-driven quadratic stability criterion

3.1 Geometrical preliminaries

Necessary geometric concepts used in this paper are briefly summarized in this subsection. Most of the geometrical definitions are taken from [6].

The convex hull for a vector set $\mathcal{C}$, denoted as $\mathbf{conv}\,\mathcal{C}$, is defined as the minimal convex set containing $\mathcal{C}$, i.e., $\mathbf{conv}\,\mathcal{C} = \{\Sigma_{i=1}^{m}\theta_i\boldsymbol{x}_i|\boldsymbol{x}_i \in \mathcal{C}, \theta_i \geq 0, \Sigma_{i=1}^{m}\theta_i = 1, i = 1, \dots, m\}$. A convex conic hull of a set $\mathcal{C}$ is the smallest convex cone for a vector set $\mathcal{C}$ defined as $\mathbf{cone}\,\mathcal{C} = \{\Sigma_{i=1}^{m}\theta_i\boldsymbol{x}_i|\boldsymbol{x}_i \in \mathcal{C}, \theta_i \geq 0, i = 1, \dots m\}$. The set $\mathbf{cone}\,\mathcal{C}^o$, which is defined as $\mathbf{cone}\,\mathcal{C}^o = \{\boldsymbol{y}|\boldsymbol{x}^T\boldsymbol{y} \leq 0, \text{for all } \boldsymbol{x} \in \mathbf{cone}\,\mathcal{C}\}$, is called the polar cone of $\mathbf{cone}\,\mathcal{C}$.

A set $\mathcal{C}$ is said to be a convex polyhedron if it can be written as $\mathbf{conv}\,\mathcal{C} = \{\boldsymbol{x}|\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}\}$ for some matrix $\boldsymbol{A}$ and vector $\boldsymbol{b}$. A set $\mathcal{C}$ is a polyhedral cone if it can be represented by the above form of a polyhedron with $\boldsymbol{b} = \boldsymbol{0}$, denoted as $\mathbf{cone}(\boldsymbol{A})$. A typical example of a polyhedral cone is the $\mathbb{R}_+^n$.

A polyhedral cone has two representation methods: the H-representation utilizing the form of the set of inequalities with respect to a matrix $\boldsymbol{A}$, denoted as $\mathbf{cone}(\boldsymbol{A})$, and the V-representation utilizing the conic combination of the vectors within a set $\mathcal{C}$, denoted as $\mathbf{cone}\,\mathcal{C}$. The Carathéodory theorem [7] shows that if the set $\mathcal{C}_p$ contains the extreme rays of the cone defined by $\mathcal{C}$, then the polyhedral cone $\mathbf{cone}\,\mathcal{C}_p$ is identical the polyhedral cone $\mathbf{cone}\,\mathcal{C}$.

3.2 The necessary and sufficient condition for one trajectory

Denote the vector obtained by left-multiplying an orthogonal matrix $\boldsymbol{\Phi}$ to the vector $\boldsymbol{x}(k)$ of the concerned trajectory as $\tilde{\boldsymbol{x}}(k)$, $\tilde{\boldsymbol{x}}(k) = \boldsymbol{\Phi}\boldsymbol{x}(k)$. Define a transformation for every vector $\tilde{\boldsymbol{x}}(k)$ as

$$\tilde{\boldsymbol{v}}(k) = \tilde{\boldsymbol{x}}(k+1) \circ \tilde{\boldsymbol{x}}(k+1) - \tilde{\boldsymbol{x}}(k) \circ \tilde{\boldsymbol{x}}(k)\,, \tag{5}$$

where $\tilde{\boldsymbol{v}}(k)$ represents the corresponding transformed vector and $\circ$ represents the calculation of Hadamard product [11] defined by

$$\boldsymbol{a} \circ \boldsymbol{b} = [a_j\, b_j],\ j = 1, \dots, n. \tag{6}$$

In (6), the symbols $\boldsymbol{a}$ and $\boldsymbol{b}$ represent two arbitrary n-dimensional vectors with a_j and b_j being their components, respectively. Unlike the inner products between two vectors, the calculation $\circ$ establish a manipulation from two vectors to a new vector.

Applying the proposed vector manipulation to all the elements within the data set $\mathcal{X}_r$ defined in equation (2) for $r = \infty$, a new vector set of $\tilde{\boldsymbol{v}}(k)$ can be obtained. Denote the complete vector set of $\tilde{\boldsymbol{v}}(k)$, $k = 1, \dots, \infty$, as $\tilde{\mathcal{V}}_\infty$, and the convex conic hull (the smallest convex cone) determined by $\tilde{\mathcal{V}}_\infty$ as $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty$. Correspondingly, the polar cone of $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty$, denoted as $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o$, can be represented as

$$\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o = \left\{\boldsymbol{y}|\tilde{\boldsymbol{v}}^T(k)\boldsymbol{y} \leq 0, \tilde{\boldsymbol{v}}(k) \in \mathbf{cone}\,\tilde{\mathcal{V}}_\infty,\, \boldsymbol{y} \in \mathbb{R}^n\right\}. \tag{7}$$

Consider the nonlinear discrete-time system (1) with an equilibrium point at the origin of its state space. The necessary and sufficient condition of the existence of a QLF for the considered system trajectory $\mathcal{X}_r$, $r=\infty$, can be given as the following theorem 1.

Theorem 1 *The trajectory $\mathcal{X}_r$, $r=\infty$, of the nonlinear discrete-time system (1) is quadratic stable, if and only if there exists an orthogonal matrix $\boldsymbol{\Phi}$ such that the following two conditions are satisfied:*

1. *The convex cone $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty$ is a proper cone.*
2. *The polar cone $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o$ defined in (7) satisfies*

$$\mathbf{int}\left(\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o\cap\mathbb{R}_+^n\right)\neq\varnothing\,, \tag{8}$$

where $\mathbf{int}\,(\cdot)$ represents the interior of a set and $\varnothing$ represents an empty set.

Proof To prove sufficiency, denote an arbitrary vector whose entries are real positive numbers as $\boldsymbol{d}$, $\boldsymbol{d}\in\mathbb{R}_+^n$, belongs to the set $\mathbf{int}\left(\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o\cap\mathbb{R}_+^n\right)$. If the convex cone $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty$ is a proper cone, the polar cone $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o$ is also proper and has non-empty interior, which indicates that $\boldsymbol{d}$ also belongs to $\mathbf{int}\,(\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o)$.

According to the definition of the polar cone $\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o$ in (7), it can be obtained that for all the transformed vectors in the data set $\tilde{\mathcal{V}}_\infty$, the following condition holds

$$\langle\boldsymbol{d},\,\tilde{\boldsymbol{v}}(k)\rangle\leq 0,\ \boldsymbol{d}\in\mathbb{R}_+^n,\ k=1,\ldots,\infty, \tag{9}$$

where $\langle\cdot\rangle$ represents the inner product. Because $\boldsymbol{d}\in\mathbf{int}\,(\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o)$, where $\mathbf{int}\,(\mathbf{cone}\,\tilde{\mathcal{V}}_\infty^o)$ is a convex set, the above inequality can be specified as

$$\langle\boldsymbol{d},\,\tilde{\boldsymbol{v}}(k)\rangle< 0,\ \boldsymbol{d}\in\mathbb{R}_+^n,\ k=1,\ldots,\infty, \tag{10}$$

Using the definition of $\tilde{\boldsymbol{v}}(k)$ in (5), the inner product between $\tilde{\boldsymbol{v}}(k)$ and $\boldsymbol{d}$ in (10) can be represented as

$$\langle\tilde{\boldsymbol{v}}(k),\boldsymbol{d}\rangle=\boldsymbol{d}^T\,\left(\tilde{\boldsymbol{x}}(k+1)\circ\tilde{\boldsymbol{x}}(k+1)-\tilde{\boldsymbol{x}}(k)\circ\tilde{\boldsymbol{x}}(k)\right). \tag{11}$$

Define a diagonal matrix $\boldsymbol{D}$ as $\boldsymbol{D}=\mathbf{diag}\,[\boldsymbol{d}]$. Obviously $\boldsymbol{D}$ is positive definite because it is diagonal and its diagonal elements vector $\boldsymbol{d}$ belongs to $\mathbb{R}_+^n$. With notation that

$$\begin{aligned}\tilde{\boldsymbol{x}}(k+1)\circ\tilde{\boldsymbol{x}}(k+1)&=\mathbf{diag}\,[\tilde{\boldsymbol{x}}(k+1)]\tilde{\boldsymbol{x}}(k+1)\,,\\ \boldsymbol{d}^T\mathbf{diag}[\tilde{\boldsymbol{x}}(k+1)]&=\tilde{\boldsymbol{x}}^T(k+1)\,\mathbf{diag}[\boldsymbol{d}]\,,\end{aligned} \tag{12}$$

and the similar relations for $\boldsymbol{d}$ and $\tilde{\boldsymbol{x}}(k)$, one can obtain the following equation by substituting (11) and (12) into the inequality (10), as

$$\begin{aligned}\langle\tilde{\boldsymbol{v}}(k),\boldsymbol{d}\rangle&=\tilde{\boldsymbol{x}}^T(k+1)\,\mathbf{diag}\,[\boldsymbol{d}]\,\tilde{\boldsymbol{x}}(k+1)\\ &\quad-\tilde{\boldsymbol{x}}^T(k)\,\mathbf{diag}\,[\boldsymbol{d}]\,\tilde{\boldsymbol{x}}(k)\\ &=\tilde{\boldsymbol{x}}^T(k+1)\,\boldsymbol{D}\,\tilde{\boldsymbol{x}}(k+1)\\ &\quad-\tilde{\boldsymbol{x}}^T(k)\,\boldsymbol{D}\,\tilde{\boldsymbol{x}}(k)<0\,.\end{aligned} \tag{13}$$

Because $\tilde{\boldsymbol{x}}(k)$ is defined as $\tilde{\boldsymbol{x}}(k) = \boldsymbol{\Phi}\boldsymbol{x}(k)$, the equation (13) can be represented as

$$\langle \tilde{\boldsymbol{v}}(k), \boldsymbol{d} \rangle = \boldsymbol{x}^T(k+1)\,\boldsymbol{Q}\,\boldsymbol{x}(k+1) - \boldsymbol{x}^T(k)\,\boldsymbol{Q}\,\boldsymbol{x}(k) < 0, \tag{14}$$

where the matrix $\boldsymbol{Q}$ is defined as $\boldsymbol{Q} = \boldsymbol{\Phi}^T\boldsymbol{D}\boldsymbol{\Phi}$. Because $\boldsymbol{D}$ is a positive definite diagonal matrix and $\boldsymbol{\Phi}$ is an orthogonal matrix, the matrix $\boldsymbol{Q}$ is also positive definite. Therefore, according to the definition of QLF it can be seen that the function $V(\boldsymbol{x}(k)) = \boldsymbol{x}^T(k)\,\boldsymbol{Q}\,\boldsymbol{x}(k)$ is a QLF for the concerned trajectory of the nonlinear discrete-time system (1), because $\boldsymbol{Q}$ is a positive definite matrix and $\Delta V(\boldsymbol{x}(k)) = \boldsymbol{x}(k+1)^T\,\boldsymbol{Q}\,\boldsymbol{x}(k+1) - \boldsymbol{x}(k)^T\,\boldsymbol{Q}\,\boldsymbol{x}(k) \leq 0$. This proves the sufficiency of the proposed theorem.

To prove the necessity, suppose there exist a quadratic Lyapunov function for the concerned trajectory, denoted as $V_d(\boldsymbol{x}(k)) = \boldsymbol{x}(k)^T\,\hat{\boldsymbol{Q}}\,\boldsymbol{x}(k)$. The matrix $\hat{\boldsymbol{Q}}$ can be decomposed as

$$\hat{\boldsymbol{Q}} = \hat{\boldsymbol{\Phi}}^T \mathbf{diag}\,[\hat{\boldsymbol{d}}]\hat{\boldsymbol{\Phi}}, \tag{15}$$

with $\hat{\boldsymbol{d}} \in \mathbb{R}^n_+$ being the eigenvalues of $\hat{\boldsymbol{Q}}$ and $\hat{\boldsymbol{\Phi}}$ the eigenvector matrix. Clearly $\mathbf{diag}\,[\hat{\boldsymbol{d}}]$ is a positive diagonal matrix and $\hat{\boldsymbol{\Phi}}$ is an orthogonal matrix, because $\hat{\boldsymbol{Q}}$ is positive definite.

Because $\boldsymbol{x}(k{+}1)^T\,\hat{\boldsymbol{Q}}\,\boldsymbol{x}(k{+}1) = \hat{\boldsymbol{d}}^T\mathbf{diag}\,[\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k{+}1)]\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k{+}1)$ and $\boldsymbol{x}(k)^T\,\hat{\boldsymbol{D}}\,\boldsymbol{x}(k) = \hat{\boldsymbol{d}}^T\mathbf{diag}\,[\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k)]\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k)$, the difference of $V_d(\boldsymbol{x}(k))$ can be expressed as

$$\begin{aligned}\Delta V_d(\boldsymbol{x}(k)) &= \boldsymbol{x}(k+1)^T\,\tilde{\boldsymbol{D}}\,\boldsymbol{x}(k+1) - \boldsymbol{x}(k)^T\,\tilde{\boldsymbol{D}}\,\boldsymbol{x}(k)\\ &= \hat{\boldsymbol{d}}^T\,(\mathbf{diag}\,[\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k+1)]\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k+1)\\ &\quad -\mathbf{diag}\,[\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k)]\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k))\\ &= \hat{\boldsymbol{d}}^T\,\boldsymbol{v}(k) < 0,\end{aligned} \tag{16}$$

where

$$\begin{aligned}\boldsymbol{v}(k) &= \mathbf{diag}\,[\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k+1)]\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k+1)\\ &\quad -\mathbf{diag}\,[\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k)]\hat{\boldsymbol{\Phi}}\boldsymbol{x}(k)\\ &= \tilde{\boldsymbol{x}}(k+1)\circ\tilde{\boldsymbol{x}}(k+1) - \tilde{\boldsymbol{x}}(k)\circ\tilde{\boldsymbol{x}}(k),\end{aligned} \tag{17}$$

with $\tilde{\boldsymbol{x}}(k) = \hat{\boldsymbol{\Phi}}\boldsymbol{x}(k)$. It is shown in equation (16) that the inner product of the vector $\boldsymbol{v}(k)$ with a vector $\hat{\boldsymbol{d}}$, $\hat{\boldsymbol{d}} \in \mathbb{R}^n_+$, is always less than zero. Due to this fact, the interior of the polar cone defined in (7) and constructed with the orthogonal matrix $\hat{\boldsymbol{\Phi}}$ is not empty, which indicates that the convex cone constructed by $\boldsymbol{v}(k)$, $k = 1, \ldots, \infty$ is also proper. Furthermore, because $\hat{\boldsymbol{d}} \in \mathbb{R}^n_+$, the polar cone has common elements with the real positive space $\mathbb{R}^n_+$.

In [8], it is shown that the complete set of the matrix $\boldsymbol{P}$ in QLF can be mapped surjectively from the special orthogonal group $SO(n,\mathbb{R})$ and the conventional topology of $\mathbb{R}^n_+$. This mapping can be defined as

$$(\boldsymbol{\Phi}, \boldsymbol{d}) \mapsto \boldsymbol{P} : \boldsymbol{P} = \boldsymbol{\Phi}^T\mathbf{diag}\,[\boldsymbol{d}]\boldsymbol{\Phi}, \tag{18}$$

where $\boldsymbol{\Phi}$ is an orthogonal matrix in $SO(n, \mathbb{R})$ and $\boldsymbol{d}$ is a real vector in $\mathbb{R}^n_+$. Because the mapping (18) is surjective, which is proven in [8], it can be concluded that no QLF exists if no element over the complete set $SO(n, \mathbb{R}) \times \mathbb{R}^n_+$ can be found and to construct a QLF, and vice versa, which also is consistent with the sufficiency and necessity of theorem 1.

Therefore, theorem 1 shows that the existence of a QLF can be determined by searching through the special orthogonal group $SO(n, \mathbb{R})$ and the conventional topology of $\mathbb{R}^n_+$. It does not require explicitly an analytical form of the nonlinear function $\boldsymbol{f}(\cdot)$ in system (1), but the complete time history of system state vectors. This fact makes it possible to apply the above theorem in the data-driven context to judge quadratic stability.

3.3 Stability condition for finite time measurements

At the time instant $t = r$, every system vector $\boldsymbol{x}(k) \in \mathbb{R}^n$, $k = 1, \dots, r-1$, can be transformed with one certain orthogonal matrix $\mathbf{\Phi}$ into a new vector $\tilde{\boldsymbol{v}}(k)$ with use of the mapping defined in equation (5). Correspondingly, these transformed vectors determine a new vector set, denoted as $\tilde{\mathcal{V}}_{r-1} = \{\tilde{\boldsymbol{v}}(k)\}$, $k = 1, \dots, r-1$.

It should be noted that the vector set $\tilde{\mathcal{V}}_{r-1}$ is different from the set $\tilde{\mathcal{V}}_\infty$ in theorem 1. The reason is that the set $\tilde{\mathcal{V}}$ is obtained by transforming *all* the states vectors $x(k)$ within the measured system states, but due to the finiteness of the measured data, the vector set $\mathcal{X}_r$ is only a subset of Ω where the nonlinear mapping $\boldsymbol{f}(\cdot)$ is defined. Thus $\tilde{\mathcal{V}}_{r-1} = \tilde{\mathcal{V}}$ is true only if $r \to \infty$.

The polar cone $\textbf{cone}\, \tilde{\mathcal{V}}^o_{r-1}$ can be determined by substituting $\tilde{\mathcal{V}}_\infty$ with $\tilde{\mathcal{V}}_{r-1}$ into equation (7), as

$$\textbf{cone}\, \tilde{\mathcal{V}}^o_{r-1} = \left\{ \boldsymbol{y} | \tilde{\boldsymbol{v}}^T \boldsymbol{y} \leq 0, \tilde{\boldsymbol{v}} \in \textbf{cone}\, \tilde{\mathcal{V}}_{r-1},\, \boldsymbol{y} \in \mathbb{R}^n \right\}. \tag{19}$$

By examining (19), it can be seen that $\textbf{cone}\, \tilde{\mathcal{V}}^o_{r-1}$ is a convex cone by adding an inequality constraint $-\tilde{\boldsymbol{v}}(r-1)^T \boldsymbol{d} > 0$ to $\textbf{cone}\, \tilde{\mathcal{V}}^o_{r-2}$, which implies that

$$\textbf{cone}\, \tilde{\mathcal{V}}^o_{r-1} = \bigcap_{l=1}^{r} \textbf{cone}\, \tilde{\mathcal{V}}^o_{l-1}\,. \tag{20}$$

Substituting equation (20) into the stability condition (8) in theorem (1), it can be obtained that

$$\begin{aligned} \textbf{int}\left(\textbf{cone}\, \tilde{\mathcal{V}}^o_\infty \cap \mathbb{R}^n_+\right) &= \textbf{int}\left((\bigcap_{l=1}^{\infty} \textbf{cone}\, \tilde{\mathcal{V}}^o_{l-1}) \cap \mathbb{R}^n_+ \right) \\ &= \bigcap_{l=1}^{\infty} \left(\textbf{int}\,(\textbf{cone}\, \tilde{\mathcal{V}}^o_{l-1} \cap \mathbb{R}^n_+) \right) \\ &\neq \varnothing\,. \end{aligned} \tag{21}$$

It is shown in equation (21) that if $\mathbf{cone}\,\tilde{\mathcal{V}}^o_{r-1} \cap \mathbb{R}^n_+ \neq \varnothing$ at $t = r$, the intersections between $\mathbb{R}^n_+$ and any of the cones $\mathbf{cone}\,\tilde{\mathcal{V}}^o_l$ formulated at former time instants $1 \le l < r-1$, are inherently nonempty. This fact implies that if the condition

$$\mathbf{int}\left(\mathbf{cone}\,\tilde{\mathcal{V}}^o_{l-1} \cap \mathbb{R}^n_+\right) \neq \varnothing \tag{22}$$

holds at every time instant, which guarantees that $\mathbf{cone}\,\tilde{\mathcal{V}}^o_{r-1} \neq \varnothing$, the concerned trajectory is quadratic stable, and vice versa.

Therefore, the quadratic stability of the concerned trajectory can be judged by checking the condition (22), the constructed convex cone is proper at every time instant. Hence, theorem 1 can be reformulated for practical implementation as

Theorem 2 *The concerned trajectory of nonlinear discrete-time system (1) is quadratic stable if and only if there exists an orthogonal matrix $\boldsymbol{\Phi}$ such that at every time instant $t = r$, $r = 1, \ldots, \infty$, the following two conditions can be satisfied:*

1. *the convex cone $\mathbf{cone}\,\tilde{\mathcal{V}}_{r-1}$ constructed by the data set $\mathcal{X}_r$ and the matrix $\boldsymbol{\Phi}$ is proper,*
2. *the polar cone $\mathbf{cone}\,\tilde{\mathcal{V}}^o_{r-1}$ of $\mathbf{cone}\,\tilde{\mathcal{V}}_{r-1}$ follows the relationship*

$$\mathbf{cone}\,\tilde{\mathcal{V}}^o_{r-1} \cap \mathbb{R}^n_+ \neq \varnothing\,. \tag{23}$$

Theorem 2 states that if the intersection between the set $\mathbf{cone}\,\tilde{\mathcal{V}}^o_{r-1}$ and $\mathbb{R}^n_+$ is not empty at every time instant, the system is quadratic stable. Furthermore, letting $\tilde{\boldsymbol{d}}$ be any vector located within $\mathbf{cone}\,\tilde{\mathcal{V}}^o_{r-1} \cap \mathbb{R}^n_+$, $r = \infty$, the QLF for this system can be expressed as

$$V(\boldsymbol{x}) = \boldsymbol{x}^{\boldsymbol{T}} \boldsymbol{\Phi}\, \mathbf{diag}\,[\tilde{\boldsymbol{d}}]\, \boldsymbol{\Phi}^T \boldsymbol{x}\,. \tag{24}$$

The condition (23) must be satisfied at every time instant for a quadratic stable system. Thus, to give a correct stability judgment to the considered trajectory, this criterion has to be implemented online and checked at every time instant. If at any time instant the constructed cone is not proper or condition (22) cannot be satisfied, the concerned trajectory can be judged as not quadratic stable.

As for numerical implementation, the stability condition (23) can be judged by solving a max-min problem [32]. If the optimized value of the max-min problem is greater than zero, the interior of the intersection is not empty and shows that there exists at least one QLF for the concerned trajectory of the system (1) at the time instant $t = r$. If the optimized value of the max-min problem is positive at every time instant, the concerned trajectory is globally quadratic stable; if at any time instant the optimized value is not positive, then the concerned trajectory is not quadratic stable.

It should be remarked that the stability judgment of the proposed algorithm is only necessary and sufficient with respect to the considered trajectory, but not to the whole system. Nevertheless, the proposed algorithm can still be used in control problem, if the control design can be realized online when only the current running trajectory needs to be considered.

For the purpose of control, if a controller can always drive the closed-loop system to satisfy the stability condition under arbitrary initial conditions, all the trajectories of the controlled system starting from this neighborhood converges to the equilibrium. This means all the perturbed motions starting from this neighborhood shall converge to the considered equilibrium. In this case the controlled system shall possess a (both sufficiently and necessarily) stable equilibrium and thereby be stabilized.

In the next section, a preliminary design of control using this stability criterion is discussed in the problem of stabilizing unknown nonlinear systems.

4 Application in stabilizing unknown systems

The control input $\boldsymbol{u}(k)$ is considered as a function of $x(k)$, as shown in equation (4). Therefore, the system (3) under control of $\boldsymbol{u}(k)$ can be expressed as the same form of (3). Correspondingly, the stability of controlled system can be discussed by using the proposed data-driven stability criterion.

The stabilization problem defined in section 2 can be reformulated as to find a control input $\boldsymbol{u}(k)$ such that the complete considered trajectory of closed-loop system

$$\mathcal{X}_\infty = \{\boldsymbol{x}(k), \;\; k = 1, \dots, \infty\}, \tag{25}$$

satisfies the condition (8) in theorem 1.

Specific to an arbitrary time instant $t = r$, it means that to find the value of $\boldsymbol{u}(r)$ such that the trajectory $\mathcal{X}_{r+1}$ satisfies the condition (23) in theorem 2, with $\mathcal{X}_{r+1}$ being defined as

$$\mathcal{X}_{r+1} = \mathcal{X}_r \cup \{\boldsymbol{x}(r+1)\} = \{\boldsymbol{x}(1), \dots, \boldsymbol{x}(r), \boldsymbol{x}(r+1)\}, \tag{26}$$

where $\boldsymbol{x}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k))$ and $\mathcal{X}_r$ is defined in 2.

Due to the fact the nonlinear function $\boldsymbol{f}(\cdot)$ in system dynamics is unknown, the states $\boldsymbol{x}(r+1)$ in (26) cannot be calculated directly with use of system equation at an arbitrary time instant r, even if the value of control input $u(r)$ is known. Consequentially, the vector set $\mathcal{X}_{r+1}$ cannot be established and the stability condition (23) cannot be judged. This problem can be solved by using the online system identification techniques using Recurrent Neural Network (RNN).

4.1 Solving the stabilization problem using RNN

The RNN system identification is a well-developed online system identification method [26,10]. Denote the identified system dynamics by RNN as

$$\hat{\boldsymbol{x}}(k+1) = \hat{\boldsymbol{f}}(\boldsymbol{x}(k), \boldsymbol{u}(k)), \tag{27}$$

where $\hat{\boldsymbol{f}}(\cdot)$ represents the identified dynamics of $\hat{\boldsymbol{f}}(\cdot)$ in (3) and $\hat{\boldsymbol{x}}(k+1)$ represents the one-step prediction of the future states of the plant at the time instant $t = k$. With suitable training algorithms, the RNN can make a one-step prediction of system states online and the error of one-step prediction by RNN is incrementally precise with time. This means that

$$\lim_{k\to\infty} \parallel \boldsymbol{e}_m(k+1) \parallel \to \boldsymbol{0}, \tag{28}$$

where

$$\boldsymbol{e}_m(k+1) = \hat{\boldsymbol{x}}(k+1) - \boldsymbol{x}(k+1), \tag{29}$$

with $\hat{\boldsymbol{x}}(k+1)$ being the one-step prediction at time instant $t = k$ and $\boldsymbol{x}(k+1)$ the true system states at $t = k+1$.

The equation (28) indicates that if the complete time series of the predicted states, denoted as

$$\hat{\mathcal{X}}_\infty = \{\hat{\boldsymbol{x}}(k),\ \ k = 1, \ldots, \infty\}, \tag{30}$$

converges to origin, the trajectory of true system states $\mathcal{X}_\infty$ defined in (25) also converges to the origin, because the error between these two time sequences goes to zero with time.

Correspondingly, specific to every arbitrary time instant $t = r$, if the vector set $\hat{\mathcal{X}}_{r+1}$, defined as

$$\hat{\mathcal{X}}_{r+1} = \hat{\mathcal{X}}_r \cup \{\hat{\boldsymbol{x}}(r+1)\} = \{\hat{\boldsymbol{x}}(1), \ldots, \hat{\boldsymbol{x}}(r), \hat{\boldsymbol{x}}(r+1)\}\ , \tag{31}$$

satisfies the condition (23) in theorem 2, the vector set $\mathcal{X}_{r+1}$ defined in (26) is also stable in the sense of theorem 2.

This fact shows that if the trajectory of predicted system states is stable, the trajectory of true system states is also stable. Therefore, the task of finding $\boldsymbol{u}(r)$ at $t = r$ such that the vector set $\mathcal{X}_{r+1}$ satisfies (23) is identical to find the $\boldsymbol{u}(r)$ that let vector set $\hat{\mathcal{X}}_{r+1}$ satisfy condition (23).

The function $\hat{\boldsymbol{f}}(\cdot)$ the identified dynamics (27) can be obtained by training the RNN at an arbitrary time instant $t = r$. As a result, the prediction $\hat{\boldsymbol{x}}(r+1)$ can be obtained if the control input $\boldsymbol{u}(r)$ is known. If a suitable control input $\boldsymbol{u}(r)$ can be found at every time instant $t = r$, for $r = 1, \ldots, \infty$, such that the data set $\hat{\mathcal{X}}_{r+1}$, including the predicted states $\hat{\boldsymbol{x}}(r+1)$, satisfies the geometrical condition (23), the data set $\hat{\mathcal{X}}_{r+1}$ is stable in the sense of theorem 2 and this input $\boldsymbol{u}(r)$ can also stabilize the system to be controlled.

4.2 Candidates of feasible control inputs

Denote the vector set obtained by applying the transformation defined in (5) to $\hat{\boldsymbol{x}}(k)$, $k = 1, \ldots, r$, as $\hat{\mathcal{V}}_{r-1}$. The elements of $\hat{\mathcal{V}}_{r-1}$ are denoted as $\hat{\boldsymbol{v}}(k)$, $k = 1, \ldots, r-1$,

Consider the problem of determining the control input $u(r)$ at an arbitrary time instant $t = r$ when the historical predicted states $\hat{\boldsymbol{x}}(k)$, $k = 1, \ldots, r$, satisfy the proposed stability condition (23). This means that the following fact

$$\mathbf{int}\left(\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^{o}\cap\mathbb{R}_{+}^{n}\right)\neq\varnothing, \tag{32}$$

is true, where the superscript $(\cdot)^o$ denotes the polar cone and the symbol $\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}$ represents the convex conic hull of $\hat{\mathcal{V}}_{r-1}$.

The prediction $\hat{\boldsymbol{x}}(k+1)$ can be obtained by substituting $\boldsymbol{u}(r)$ into the identified system (27). A new transformed vector $\hat{\boldsymbol{v}}(r)$ can be obtained by applying the transformation (5) of $\hat{\boldsymbol{x}}(k)$ and $\hat{\boldsymbol{x}}(k+1)$. Defining the set $\hat{\mathcal{V}}_r$ as

$$\hat{\mathcal{V}}_r=\{\hat{\boldsymbol{v}}(r)\}\cup\hat{\mathcal{V}}_{r-1}, \tag{33}$$

the remaining problem is how to determine the value of $\boldsymbol{u}(r)$, so that the condition

$$\mathbf{cone}\,\hat{\mathcal{V}}_{r}^{o}\cap\mathbb{R}_{+}^{n}\neq\varnothing, \tag{34}$$

can be fulfilled.

Assume the condition (34) is true. It can be obtained from (33) that

$$\mathbf{cone}\,\hat{\mathcal{V}}_r=\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}\cup\mathbf{cone}\,\{\hat{v}(r)\}, \tag{35}$$

and correspondingly

$$\mathbf{cone}\,\hat{\mathcal{V}}_r^o=\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^o\cap\mathbf{cone}\,\{\hat{v}(r)\}^o, \tag{36}$$

where $\{\hat{\boldsymbol{v}}(r)\}$ represents the vector set containing only the vector $\hat{\boldsymbol{v}}(r)$.

Substituting the equation above into (34), it can be obtained that

$$\mathbf{int}\left(\mathbf{cone}\,\{\hat{v}(r)\}^o\cap\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^o\cap\mathbb{R}_{+}^{n}\right)\neq\varnothing. \tag{37}$$

The equation (32) shows that the polyhedral cone $\hat{\mathcal{V}}_{r-1}^o\cap\mathbb{R}_+^n$ is not empty. According to the computational geometrical theory, the equation (37) is true if the vector $\hat{\boldsymbol{v}}(r)$ is not located in the dual cone of $\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^o\cap\mathbb{R}_+^n$, which contains all the vectors having positive inner products with elements of $\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^o\cap\mathbb{R}_+^n$.

In other words, the condition (34) is true if the vector $\hat{\boldsymbol{v}}(r)$ satisfies

$$\hat{\boldsymbol{v}}(r)\in\Lambda, \tag{38}$$

where Λ is the complementary set of the dual cone of $\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^o\cap\mathbb{R}_+^n$, denoted as

$$\Lambda=\mathbb{R}^n\backslash(\mathbf{cone}\,\hat{\mathcal{V}}_{r-1}^o\cap\mathbb{R}_+^n)^*. \tag{39}$$

If the condition (34) is true, the trajectory of predicted system states $\hat{\boldsymbol{x}}(k)$, $k=1,\,...\,,\,r,\,r+1$ is stable in the sense of theorem 2. As a result, the feasible values of control input $\boldsymbol{u}(r)$ which can guarantee stability at $t=r$ are the ones which can make the vector $\hat{\boldsymbol{v}}(r)$ located in the set Λ defined in (39).

```
input      : current system state x(k);
             searching region Ξ;
             identified plant dynamics at t = k
             x̂(k+1) = f̂(x(k), u(k));
             data set X̂_k = {x̂(0), ..., x̂(k)};
output     : feedback gain matrix K;
initialize: set value i ← 0, j ← 0;
             set value J ← ∞;
             set value n, number of elements of Ξ
while i ≤ n do
    set value K ← K_i with K_i ∈ Ξ;
    calculate u_i(k) = −K x(k);
    calculate x̂_i(k+1) = f̂(x(k), u_i(k));
    for j ← 1 to k do
        calculate X̂_{k+1} with x̂(k+1);
        calculate ṽ(j) ;                      // according to the equation (5)
    end
    calculate Λ = R^n \ (cone Ṽ^o_{r−1} ∩ R^n_+)* ;   // feasible region for ṽ(j)
    if ṽ(r) ∈ Λ then
        calculate J_i;
        if J_i < J then
            calculate J = J_i
        end
        calculate i = i + 1;
    else
        calculate i = i + 1;
    end
end
```

Algorithm 1: Solving optimization task given in (40)

4.3 Choosing suitable control input from candidates

The discussion about feasible control inputs shows that the control input $\boldsymbol{u}(k)$ should be chosen from the ones fulfilling (38). In this paper, the problem of seeking $\boldsymbol{u}(k)$ at one certain time instant is solved by solving an optimization problem, as

$$\begin{aligned} \min_{\boldsymbol{K}}.\ & J \\ s.t.\ & \boldsymbol{u}(k) = -\boldsymbol{K}(k)\boldsymbol{x}(k), \\ & \tilde{\boldsymbol{v}}(r) \in \mathbb{R}^n \backslash (\mathbf{cone}\, \tilde{\mathcal{V}}^o_{r-1} \cap \mathbb{R}^n_+)^*, \end{aligned} \tag{40}$$

where J is a suitably chosen performance measure that can be evaluated online with respect to the system states. For example, this measure can be chosen as

$$J = \int_{t_0}^{t_1} \boldsymbol{x}^T \boldsymbol{E} \boldsymbol{x} + \boldsymbol{u}^T \boldsymbol{F} \boldsymbol{u}\, \mathrm{d}t, \tag{41}$$

where t_0 and t_1 are the starting time and the final time of the evaluation, $\boldsymbol{E}$ and $\boldsymbol{F}$ are positive definite matrices, which has been used in the classic linear optimal control as one kind of representation of the input and state energy.

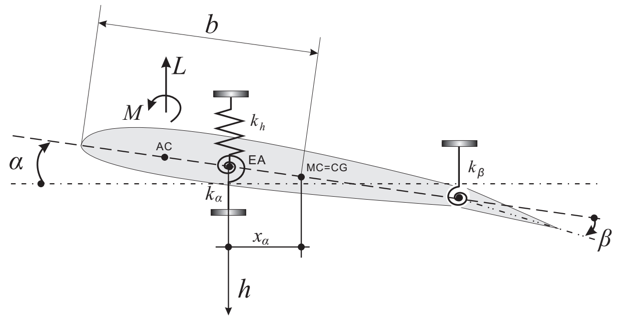


Fig. 1 2-D Wing-flap aeroelastic model.

It should be noted here that to minimize the performance measure J is not the primary goal at the current stage of this research. Instead, it serves as the selecting measure to obtain one certain control input which can be used to achieve the primary goal of stabilization.

One possible algorithm to find the suboptimal solution to the optimization problem (40) is shown in algorithm 1. In this example algorithm, a finite set of the feedback gain matrix $\boldsymbol{K}$ is given and denoted as Ξ. The example algorithm calculates the value J of the cost function defined in the optimization problem (40) for every $\boldsymbol{K}(k)$ in Ξ, determines the one which possesses the smallest J, and additionally, make the sequence of the predicted states satisfy the stability condition. So a suitable feedback gain $\boldsymbol{K}$ can be defined to stabilize the plant at the time instant $t = r$. Although this $\boldsymbol{K}(k)$ is only optimal for the present time instant, it can be applied to construct the suitable control input $\boldsymbol{u}(k)$ and applied to the plant to fulfill the goal of control.

5 Numerical example of stabilizing a nonlinear aeroelastic system

In this paper, a nonlinear aeroelastic system [22] is used as a benchmark example to show the effectiveness and flexibility of the proposed control approach. The control of this benchmark system has been widely studied with different control approaches [25,29,28,23,30,31].

5.1 Configuration of the aeroelastic system

The configuration of the considered 2-D nonlinear aeroelastic wing section is shown in figure 1. The two degrees of freedom, the pitching movement and the plunging one, are respectively restrained by a pair of springs attached to the elastic axis (EA) of the airfoil. A single trailing-edge control surface is used to control the air flow, thereby providing more maneuverability to suppress instability. This model is accurate for airfoils at low velocity and has been confirmed by both computational tests [4] and wind tunnel experiments [21].

According to [22], the equations of motion governing the aerolastic system are given as

$$\begin{bmatrix} m_T & m_W x_\alpha b \\ m_W x_\alpha b & I_\alpha \end{bmatrix} \begin{bmatrix} \ddot{h} \\ \ddot{\alpha} \end{bmatrix} + \begin{bmatrix} c_h & 0 \\ 0 & c_\alpha \end{bmatrix} \begin{bmatrix} \dot{h} \\ \dot{\alpha} \end{bmatrix} + \begin{bmatrix} k_h & 0 \\ 0 & k_\alpha \end{bmatrix} \begin{bmatrix} h \\ \alpha \end{bmatrix} = \begin{bmatrix} -L \\ M \end{bmatrix}, \tag{42}$$

where plunging and pitching displacement are denoted as h and α respectively. In Eq. (42) m_W denotes the mass of the wing, m_T represents the total mass of the wing and its support structure, b the semi-chord of the wing, I_α the moment of inertia, x_α the non-dimensional distance from the center of mass to the elastic axis, c_α and c_h the pitch and plunge damping coefficients respectively, k_α and k_h the pitch and plunge spring constants respectively, and M and L denote the quasi-steady aerodynamic lift and moment. In the case when the quasi-steady aerodynamics is considered, M and L should be written as

$$\begin{cases} L = \rho U^2 b c_{l_\alpha} \left[\alpha + \frac{\dot{h}}{U} + (\frac{1}{2} - a) b \frac{\dot{\alpha}}{U}\right] + \rho U^2 b c_{l_\beta} \beta \\ M = \rho U^2 b^2 c_{m_\alpha} \left[\alpha + \frac{\dot{h}}{U} + (\frac{1}{2} - a) b \frac{\dot{\alpha}}{U}\right] + \rho U^2 b^2 c_{m_\beta} \beta \end{cases}, \tag{43}$$

where the denotations are explained in table 5.1.

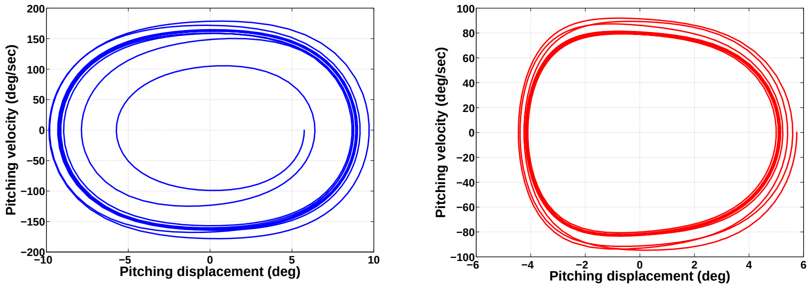
Table 1 Denotation list of aerodynamic coefficients

Symbols	Representations
ρ	Density of air
a	Nondimensional distance from mid chord to elastic axis
b	Semi-chord of the wing
c_h, c_α	Pressure coefficients
$c_{m\alpha}$, c_{m_α}	Lift and moment coefficients per angle of attack
c_{l_β}, c_{m_β}	Lift and moment coefficients per angle of control surface deflection
U	Free stream velocity
x_α	Distance from elastic axis to mass center

The structural nonlinearity is supposed to exist in the pitching spring constant k_α, which is assumed as to be a polynomial of α, shown as

$$k_\alpha = \sum_{i=0}^{4} k_{\alpha_i} \alpha^i. \tag{44}$$

The control objective is to drive the flap angle β properly so that the instability caused by structural nonlinearities can be suppressed in the vicinity of the nominal system flutter speed with smaller control errors and less input energy.



(a) Open-loop response with $k_{\alpha 1}$ (b) Open-loop response with $k_{\alpha 2}$

Fig. 2 Open-loop system phase portraits with different nonlinearities

5.2 Problem settings and specific tasks of control

As pointed in [17], a stable pitch motion of the 2-D wing section leads at the same time to a stable plunging motion, and vice versa. Due to this knowledge, it is reasonable to only take the subsystem consisted of pitching angels and pitching velocity into consideration. If this subsystem can be stabilized, the stability of global system can be guaranteed.

It is assumed in this example that the pitching motion is fully measurable, i.e., the states α and $\dot{\alpha}$ are known as measurements. As mentioned in the introduction, the measurements are assumed as noise-free.

The task of control is to stabilize the system following the above assumptions with two different nonlinearities of the pitching spring stiffness according to [16] and [1], respectively, as

$$k_{\alpha 1} = \left[6.8 \; 10.0 \; 667.7 \; 26.6 \; -5087.9 \right] [\alpha^i], \tag{45}$$

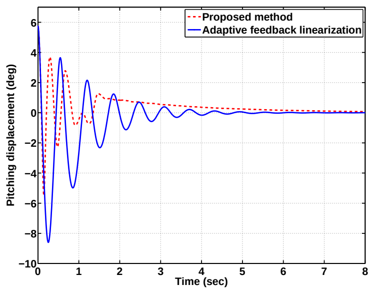
and

$$k_{\alpha 2} = \left[2.8 \; -62.3 \; 3709.7 \; -24195.6 \; 48756.9 \right] [\alpha^i]. \tag{46}$$

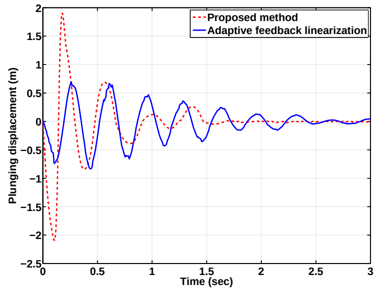
The adaptive feedback linearization control in [16] is also applied in the simulation to be compared with the proposed control method. It should be mentioned that the parameter settings of the both control methods, including the feedback gains in adaptive feedback linearization method, are *kept as the same* in the both simulations when the nonlinearity $k_{\alpha 1}$ is changed into $k_{\alpha 2}$, in order to compare the adaptive abilities of the both control approaches.

5.3 Simulation results

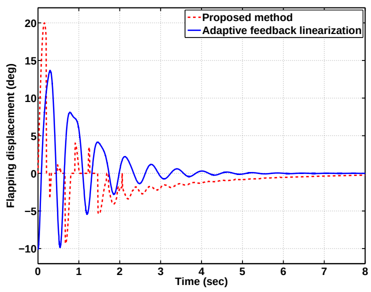
The open-loop responses of the concerned aeroelastic system with different nonlinearities $k_{\alpha 1}$ and $k_{\alpha 2}$ are shown in figure 2(a) and figure 2(b). It can be seen from these two figures that the structural nonlinearity in the pitching stiffness leads the system response to the Limit Cycle Oscillation (LCO).



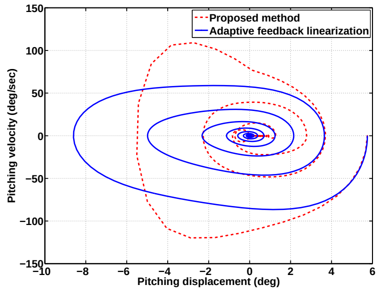
(a) Time history of pitching motion



(b) Time history of plunging motion



(c) Time history of control input



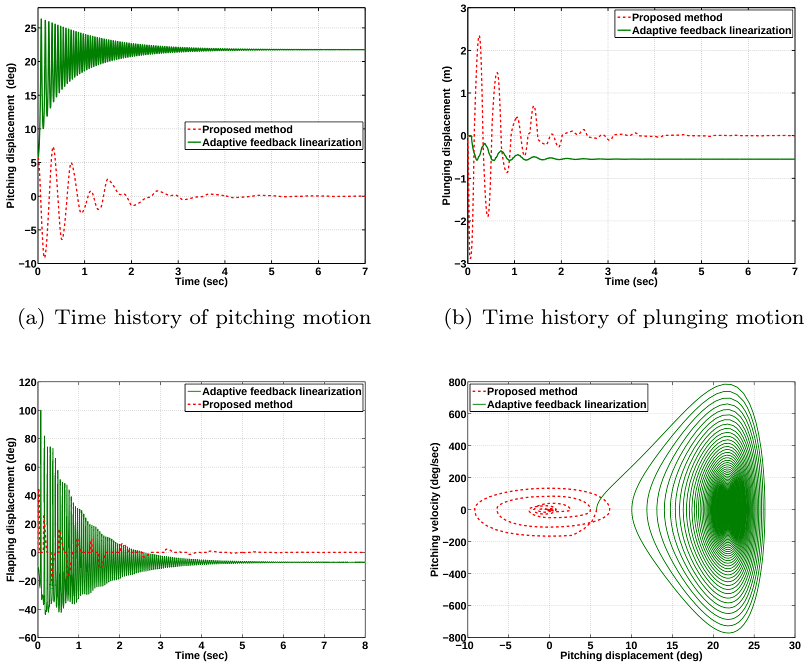
(d) Phase portrait of pitching motion with control

Fig. 3 Closed-loop response with nonlinearity $k_{\alpha 1}$

The system functionality measure in this example uses the form similar to the objective function in linear optimal control introduced in equation (41). The optimization problem (40) here is solved by heuristic search. The searching space is defined as the varying region of the flapping motion β, which is detailed as $(-45\text{deg},\ 45\text{deg})$.

Simulations of the close-loop system are performed with wind speed $U = 20\text{m/s}$ and structural parameter $a = 0.8$ (nondimensional distances from mid-chord to the elastic axis). The initial conditions for the state variables of the system are selected as $\alpha(0) = 5.75\,(\text{deg})$, $h(0) = 0.01\text{m}$, $\dot{\alpha}(0) = 0\,(\text{deg/s})$, and $\dot{h}(0) = 0\text{m/s}$. The sampling time is set as $1 \times 10^{-4}\text{sec}$.

The simulation results of the closed-loop system with the nonlinearity $k_{\alpha 1}$ and $k_{\alpha 2}$ are given in figure 3 and figure 4, respectively. The red dashed curves in figure 3 and figure 4 show the simulation results with the proposed control method, from which it can be seen that both system trajectories converge to the origin of the state space with increasing time. From these simulation results it can be seen that the proposed control method can stabilize the unknown nonlinear aeroelastic system with different nonlinearities. This fact shows that the proposed control strategy can not only be used to stabilize the unknown



(a) Time history of pitching motion
(b) Time history of plunging motion
(c) Time history of control input
(d) Phase portrait of pitching motion with control

Fig. 4 Closed-loop response with nonlinearity $k_{\alpha 2}$

nonlinear system, but also adapt the variation of system dynamics and realize the goal of stabilizing unknown plants.

The simulation results of adaptive feedback linearization control with respect to the two different nonlinearities are shown as the blue curves in figure 3 and green curves in figure 4, respectively. It can be seen that in the case of $k_{\alpha 1}$ the adaptive feedback linearization has better control performance with respect to the error and settling time than the proposed method. However, it cannot stabilize the system to the desired position (origin in the state space) in the second simulation with $k_{\alpha 2}$, in which case the sytem dynamics are assumed unknown to feedback linearization control and its parameters are *kept as the same as* those in the first simulation.

From this comparison it is shown that the proposed method possesses higher adaptability than the adaptive feedback linearization method when the system dynamics is unknown.

6 Conclusion

This paper firstly proposes the necessary and sufficient condition of the existence of a QLF only for a trajectory of nonlinear discrete-time systems, without necessity of knowing the mathematical description of the system. The existence of a QLF for the considered trajectory can be determined by examining the existence of a suitable orthogonal matrix, with which a convex cone can be constructed and have a non-empty intersection with positive real space.

Based the proposed stability criterion, a new control method is proposed and can be applied to stabilize a nonlinear aeroelastic system using only the information of the measured system states. Simulation results show that the proposed control can adaptively fulfill the task of stabilizing unknown systems without changing any parameters.

References

1. A. Behal P. Marzocca, V.M.R., Gnann, A.: Nonlinear adaptive model free control of an aeroelastic 2-d lifting surface. AIAA Journal of Guidance, Control, and Dynamics **29**(2), 382–390 (2006)
2. Acikmese, B., Corless, M.: Stability analysis with quadratic lyapunov functions: Some necessary and sufficient multiplier conditions. Systems & Control Letters **57**(1), 78–94 (2008)
3. Barmish, B.R.: Necessary and sufficient conditions for quadratic stabilizability of an uncertain system. Journal of Optimization Theory and Applications **46**, 399–408 (1985)
4. Bennett, R.M., Scott, R.C., Wieseman, C.: Computational test cases for the benchmark active controls model. AIAA Journal of Guidance, Control, and Dynamics **23**(5), 922–929 (2000)
5. Bernussou, J., Peres, P.L.D., Geromel, J.C.: A linear-programming oriented procedure for quadratic stabilization of uncertain systems. Systems & Control Letters **13**(1), 65–72 (1989)
6. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge, United Kindom (2004)
7. Carathéodory, C.: Über den Variabilittsbereich der Fourierschen Konstanten von positiven harmonischen Funktionen. Rendiconti del Circolo Matematico di Palermo **32**(1), 193–217 (1911)
8. Cheng, D., Guo, L., Huang, J.: On quadratic lyapunov functions. IEEE Transactions on Automatic Control **48**, 885–890 (2003)
9. Feng, G.: Approaches to quadratic stabilization of uncertain fuzzy dynamic systems. IEEE Transactions on Circuits and Systems I-Fundamental Theory and Applications **48**(6), 760–769 (2001)
10. Haykin, S.: Neural Networks, 2nd edn. Prentice Hall International, New Jersey, USA (1999)
11. Horn, R.A., Johnson, C.R.: Topics in matrix analysis. Cambridge University Press, Cambridge, UK (1994)
12. Kar, I.N., Dasgupta, A.: Quadratic stabilization of a class of linear systems and its application to robot control. International Journal of Systems Science **29**(3), 267–274 (1998)
13. Khalil, H.K.: Nonlinear Systems, 3rd edn. Prentice Hall, New Jersey (2002)
14. Khargonekar, P.P., Petersen, I.R., Zhou, K.M.: Robust stabilization of uncertain linear-systems - quadratic stabilizability and h infinity-control theory. IEEE Transactions on Automatic Control **35**(3), 356–361 (1990)
15. Kiriakidis, K., Grivas, A., Tzes, A.: Quadratic stability analysis of the takagi-sugeno fuzzy model. Fuzzy Sets and Systems **98**(1), 1–14 (1998)
16. Ko, J., Strganac, T.W., Kurdila, A.J.: Adaptive feedback linearization for the control of a typical wing section with structural nonlinearity. Nonlinear Dynamics **18**(3), 289–301 (1999)
17. Lindgren, A.G., Pinkos, R.F.: Stability of symmetric nonlinear multivariable systems. Journal of the Franklin Institute **282**(2), 92–101 (1966)

18. Liu, X.D., Zhang, Q.L.: Approaches to quadratic stability conditions and h infinity control designs for t-s fuzzy systems. IEEE Transactions on Fuzzy Systems **11**(6), 830–839 (2003)
19. Mahmoud, M.S., Almuthairi, N.F.: Quadratic stabilization of continuous-time systems with state-delay and norm-bounded time-varying uncertainties. IEEE Transactions on Automatic Control **39**(10), 2135–2139 (1994)
20. Montagner, V.F., Oliveira, R.C.L.F., Peres, P.L.D.: Convergent LMI relaxations for quadratic stabilizability and h-infinity control of takagi-sugeno fuzzy systems. IEEE Transactions on Fuzzy Systems **17**(4), 863–873 (2009)
21. Mukhopadhyay, V.: Transonic flutter suppression control law design and wind-tunnel test results. AIAA Journal of Guidance, Control, and Dynamics **23**(5), 930–937 (2000)
22. Scott, R.C., Hoadley, S.T., Wieseman, C.D., Durham, M.H.: The benchmark active controls technology model aerodynamic data. AIAA Journal of Guidance, Control, and Dynamics **23**(5), 914–921 (2000)
23. Singh, S., Wang, L.: Output feedback form and adaptive control of a nonlinear aeroelastic system. AIAA Journal of Guidance, Control, and Dynamics **25**(4), 725–732 (2002)
24. Tanaka, K., Ikeda, T., Wang, H.O.: Robust stabilization of a class of uncertain nonlinear systems via fuzzy control: Quadratic stabilizability, h-infinity control theory, and linear matrix inequalities. IEEE Transactions on Fuzzy Systems **4**(1), 1–13 (1996)
25. Wang, M., Zhao, J.: Quadratic stabilization of a class of switched nonlinear systems via single lyapunov function. Nonlinear Analysis-Hybrid Systems **4**(1), 44–53 (2010)
26. Williams, R., Zipser, D.: A learning algorithm for continually running fully recurrent neural networks. Neural Computation **1**(2), 270–280 (1989)
27. Xie, L.H., Fu, M.Y., Desouza, C.E.: H-infinity control and quadratic stabilization of systems with parameter uncertainty via output-feedback. IEEE Transactions on Automatic Control **37**(8), 1253–1256 (1992)
28. Xing, W., Singh, S.: Adaptive output feedback control of a nonlinear aeroelastic structure. AIAA Journal of Guidance, Control and Dynamics **23**(6), 1109–1116 (2000)
29. Zeng, Y., Singh, S.: Output feedback variable structure adaptive control of an aeroelastic systems. AIAA Journal of Guidance, Control, and Dynamics **21**(6), 830–837 (1998)
30. Zhang, F., Söffker, D.: Active flutter suppression of a nonlinear aeroelastic system using pi-observer. In: H. Ulbrich, L. Ginzinger (eds.) Motion and Vibration Control, pp. 367–376. Springer, Dordrecht, The Netherlands (2009)
31. Zhang, F., Söffker, D.: Stabilization of a nonlinear aeroelastic wing-section using modified model-free control with self-improvement of control performance. In: 2010 AIAA Guidance, Navigation, and Control Conference (AIAA-GNC). Toronto, Canada (2010). AIAA-2010-8284
32. Zhang, F., Söffker, D.: A data-driven online stability monitoring method for unknown discrete-time nonlinear systems. In: 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), pp. 7356 –7361. Orlando, USA (2011)
33. Zhao, J., Dimirovski, G.M.: Quadratic stability of a class of switched nonlinear systems. IEEE Transactions on Automatic Control **49**(4), 574–578 (2004)