Neurocomputing 128 (2014) 232–241

Contents lists available at ScienceDirect

Neurocomputing

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

Discrete-time hypersonic flight control based on extreme learning machine

Bin Xu [a,b,*], Yongping Pan [b], Danwei Wang [b], Fuchun Sun [c]

[a] School of Automation, Northwestern Polytechnical University, Shaanxi, Xi'an 710072, China
[b] School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798, Singapore
[c] Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 28 August 2012
Received in revised form
16 December 2012
Accepted 11 February 2013
Available online 28 October 2013

Keywords:
Hypersonic aircraft
Extreme learning machine
Neural networks
Single-hidden layer feedforward network

ABSTRACT

This paper describes the neural controller design for the longitudinal dynamics of a generic hypersonic flight vehicle (HFV). The dynamics are transformed into the strict-feedback form. Considering the uncertainty, the neural controller is constructed based on the single-hidden layer feedforward network (SLFN). The hidden node parameters are modified using extreme learning machine (ELM) by assigning random values. Instead of using online sequential learning algorithm (OSLA), the output weight is updated based on the Lyapunov synthesis approach to guarantee the stability of closed-loop system. By estimating the bound of output weight vector, a novel back-stepping design is presented where less online parameters are required to be tuned. The simulation study is presented to show the effectiveness of the proposed control approach.

1. Introduction

Hypersonic flight vehicles may offer a reliable and more cost efficient way to access space by reducing flight time. Also quick response and global attack became possible. Hypersonic flight control is challenging since the longitudinal model of the dynamics is known to be unstable, non-minimum phase with respect to the regulated output, and affected by significant model uncertainty. The main difficulty of the control law design for the hypersonic aircraft is due to the high complexity of the motion equations and there is little knowledge of the aerodynamic parameters of the vehicle.

Recently adaptive control and robust control are popularly studied on hypersonic flight controller design [1]. Back-stepping design [2] is an explicit tool for systematic nonlinear design. The HFV dynamics are written in the linearly parameterized form [3] and then robust adaptive dynamic inversion with back-stepping arguments is conducted. Dynamic surface control with control inputs saturation design is studied in [4].

Intelligent control is one important aspect for hypersonic flight control since it is with the capability of uncertainty approximation [5–7,22]. Since modern aircrafts are equipped with digital computers, the controller should be designed in discrete-time form [8]. Controller on the basis of continuous system is usually implemented by a digital computer with a certain sampling interval [9]. There are two methods for designing the digital controller. One method, called emulation, designs a controller with the continuous-time system, and then discretizes the controller. The other is to design the controllers directly based on the discrete system. In contrast to the emulation method, the discrete controller is designed in a discrete domain so that the performance of the controller may not depend on the sampling rate and the upper bounds of the neural network (NN) weight update rates guaranteeing the convergence can be estimated analytically while emulation method is otherwise [10].

Focused on discrete time design, the adaptive NN back-stepping HFV control [11] is studied to deal with the system uncertainty. The Kriging based adaptive controller is designed in [12] where the uncertainty is described as the realization of the Gaussian random functions [13]. The simulation shows the effectiveness of the controller design [11,14]. In the above schemes, the structure of NN is determined according to some prior information regarding the system to be approximated. Then the stable adaptive laws can be generated in a linear fashion. However in practice, systems are time-varying and the prior information is difficult to obtain. In this case, the exact values for the NN are hard to determine.

In this paper, a new stable neural control scheme is presented. The SLFN with RBF nodes is used as the function approximator to estimate the unknown nonlinearity. Different from the existing methods, the parameters of SLFN are adjusted based on the ELM. ELM has attracted widespread concern in recent years [15–17]

* Corresponding author at: School of Automation, Northwestern Polytechnical University, Shaanxi, Xi'an 710072, China. Tel.: +86 29 88431389.
E-mail address: smileface.binxu@gmail.com (B. Xu).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.02.049

since it overcomes some challenges faced by other techniques [18] such as (1) slow learning speed, (2) trivial human intervene, and/or (3) poor computational scalability. ELM works for generalized SLFN. The essence of ELM is that the hidden layer of SLFN needs not to be tuned. Compared with those traditional computational intelligence techniques, ELM provides better generalization performance at a much faster learning speed and with least human intervene. In [19], an approach for performing regression on large data sets in reasonable time is proposed using an ensemble of ELMs and the experiments show that competitive performance is obtained on the regression tasks. In [20], the ELM is utilized to train the controller by randomly assigning the parameters of hidden nodes. The output weights are synthesized using a Lyapunov function for guaranteeing the stability of the closed-loop system. Also it is indicated that original ELM cannot follow its reference trajectory well since the original ELM lacks stability proof of the whole control system and thus the convergence of the tracking error cannot be satisfied.

In this paper, considering the use of digital computer, the backstepping controller is designed with ELM by randomly assigning the parameters of hidden nodes. The updating law is designed with Lyapunov synthesis approach in discrete-time. Following the functional decomposition [11], we design the controller separately for the subsystems. Furthermore, the “minimal learning parameter” technique based on bound estimation of weight vector [14,21] is incorporated to reduce the computation burden. In this paper, only the cruise trajectories are considered for the control problem in this paper and we does not consider the ascent or the reentry of the vehicle.

This paper is organized as follows. Section 2 describes the longitudinal dynamics of a generic hypersonic flight vehicle. The strict-feedback form is formulated and the discrete analysis model is obtained in Section 3. SLFN based on ELM is illustrated in Section 4. Section 5 presents the adaptive controller design based on ELM. The weight bound estimation based controller is designed in Section 6. The simulation result is included in Section 7. Section 8 presents several comments and final remarks.

2. Hypersonic air vehicle model

The model of the longitudinal dynamics of a generic hypersonic aircraft in [1] is considered. This model is composed of five state variables $\mathbf{X}=[V,h,\alpha,\gamma,q]^T$ and two control inputs $\mathbf{U}_c=[\delta_e,\Phi]^T$ where V is the velocity, γ is the flight path angle, h is the altitude, α is the attack angle, q is the pitch rate, δ_e is the elevator deflection and Φ is the throttle setting.

The dynamics of hypersonic aircraft are described by the following nonlinear equations:

$$\dot{V}=\frac{T\cos\alpha-D}{m}-\frac{\mu\sin\gamma}{r^2} \tag{1}$$

$$\dot{h}=V\sin\gamma \tag{2}$$

$$\dot{\gamma}=\frac{L+T\sin\alpha}{mV}-\frac{(\mu-V^2r)\cos\gamma}{Vr^2} \tag{3}$$

$$\dot{\alpha}=q-\dot{\gamma} \tag{4}$$

$$\dot{q}=\frac{M_{yy}}{I_{yy}} \tag{5}$$

where T, D, L and M_{yy} represent thrust, drag, lift-force and pitching moment respectively, m, I_{yy} and μ represent the mass of aircraft, moment of inertia about pitch axis and gravity constant. r is the radial distance from center of the earth.

Refer to Appendix A for more information about the model.

3. System transformation

3.1. Strict-feedback formulation

Referred to [11,14], the formulation of the subsystems is presented in (6) and (8). The related definition of the system is listed in Appendix B.

The velocity subsystem (1) can be rewritten as follows:

$$\begin{aligned}\dot{V}&=f_v+g_vu_v\\ u_v&=\Phi\\ y_v&=V\end{aligned} \tag{6}$$

The tracking error of the altitude is defined as $\tilde{h}=h-h_d$ and the flight path command is chosen as

$$\gamma_d=\arcsin\left[\frac{-k_h(h-h_d)-k_I\int(h-h_d)\,dt+\dot{h}_d}{V}\right] \tag{7}$$

if $k_h>0$ and $k_I>0$ are chosen and the flight path angle is controlled to follow γ_d, the altitude tracking error is regulated to zero exponentially [5].

Assumption 1. The thrust term $T\sin\alpha$ in (3) is neglected because it is generally much smaller than L.

Define $\mathbf{X}_A=[x_1,x_2,x_3]^T$, $x_1=\gamma$, $x_2=\theta_p$, $x_3=q$ where $\theta_p=\alpha+\gamma$. Then the strict-feedback form equations of the attitude subsystem (3)–(5) are written as

$$\begin{aligned}\dot{x}_1&=f_1(x_1)+g_1(x_1)x_2\\ \dot{x}_2&=f_2(x_1,x_2)+g_2(x_1,x_2)x_3\\ \dot{x}_3&=f_3(x_1,x_2,x_3)+g_3(x_1,x_2,x_3)u_A\\ u_A&=\delta_e\\ y&=x_1\end{aligned} \tag{8}$$

Assumption 2. f_i and g_i are unknown smooth functions. There exist known constants $\overline{g_i}$ and $\underline{g}_i$ such that $\overline{g}_i\geq g_i\geq \underline{g}_i>0$, $i=1,\ 3,\ v$.

Remark 1. From Appendix B, the f_i is really complicated and it is considered to be totally unknown. $g(k)$ is time varying however according to the parameter perturbation, the bound of gain is easy to derive with the simple linear expression.

The goal pursued in this study is to design a dynamic controller δ_e and Φ to steer system altitude and velocity from a given set of initial values to desired trim conditions with the tracking reference h_d and V_d. With the command transformation (7), the control objective of system (8) is to design an adaptive controller, which makes $\gamma\to\gamma_d$, further $h\to h_d$ and all the signals involved are bounded.

3.2. Discrete-time model

By Euler expansion with sample time T_s, systems (6) and (8) can be approximated as

$$V(k+1)=V(k)+T_s[f_v(k)+g_v(k)u_v(k)] \tag{9}$$

$$\begin{aligned}x_1(k+1)&=x_1(k)+T_s[f_1(k)+g_1(k)x_2(k)]\\ x_2(k+1)&=x_2(k)+T_s[f_2(k)+g_2(k)x_3(k)]\\ x_3(k+1)&=x_3(k)+T_s[f_3(k)+g_3(k)u_A(k)]\end{aligned} \tag{10}$$

4. SLFN based on ELM

For N arbitrary distinct samples $(\mathbf{x}_i,\mathbf{t}_i)$, where $\mathbf{x}_i=[x_{i1},x_{i2},\ldots,x_{in}]^T\in R^n$ and $\mathbf{t}_i=[t_{i1},t_{i2},\ldots,t_{im}]^T\in R^m$, standard SLFN with