ISA Transactions 70 (2017) 161–172

Contents lists available at ScienceDirect

ISA Transactions

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

Research article

Control-oriented modeling and adaptive backstepping control for a nonminimum phase hypersonic vehicle

Linqi Ye [a], Qun Zong [a], Bailing Tian [a,*], Xiuyun Zhang [a], Fang Wang [b]

[a] School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China
[b] School of Science, Yanshan University, Qinhuangdao, Hebei 066004, China

ARTICLE INFO

Article history:
Received 30 June 2016
Received in revised form
26 May 2017
Accepted 17 July 2017
Available online 25 July 2017

Keywords:
Hypersonic vehicle
Nonminimum phase
Control-oriented modeling
Adaptive backstepping control

ABSTRACT

In this paper, the nonminimum phase problem of a flexible hypersonic vehicle is investigated. The main challenge of nonminimum phase is the prevention of dynamic inversion methods to nonlinear control design. To solve this problem, we make research on the relationship between nonminimum phase and backstepping control, finding that a stable nonlinear controller can be obtained by changing the control loop on the basis of backstepping control. By extending the control loop to cover the internal dynamics in it, the internal states are directly controlled by the inputs and simultaneously serve as virtual control for the external states, making it possible to guarantee output tracking as well as internal stability. Then, based on the extended control loop, a simplified control-oriented model is developed to enable the applicability of adaptive backstepping method. It simplifies the design process and releases some limitations caused by direct use of the no simplified control-oriented model. Next, under proper assumptions, asymptotic stability is proved for constant commands, while bounded stability is proved for varying commands. The proposed method is compared with approximate backstepping control and dynamic surface control and is shown to have superior tracking accuracy as well as robustness from the simulation results. This paper may also provide a beneficial guidance for control design of other complex systems.

1. Introduction

Nonminimum phase problem of aircrafts has been studied in the past few years [1–6]. Before the emergence of canard configured fighters in the early 1990s, most aircrafts are designed with a conventional horizontal tail that uses an elevator for pitch control. A typical character of this kind of aircrafts is the existence of nonminimum phase zeros in the transfer function from elevator to flight-path angle [1,2]. As a result, the step response of the elevator angular deflection to pitch angle exhibits initial undershoot, which is a common property of nonminimum phase systems [3,4]. The nonminimum phase behavior of aircrafts results from the fact that the generation of an upward pitch moment produces a small parasitic downward force at the same time [5]. In the case of hypersonic vehicle, there is not much difference from conventional aircrafts. As stated in [2], the nonminimum phase behavior of a hypersonic vehicle stems from elevator-to-lift coupling. When the elevator is actuated to produce a nose-up moment, the aircraft experiences a loss of lift from the elevator. So the vehicle will instantaneously lose altitude before achieving a positive climb rate. Adding a canard to the vehicle may suppress the nonminimum phase behavior, but it will cause a significant thermal stress problem, making it difficult to carry out in practice [6].

Nonminimum phase is an interesting and existing problem in output tracking control. For linear system, nonminimum phase problem may be less concerned, since linear control methods such as pole assignment and LQR (linear quadratic regulator) are not restricted by the nonminimum phase character. But when it turns to nonlinear control, things become quite different. As to nonlinear system, nonminimum phase system refers to the system with unstable zero dynamics [7], which are equivalent to unstable transmission zeros in linear system. The nonminimum phase character of a plant restricts direct application of the powerful nonlinear control technique, dynamic inversion [8]. Dynamic inversion provides a way to construct a controller for exact output tracking. But for nonminimum phase system, the dynamic inversion method and all of its variants (such as feedback linearization) become invalid since they make the unstable zero dynamics an unobservable part in the closed loop system. The reason can be summarized as that dynamic inversion focuses on exact tracking of external states, while no control is imposed on the unstable internal dynamics. Therefore, exact tracking for nonminimum

* Corresponding author.
E-mail address: bailing_tian@tju.edu.cn (B. Tian).

http://dx.doi.org/10.1016/j.isatra.2017.07.019
0019-0578/

phase system by dynamic inversion is at the price of the divergency of internal dynamics.

Another influence brought by nonminimum phase is the performance limitations. It has long been recognized that a fundamental limitation exists in the achievable transient tracking performance for a nonminimum phase system. As shown in [9], perfect tracking of any reference signal is possible in the absence of unstable zero dynamics, that is, the L2-norm of the tracking error can be made arbitrarily small. But this is no longer possible for systems with unstable zero dynamics, because an amount of "output energy" must be used for stabilizing the internal dynamics.

In conclusion, compared with minimum phase systems, the main challenges for nonminimum phase systems are: (1) It is much more difficult to design a stable nonlinear controller since the direct application of dynamic inversion and all of its variants fail to be used; (2) Inherent performance limitations exist for nonminimum phase systems.

For the second challenge, as stated in [2], the presence of nonminimum phase severely limits the achievable bandwidth of the hypersonic vehicle control system. However, this is what we can't change. What we concern about is the first challenge: how to design a stable nonlinear controller for nonminimum phase hypersonic vehicle. To solve this problem, some beneficial work has been done over the past few years. Two remarkable works among them are made by Parker [10,11] and Fiorentini [8].

Parker [10,11] proposes a useful method, approximate feedback linearization, which is derived from [12], to solve the nonminimum phase problem of hypersonic vehicle. Due to the elevator-to-lift coupling, the elevator angular deflection appears in lower-order derivative of altitude. By neglecting the weak elevator coupling and making other simplifications, a control-oriented model (COM) is obtained. Dynamic inversion can then be applied to the COM, and it results in approximate linearization of the original model. Fiorentini [8] adopts another method, output redefinition, to solve the nonminimum phase problem of hypersonic vehicle. Output redefinition is proposed by Gopalswamy and Hedrick [13] and is a general method to deal with the nonminimum phase problem. The main idea of output redefinition is to find a new output to adjust the zero dynamics such that the zero dynamics with respect to the new output are acceptable, and then design controller based on the new output. By virtue of this idea, Fiorentini [8] takes coordinate change to obtain the normal form of zero dynamics and then selects a suitable command for the pitch angle to stabilize the zero dynamics. Therefore by selecting the pitch angle tracking error as a new output, the corresponding zero dynamics are stable. Flight path angle (FPA) is chosen as actual output and an integral augmentation is taken on the FPA to guarantee there is no steady-state error. These two papers both solved the nonminimum phase problem of hypersonic vehicle successfully. However, their design process may be a little complex. For the approximate feedback linearization method, it requires taking dynamic extension at the input side and calculating the analytic expressions of the third derivative of velocity and the fourth derivative of altitude, which is a huge amount of work. And for output redefinition, we need to take several complicated coordinate changes to remove the control inputs from the zero dynamics. Besides, since they both select FPA rather than altitude as an output, the altitude tracking error will not converge to zero.

In other literatures, researchers try to apply backstepping control to design controller for the COM [14–18]. However, restriction is imposed on direct use of backstepping control since the complexity of the model will cause the "explosion of terms" problem [22] in calculating the analytical expressions of the command derivatives for the virtual control. In [14], approximate backstepping is adopted where the command derivatives are neglected. In [15,16], dynamic surface is employed where the command derivatives are estimated by a filter. Similarly, in [17,18], command filter backstepping is used, and the command derivatives are also estimated by a filter. However, the relationship between nonminimum phase and backstepping control is not mentioned in these literatures.

Motivated by the literatures above, we design a stable nonlinear controller for a nonminimum phase hypersonic vehicle from a new way by combining model simplification and adaptive backstepping control. First, we make further simplification on the COM and develop a simplified control-oriented model (SCOM). Then adaptive backstepping control is applied to the SCOM. The proposed method is compared with approximate backstepping and dynamic surface method developed on the COM and is shown to have superior tracking accuracy as well as robustness from the simulation results.

The main contributions of the paper are twofold. First, we find out the principle to design a stable nonlinear controller through backstepping for nonminimum phase hypersonic vehicle. The key is extending the control loop to cover the internal dynamics. Similar to hypersonic vehicle, the nonminimum phase behavior of a wide variety of systems is caused by some undesired couplings, which renders the input appears in lower order derivative of the output and reduces the system's relative degree. For those systems, the natural control loop which chooses the shortest route from the output to the input, though achieves exact output tracking, is unstable since it puts no control on the unstable internal dynamics and leaves it as an unobservable part in the closed loop system. On the contrary, in the extended control loop which covers the internal dynamics, the internal states are directly controlled by the inputs and simultaneously serve as virtual control for the external states, which allows it to have the possibility of realizing output tracking as well as guaranteeing internal stability. Although aimed at hypersonic vehicle, this principle may be also applicable to other nonminimum phase systems, such as VTOL (vertical takeoff and landing aircraft) [12], inverted pendulum [19], flexible manipulator [20], and ship [21]. Second, we present a new way to design a stable nonlinear controller for the nonminimum phase hypersonic vehicle by combining model simplification and adaptive backstepping control. It simplifies the design process and releases some limitations such as "explosion of terms" problem existed in using backstepping control to the COM. Therefore this paper may also provide a beneficial guidance for control design of other complex systems.

The remainder of the paper is organized as follows: In Section 2, the flexible hypersonic vehicle model is introduced and the control objective is defined. Section 3 gives problem formulation and the main results. Section 4 presents the process of control-oriented modeling. Section 5 introduces approximate backstepping and dynamic surface control design while Section 6 introduces adaptive backstepping control design. The simulation results are shown in Section 7, and conclusions are offered in Section 8.

2. Flexible hypersonic vehicle model

The full nonlinear model of a flexible hypersonic vehicle is proposed by Bolender and Doman [25], in which the forces and moments are not expressed in an analytical form. Afterwards, Fiorentini [26] replaces the expressions of forces and moments by curve-fit approximations and obtains a curve-fitted model (CFM). The CFM is of high fidelity to the full nonlinear model and is easy to understand. We will use the CFM as control plant in the simulation. The CFM is described as

$$\begin{aligned}
\dot{V} &= (T\cos\alpha - D - mg\sin\gamma)/m \\
\dot{h} &= V\sin\gamma \\
\dot{\gamma} &= (L + T\sin\alpha - mg\cos\gamma)/(mV) \\
\dot{\theta} &= Q \\
\dot{Q} &= M/I_{yy} \\
\ddot{\eta}_i &= -2\zeta_i\omega_i\dot{\eta}_i - \omega_i^2\eta_i + N_i \quad , \quad i = 1, 2, 3.
\end{aligned} \tag{1}$$

The model comprises of five rigid body states $[V, h, \gamma, \theta, Q]^T$, representing velocity, altitude, flight path angle, pitch angle, and pitch rate, respectively, and six flexible modes $[\eta_1, \dot{\eta}_1, \eta_2, \dot{\eta}_2, \eta_3, \dot{\eta}_3]^T$, which reflect the first three bending modes of the fuselage. $\alpha = \theta - \gamma$ is the angle of attack. m is the vehicle mass. ζ_i, ω_i are damping ratio and natural frequency of the flexible modes. g is the acceleration of gravity. I_{yy} is the moment of inertia.

There are two control inputs $u = [\phi, \delta_e]^T$, representing fuel air ratio and elevator angular deflection, respectively. Note that the control inputs don't display explicitly in (1) but affect it through the thrust T, lift L, drag D, pitching moment M and generalized forces N_i. The detailed formulations of the forces and moments and all the parameter values can be found in [27,28].

The outputs to be controlled are selected as $y = [V, h]^T$. Based on the flexible hypersonic vehicle model, the control objective is defined as follows:

Control Objective. For given smooth and bounded commands V_r, h_r, to design control law for the control inputs $u = [\phi, \delta_e]^T$ such that all states remain bounded and the tracking errors $\tilde{V} = V - V_r$, $\tilde{h} = h - h_r$ converge to zero when the commands turn to constants.

Remark 1. It is worth noting that the hypersonic vehicle in the presence of stochastic elevator faults and stochastic noise can be modeled as a Markov jump system [29]. Then, the corresponding theory, such as [29–31], may provide an effective way to this issue, which will be investigated in our future work.

3. Problem formulation and the main results

As a preliminary for control design, we will analyze the stability of the zero dynamics and explain how to design a stable nonlinear controller through backstepping.

In system (1), V, h, γ are external states and θ, Q, η are internal states. According to the definition of zero dynamics [7], the zero dynamics about the tracking errors $\tilde{V}$, $\tilde{h}$ are the dynamics of the internal states when $\tilde{V} = 0$, $\tilde{h} = 0$, i.e., $V = V_r$, $h = h_r$. Hence the first three equations of (1) can be written as

$$\begin{aligned}
\dot{V}_r &= f_1(V_r, h_r, \gamma, \theta, \eta) + g_1(V_r, h_r, \gamma, \theta)\phi \\
\dot{h}_r &= V_r\sin\gamma \\
\dot{\gamma} &= f_2(V_r, h_r, \gamma, \theta, \eta, \phi) + g_2(V_r, h_r, \gamma, \theta)\delta_e
\end{aligned} \tag{2}$$

For convenience, terms about δ_e in D have been neglected here so that ϕ becomes the only input in velocity loop, thus decoupling is not needed. By solving (2) we can get

$$\begin{aligned}
\gamma &= \arcsin(\dot{h}_r/V_r) \\
\phi &= (\dot{V}_r - f_1)/g_1 \\
\delta_e &= (\dot{\gamma} - f_2)/g_2
\end{aligned} \tag{3}$$

By substituting (3) into the rest equations of (1), we can get the zero dynamics, which might as well be denoted as

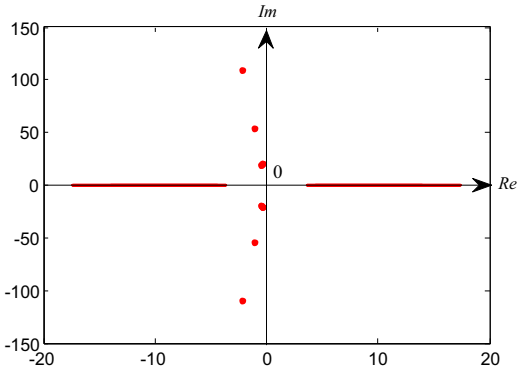

Fig. 1. Root map of the linearized zero dynamics.

$$\begin{aligned}
\dot{\theta} &= Q \\
\dot{Q} &= f_q(V_r, h_r, \theta, \eta) \\
\ddot{\eta}_i &= f_{\eta_i}(\dot{\eta}_i, \eta_i, \theta), \quad i = 1, 2, 3.
\end{aligned} \tag{4}$$

This is the zero dynamics corresponding to the outputs $\tilde{V}$, $\tilde{h}$, which represents the remaining dynamics when the outputs are regulated to zero. We can see from (4) that the zero dynamics is an 8th order nonlinear dynamic equation, whose stability depends on the commands V_r, h_r and the initial values of the states. Since it's difficult to analyze its global stability with respect to arbitrary commands, we will instead focus on its local stability near the equilibrium point with respect to constant commands as done in [8].

Consider the admissible flight range of this model [28], which is $V \in [7500, 11000]ft/s$, $h \in [70000, 135000]ft$. Each time we select a pair of constant commands V_r, h_r from this range, and then we calculate the equilibrium point and linearize the zero dynamics. When the whole range is covered, we obtain the root map of the linearized zero dynamics as Fig. 1.

From Fig. 1 we can see that the linearized zero dynamics has six complex eigenvalues which stay in the left half plane and two real eigenvalues which are one positive and one negative. So the zero dynamics are unstable due to the existence of the positive real eigenvalue, indicating that system (1) is nonminimum phase with respect to the tracking errors $\tilde{V}$, $\tilde{h}$. Next we will analyze the impact of the unstable zero dynamics and give the main results.

For convenience, we will not consider the flexible modes in the analysis since they count little to the nonminimum phase problem. Consider model (1), despite the flexible modes, the structure of the altitude loop is shown in Fig. 2.

Due to elevator-to-lift coupling, the control input δ_e appears in lower-order derivative of the output h. As a result, there are two routes from the output to the intput: the shorter route across the external states and the longer route across both the external and internal states. In fact this structure is very common in other

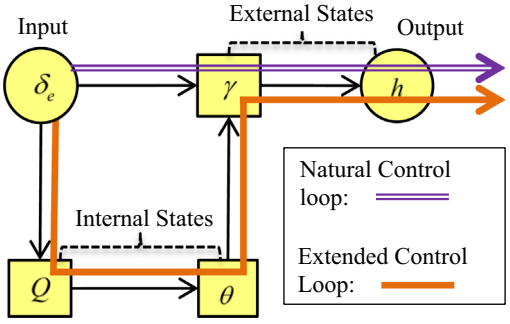

Fig. 2. System structure of hypersonic vehicle.

nonminimum phase system, such as VTOL [12], inverted pendulum [19], and ship [21]. In the philosophy of backstepping, a state is used as virtual control to control the state it affects. For this kind of systems, backstepping can be conducted from the two intput-output routes. We will call the shorter route as the natural control loop and the longer one as the extended control loop as shown in Fig. 2.

If we use dynamic inversion, it means to obtain the control law δ_e by taking inversion of the γ, h subsystem. From the perspective of backstepping, it uses the natural control loop: $\delta_e \to \gamma \to h$. As a result, when the goal of exact tracking is achieved, i.e., $\tilde{V}=0$, $\tilde{h}=0$, the internal dynamics θ, Q become equivalent to the zero dynamics (4) and cannot remain bounded as analyzed in the previous. Therefore, to guarantee stability of the internal states, we need to use the extended control loop. That is $\delta_e \to Q \to \theta \to \gamma \to h$, with γ, θ, Q being the virtual control in the system. It can be seen that in the extended control loop, the internal states are directly controlled by the inputs and simultaneously serve as virtual control for the external states. This change is crucial and enables it to have the possibility of realizing output tracking as well as stabilizing the internal dynamics. The desired value of the virtual control γ_r, θ_r, Q_r will be designed one by one to realize tracking of h and finally lead to the actual control law δ_e. Based on this thought, we will treat δ_e in γ, h subsystem as a perturbation and establish the control-oriented model in the next section.

4. Control-oriented modeling

Due to the complexity of the hypersonic vehicle model, there are several obstacles prevent us from nonlinear control design such as unknown flexible states, input-output couplings and nonminimum phase character. In this section we will follow the idea of control-oriented modeling [11] to simplify the original model (1) to render it more suitable for control design. The differences between our method and Ref. [11] are: (1) Other than neglecting the unavailable items in CFM, we treat them as uncertainties; (2) Ref. [11] is oriented to feedback linearization while our method is oriented to adaptive backstepping method; (3) In order to use adaptive backstepping control, further simplification on FPA dynamics is made so that the altitude loop becomes an integral chain with matched and mismatched uncertainties. While Ref. [11] takes dynamic extension at the inputs, which is not needed in our method.

As shown in Fig. 3, the process of control-oriented modeling is divided into two steps. In the first step, we will treat the unavailable items in CFM as uncertainties to make it available for control design. The obtained model after the first simplification is COM, which is in approximate strict-feedback form. In the second step, further simplification will be made on the FPA dynamics to enable the application of adaptive backstepping control. The obtained model after the second simplification is SCOM, which is in an integral chain form.

In the first step of control-oriented modeling, there are mainly three aspects considered here: (1) Flexible modes: they are not measurable so that they cannot be used in the controller; (2) Input-output couplings: δ_e in velocity loop and ϕ in altitude loop are undesired couplings, they should be viewed as uncertainties so we can decouple the system; (3) Nonminimum phase: δ_e in $\dot{\gamma}$ causes nonminimum phase problem, it should be treated as uncertainty to raise the relative degree of the system.

According to the above analysis, we separate the available and unavailable items in the CFM. By denoting the unavailable items as uncertainties, we can rewrite system (1) as follows

$$\begin{aligned}
\dot{V} &= f_V + g_V\phi + d_V \\
\dot{h} &= V\sin\gamma \\
\dot{\gamma} &= f_{\gamma 0} + d_\gamma \\
\dot{\theta} &= Q \\
\dot{Q} &= f_q + g_q\delta_e + d_q
\end{aligned} \tag{5}$$

where

$$\begin{aligned}
f_V &= (\bar{q}SC_T\cos\alpha - \bar{q}SC_D - mg\sin\gamma)/m,\ g_V = \bar{q}SC_{T\phi}\cos\alpha/m \\
d_V &= \left[\bar{q}SC_T^{\eta}\eta\cos\alpha + \bar{q}S\left(C_D^{\delta_e^2}\delta_e^2 + C_D^{\delta_e}\delta_e + C_D^{\eta}\eta\right)\right]/m \\
f_{\gamma 0} &= (\bar{q}SC_L + \bar{q}SC_T\sin\alpha - mg\cos\gamma)/(mV) \\
d_\gamma &= \left[\bar{q}S\left(C_L^{\delta_e}\delta_e + C_L^{\eta}\eta\right) + \bar{q}S\left(C_{T\phi}\phi + C_T^{\eta}\eta\right)\sin\alpha\right]/(mV) \\
f_q &= (z_T\bar{q}SC_T + \bar{q}\,\bar{c}SC_M)/I_{yy},\ g_q = \bar{q}\,\bar{c}SC_M^{\delta_e}/I_{yy} \\
d_q &= \left[z_T\bar{q}S\left(C_{T\phi}\phi + C_T^{\eta}\eta\right) + \bar{q}\,\bar{c}SC_M^{\eta}\eta\right]/I_{yy}
\end{aligned} \tag{6}$$

In (5), the first equation corresponds to the output V and the input ϕ, which forms the velocity loop. The other four equations correspond to the output h and the input δ_e, which forms the altitude loop.

As mentioned before, to deal with the nonminimum phase problem, we should take θ as a virtual control of γ. Therefore we rewrite $f_{\gamma 0}$ in an affine form with respect to θ

$$f_{\gamma 0} = f_\gamma + g_\gamma\theta \tag{7}$$

where

$$\begin{aligned}
f_\gamma &= \left(\bar{q}SC_L^0 - \bar{q}SC_L^{\alpha}\gamma + \bar{q}SC_T\sin\alpha - mg\cos\gamma\right)/(mV) \\
g_\gamma &= \bar{q}SC_L^{\alpha}/(mV)
\end{aligned} \tag{8}$$

What's more, since γ is very small through the flight envelope [26], we will use the following approximation

$$\dot{h} = V\sin\gamma \approx V\gamma \tag{9}$$

Then the altitude loop is in an approximate strict-feedback form:

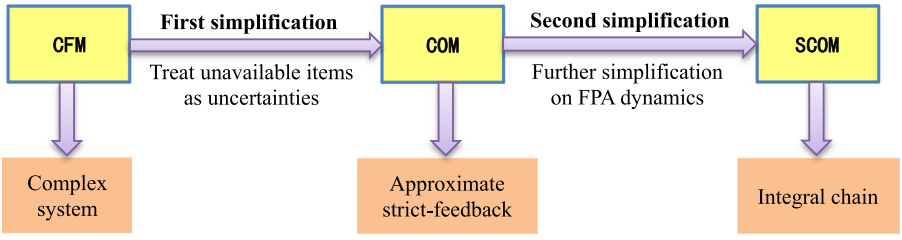

Fig. 3. Process of control-oriented modeling.

$$\begin{aligned}\dot{h} &= V\gamma\\ \dot{\gamma} &= f_\gamma + g_\gamma\theta + d_\gamma\\ \dot{\theta} &= Q\\ \dot{Q} &= f_q + g_q\delta_e + d_q\end{aligned} \tag{10}$$

The above equation along with the velocity loop in (5) forms the COM.

Remark 2. The definition of strict-feedback system can be found in [32]. It should be noted that (10) is not a real strict-feedback system because γ, θ appear in f_γ. So we call it approximate strict-feedback form.

Unlike model parameter uncertainties or external disturbances, the uncertainties caused by model simplifying are specific. The uncertainties actually consist of system states and inputs. For the need of control design and stability analysis, it is usually assumed that the uncertainties (including system states) or its derivatives are bounded [14,23] or constants in steady state [24]. In a similar way, we make the following assumption:

Assumption 1. For constant commands, d_V, d_γ, d_q can be seen as constant uncertainties. And for varying commands, d_V, d_γ, d_q can be seen as uncertainties with limited change rate, i.e., $\dot{d}_V$, $\dot{d}_\gamma$, $\dot{d}_q$ are bounded.

Remark 3. The assumption allows us to use adaptive methods to compensate the uncertainties. Since the uncertainties are expressions of system states, they will turn to constants when system achieves steady state. It is helpful to make steady-state analysis and get some valuable conclusions. However, the actual result should be validated by simulation. If the simulation results can support the analysis conclusions, then it proves the assumption is reasonable.

Since the COM is in a form of approximate strict-feedback, direct use of backstepping will cause the so-called "explosion of terms" problem [22]. This problem can be solved by approximate backstepping [14], dynamic surface [15,16,22] or command filter backstepping [17,18]. In this paper, we consider a new way to solve this problem by means of model simplification. Considering backstepping is applicable to strict-feedback system, and especially suitable for integral chain system, further simplification will be made to allow the use of backstepping. Recall the transformation from CFM to COM, we treat some intractable items as uncertainties to make it suitable for control design. Next we will make the second simplification to obtain the SCOM.

For the altitude loop (10), the main obstacle lies in the FPA dynamics. Along with the desired value γ_r, the FPA dynamics can be rewritten as

$$\dot{\tilde{\gamma}} = f_\gamma + g_\gamma\theta + d_\gamma - \dot{\gamma}_r \tag{11}$$

Here θ is the virtual control, with the coefficient g_γ being identically positive, indicating that the control direction is always positive. Therefore it is natural to think about scaling the coefficient g_γ to one and all the other items as uncertainties. Then (11) turns into

$$\dot{\tilde{\gamma}} = \theta + d_{\gamma 1} \tag{12}$$

where

$$d_{\gamma 1} = f_{\gamma 0} + d_\gamma - \theta - \dot{\gamma}_r \tag{13}$$

is the lumped uncertainty. This idea is also adopted in [16]. With this secondary simplification, the last three equations of the altitude loop become

$$\begin{aligned}\dot{\tilde{\gamma}} &= \theta + d_{\gamma 1}\\ \dot{\theta} &= Q\\ \dot{Q} &= f_q + g_q\delta_e + d_q\end{aligned} \tag{14}$$

It is an integral chain with matched and mismatched uncertainties, which is ready for the application of adaptive backstepping control. Eq. (14) along with the velocity loop in (5) and the first equation of the altitude loop in (10) constitute the SCOM.

In the same way as Assumption 1, the following assumption is made for preparation of control design and stability analysis:

Assumption 2. For constant commands, $d_{\gamma 1}$ can be seen as constant uncertainty. And for varying commands, $d_{\gamma 1}$ can be seen as uncertainty with limited change rate, i.e., $\dot{d}_{\gamma 1}$ is bounded.

From the control-oriented modeling process, we can summarize that the model simplification has the following advantages:

(1) Convenience. A direct benefit of model simplification is to make the model concise and more convenient for controller design. The simplifications we made serve for the controller design and cooperate with the control method to work. By treating the intractable items in the original model as uncertainties and compensate it through adaptive laws, we avoid the difficulties caused by unknown flexible states, input-output couplings and nonminimum phase problems and deal with them in an indirect way. As a result, we have achieved dynamic decoupling and dynamic compensation for flexible states as well as the stabilization of the unstable zero dynamics.
(2) Effectiveness. The simplified model is a nonlinear model with uncertainties, which retains some dominant feature of the original model and is closer to the original model than the model obtained by Jacobian linearization. On one hand, it allows us to take advantage of the powerful nonlinear control tools, e.g. the SCOM in this paper is suitable for adaptive backstepping control method. On the other hand, since the simplified model retains the dominant feature of the original model, the performance can be guaranteed when the controller is applied to the original model.

5. Approximate backstepping and dynamic surface control design based on COM

In this section, approximate backstepping and dynamic surface control design based on COM will be derived to serve as compare groups for adaptive backstepping control. As a preliminary, two lemmas will be introduced.

Lemma 1. *(LaSalle's Theorem [33]): Let $\Omega \subset D$ be a compact set that is positively invariant with respect to system $\dot{x} = f(x)$. Let $V: D \to R$ be a continuously differentiable function such that $\dot{V}(x) \le 0$ in Ω. Let E be the set of all points in Ω where $V(x) = 0$. Let M be the largest invariant set in E. Then every solution starting in Ω approaches M as $t \to \infty$.*

Lemma 2. *(Input-to-State Stability [33]): For the linear time-invariant system.*

$$\dot{x} = Ax + Bu$$

with a Hurwitz matrix A. Then the system is input-to-state stable and there exists positive constants λ, k such that

$$\|x(t)\| \le ke^{-\lambda t}\|x(0)\| + \frac{k\|B\|}{\lambda}\sup_{0\le\tau\le t}\|u(\tau)\|$$

In the following controller design process, we will first study the case of constant uncertainties and design adaptive law with the help of Lyapunov function (Lemma 1). Then bounded stability will be proved for varying uncertainties by using Lemma 2.

5.1. Control design for velocity loop

Consider the tracking error dynamic of velocity loop:

$$\dot{\tilde{V}} = f_V + g_V\phi + d_V - \dot{V}_r \tag{15}$$

It's a first-order system with matched uncertainty. Since d_V is unknown, we will use an estimated value $\hat{d}_V$ to replace it in the controller. The control input is designed as

$$\phi = \left(-k_V\tilde{V} - f_V - \hat{d}_V + \dot{V}_r\right)/g_V \tag{16}$$

where k_V is a positive constant. Denote the estimation error $\tilde{d}_V = d_V - \hat{d}_V$. Then the closed loop becomes

$$\dot{\tilde{V}} = -k_V\tilde{V} + \tilde{d}_V \tag{17}$$

To ensure the stability of the velocity loop, we use Lyapunov theory to design the update law of $\hat{d}_V$. Define the candidate Lyapunov function

$$W_1 = \frac{1}{2}\tilde{V}^2 + \frac{1}{2b_V}\tilde{d}_V^2 \tag{18}$$

where b_V is a positive constant. We will start from the constant commands so that d_V is a constant according to Assumption 1, then we have

$$\dot{\tilde{d}}_V = \dot{d}_V - \dot{\hat{d}}_V = -\dot{\hat{d}}_V \tag{19}$$

So the derivative of W_1 is

$$\begin{aligned}\dot{W}_1 &= \tilde{V}\dot{\tilde{V}} + \frac{1}{b_V}\tilde{d}_V\dot{\tilde{d}}_V \\ &= \tilde{V}\left(-k_V\tilde{V} + \tilde{d}_V\right) - \frac{1}{b_V}\tilde{d}_V\dot{\hat{d}}_V \\ &= -\tilde{V}\left(-k_V\tilde{V} + \tilde{d}_V\right) - \frac{1}{b_V}\tilde{d}_V\dot{\hat{d}}_V\end{aligned} \tag{20}$$

By choosing the update law of $\hat{d}_V$ as

$$\dot{\hat{d}}_V = b_V\tilde{V} \tag{21}$$

It follows that $\dot{W}_1 = -k_V\tilde{V}^2 \le 0$. Since $\dot{W}_1$ is negative semi-definite rather than negative definite, we need to use LaSalle's Theorem to analyze the stability. Solving $\dot{W}_1 = 0$ yields $\tilde{V} = 0$. If $\tilde{V} \equiv 0$, then $\dot{\tilde{V}} = 0$. According to (17), we can obtain $\tilde{d}_V = 0$. So the largest invariant set is $\left\{\left(\tilde{V}, \tilde{d}_V\right) \middle| \tilde{V} = 0, \tilde{d}_V = 0\right\}$. So according to Lemma 1, we have $\lim_{t\to\infty}\tilde{V} = 0$, $\lim_{t\to\infty}\tilde{d}_V = 0$, i.e., the closed loop of velocity is asymptotic stable.

When tracking varying commands, bounded stability rather than asymptotic stability can be proved by Lemma 2. According to (17) and (21), the closed loop of the velocity loop can be written as

$$\dot{e}_1 = A_1e_1 + B_1u_1 \tag{22}$$

where

$$e_1 = \begin{bmatrix}\tilde{V}\\ \tilde{d}_V\end{bmatrix},\ u_1 = \begin{bmatrix}0\\ \dot{d}_V\end{bmatrix},\ A_1 = \begin{bmatrix}-k_V & 1\\ -b_V & 0\end{bmatrix},\ B_1 = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} \tag{23}$$

It is not difficult to verify that A_1 is Hurwitz. So according to Lemma 2, we have

$$\|\tilde{V}(t)\| \le \|e_1(t)\| \le a_1\|e_1(0)\| + b_1 \tag{24}$$

with

$$a_1 = k_1e^{-\lambda_1 t},\ b_1 = \frac{k_1\|B_1\|}{\lambda_1}\sup_{0\le\tau\le t}\|u_1(\tau)\| \tag{25}$$

where λ_1, k_1 are positive constants. According to Assumption 1, $\left|\dot{d}_V\right|$ is bounded, then $\|u_1\|$ is bounded. Thus $\|\tilde{V}\|$ is bounded according to (24).

Remark 4. We adopt a strategic way to proceed with the stability proof. Based on the assumptions about the uncertainties, the stability proof goes step by step from the simple case of constant uncertainties to the more complex case of varying uncertainties. For constant uncertainties, we can get a nominal closed loop system which can be proved asymptotic stable by LaSalle's Theorem. And then we can turn to the more generalized case of varying uncertainties, where the resulted closed loop system can be seen as a perturbed system of the nominal closed loop system. Therefore it's easy to prove the bounded stability by Input-to-State Stability Theorem without knowing the bounds of the uncertainties. This strategy will be adopted repeatedly in the later design.

5.2. Control design for altitude loop

As mentioned before, the control loop of altitude loop is $\delta_e \to Q \to \theta \to \gamma \to h$. Here γ, θ, Q are treated as virtual control, with γ_r, θ_r, Q_r being their desired values to be designed. Define the corresponding tracking errors as

$$\tilde{\gamma} = \gamma - \gamma_r,\ \tilde{\theta} = \theta - \theta_r,\ \tilde{Q} = Q - Q_r \tag{26}$$

Then the dynamics of altitude loop (10) can be rewritten as

$$\begin{aligned}\dot{\tilde{h}} &= V\tilde{\gamma} + V\gamma_r - \dot{h}_r \\ \dot{\tilde{\gamma}} &= f_\gamma + g_\gamma\tilde{\theta} + g_\gamma\theta_r + d_\gamma - \dot{\gamma}_r \\ \dot{\tilde{\theta}} &= \tilde{Q} + Q_r - \dot{\theta}_r \\ \dot{\tilde{Q}} &= f_q + g_q\delta_e + d_q - \dot{Q}_r\end{aligned} \tag{27}$$

Thus the altitude loop is decomposed into four subsystems, with γ_r, θ_r, Q_r being the virtual control inputs of the first three equations respectively and δ_e being the control input of the last equation. So the control task becomes to design the virtual control laws γ_r, θ_r, Q_r and the real control input δ_e to stabilize the error dynamics of the altitude loop. It is worth noting that $\dot{\gamma}_r$, $\dot{\theta}_r$, $\dot{Q}_r$ appear in (27). Since γ_r, θ_r, Q_r are feedbacks of system states, their derivatives are difficult to get. Approximate backstepping and dynamic surface method will be used to solve this problem.

5.2.1. Method 1: approximate backstepping

By using approximate backstepping method, we will not contain the command derivatives $\dot{\gamma}_r$, $\dot{\theta}_r$, $\dot{Q}_r$ in the control laws of θ_r, Q_r, δ_e. The virtual control laws and real control input are designed as

$$\begin{aligned}\gamma_r &= \left(-k_h\tilde{h} + \dot{h}_r\right)/V\\ \theta_r &= \left(-k_\gamma\tilde{\gamma} - f_\gamma - \hat{d}_\gamma\right)/g_\gamma\\ Q_r &= -k_\theta\tilde{\theta} - g_\gamma\tilde{\gamma}\\ \delta_e &= \left(-k_q\tilde{Q} - \tilde{\theta} - f_q - \hat{d}_q\right)/g_q\end{aligned} \tag{28}$$

where $k_h > 0, k_\gamma > 0, k_\theta > 0, k_q > 0$. Substituting (28) into (27), the resulted closed loop system is

$$\begin{aligned}\dot{\tilde{h}} &= -k_h\tilde{h} + V\tilde{\gamma}\\ \dot{\tilde{\gamma}} &= -k_\gamma\tilde{\gamma} + g_\gamma\tilde{\theta} + \tilde{d}_\gamma - \dot{\gamma}_r\\ \dot{\tilde{\theta}} &= -k_\theta\tilde{\theta} - g_\gamma\tilde{\gamma} + \tilde{Q} - \dot{\theta}_r\\ \dot{\tilde{Q}} &= -k_q\tilde{Q} - \tilde{\theta} + \tilde{d}_q - \dot{Q}_r\end{aligned} \tag{29}$$

We can see that $\dot{\gamma}_r, \dot{\theta}_r, \dot{Q}_r$ are still remained in the closed loop system since we don't eliminate $\dot{\gamma}_r, \dot{\theta}_r, \dot{Q}_r$ in the control laws.

The adaptive laws are designed as

$$\begin{aligned}\dot{\hat{d}}_\gamma &= b_\gamma\tilde{\gamma}\\ \dot{\hat{d}}_q &= b_q\tilde{Q}\end{aligned} \tag{30}$$

with $b_\gamma > 0, b_q > 0$. Denote the estimation errors $\tilde{d}_\gamma = d_\gamma - \hat{d}_\gamma$, $\tilde{d}_q = d_q - \hat{d}_q$, then we have

$$\begin{aligned}\dot{\tilde{d}}_\gamma &= \dot{d}_\gamma - b_\gamma\tilde{\gamma}\\ \dot{\tilde{d}}_q &= \dot{d}_q - b_q\tilde{Q}\end{aligned} \tag{31}$$

Next we will analyze the stability of the whole closed loop system (29) and (31). The analysis will be conducted for constant commands and varying commands respectively.

As a preliminary of stability analysis and motivated by [14], the following assumption is made:

Assumption 3. For constant commands, $\dot{\gamma}_r, \dot{\theta}_r, \dot{Q}_r$ are regarded as zero. And for varying commands, $\dot{\gamma}_r, \dot{\theta}_r, \dot{Q}_r$ are bounded.

First, consider the case when tracking constant commands. Under Assumption 1 and Assumption 3, it can be known that $\dot{\gamma}_r = 0, \dot{\theta}_r = 0, \dot{Q}_r = 0, \dot{d}_\gamma = 0, \dot{d}_q = 0$. Denote $e_2 = \left[\tilde{\gamma}, \tilde{\theta}, \tilde{Q}, \tilde{d}_\gamma, \tilde{d}_q\right]^T$, then the closed loop system (29) and (31) can be written as follows:

$$\dot{\tilde{h}} = -k_h\tilde{h} + V\tilde{\gamma} \tag{32}$$

$$\dot{e}_2 = A_2e_2 \tag{33}$$

where

$$A_2 = \begin{bmatrix} -k_\gamma & g_\gamma & 0 & 1 & 0\\ -g_\gamma & -k_\theta & 1 & 0 & 0\\ 0 & -1 & -k_q & 0 & 1\\ -b_\gamma & 0 & 0 & 0 & 0\\ 0 & 0 & -b_q & 0 & 0\end{bmatrix} \tag{34}$$

Select the candidate Lyapunov function as

$$W_2 = \frac{1}{2}\tilde{\gamma}^2 + \frac{1}{2}\tilde{\theta}^2 + \frac{1}{2}\tilde{Q}^2 + \frac{1}{2b_\gamma}\tilde{d}_\gamma^2 + \frac{1}{2b_q}\tilde{d}_q^2 \tag{35}$$

Differentiating both sides of (35) along (33) yields

$$\dot{W}_2 = -k_\gamma\tilde{\gamma}^2 - k_\theta\tilde{\theta}^2 - k_q\tilde{Q}^2 \le 0 \tag{36}$$

Since $\dot{W}_2$ is negative semidefinite rather than negative definite, we need to use LaSalle's Theorem to analyze the stability. Solving $\dot{W}_2 = 0$ yields $\tilde{\gamma} = 0, \tilde{\theta} = 0, \tilde{Q} = 0$. If $\tilde{\gamma} \equiv 0, \tilde{\theta} \equiv 0, \tilde{Q} \equiv 0$, then $\dot{\tilde{\gamma}} = 0, \dot{\tilde{\theta}} = 0, \dot{\tilde{Q}} = 0$. According to (33), we can obtain $\tilde{d}_\gamma = 0, \tilde{d}_q = 0$. So the largest invariant set is $\left\{e_2 | e_2 = 0\right\}$. According to Lemma 1, we have $\lim_{t\to\infty} e_2 = 0$. Combine (32), we have $\lim_{t\to\infty}\tilde{h} = 0$. Therefore, the closed loop system of altitude loop is asymptotic stable when tracking constant commands.

Next, consider the case when tracking varying commands. In this case the closed loop system can be seen as a perturbed system of (32), (33), that is

$$\begin{aligned}\dot{\tilde{h}} &= -k_h\tilde{h} + V\tilde{\gamma}\\ \dot{e}_2 &= A_2e_2 + B_2^1u_2^1\end{aligned} \tag{37}$$

where $u_2^1 = \left[-\dot{\gamma}_r, -\dot{\theta}_r, -\dot{Q}_r, \dot{d}_\gamma, \dot{d}_q\right]^T$ and $B_2^1 \in R^{5\times5}$ is an identity matrix. Since (33) is asymptotic stable, it can be known that A_2 is Hurwitz according to Lyapunov's First Theorem. Since $-k_h$ and A_2 are both Hurwitz, it is input-to-state stable from u_2^1 to e_2 and from $V\tilde{\gamma}$ to $\tilde{h}$. According to Assumption 1 and Assumption 3, u_2^1 is bounded, which indicates that e_2 is bounded according to Lemma 2. So $\tilde{\gamma}$ is bounded, and thus $\tilde{h}$ is bounded, too. Therefore, the closed loop system of altitude loop is bounded stable when tracking varying commands.

5.2.2. Method 2: dynamic surface

By using dynamic surface, a first-order filter will be used to estimate the derivatives $\dot{\gamma}_r, \dot{\theta}_r, \dot{Q}_r$ as follows:

$$\begin{aligned}\dot{\gamma}_r &= \lambda_\gamma(\gamma_c - \gamma_r)\\ \dot{\theta}_r &= \lambda_\theta(\theta_c - \theta_r)\\ \dot{Q}_r &= \lambda_q(Q_c - Q_r)\end{aligned} \tag{38}$$

Here $\lambda_\gamma, \lambda_\theta, \lambda_q$ are positive filter parameters, and γ_c, θ_c, Q_c are the virtual control laws, with γ_r, θ_r, Q_r being their estimations. Pay attention that the meanings of the symbols γ_r, θ_r, Q_r are different from the previous section. Define the virtual control estimation errors as

$$e_\gamma = \gamma_r - \gamma_c, e_\theta = \theta_r - \theta_c, e_q = Q_r - Q_c \tag{39}$$

and $\tilde{\gamma}, \tilde{\theta}, \tilde{Q}$ are still defined as (26). Then the dynamics of altitude loop (10) can be rewritten as

$$\begin{aligned}\dot{\tilde{h}} &= V\tilde{\gamma} + Ve_\gamma + V\gamma_c - \dot{h}_r\\ \dot{\tilde{\gamma}} &= f_\gamma + g_\gamma\tilde{\theta} + g_\gamma e_\theta + g_\gamma\theta_c + d_\gamma - \dot{\gamma}_r\\ \dot{\tilde{\theta}} &= \tilde{Q} + e_q + Q_c - \dot{\theta}_r\\ \dot{\tilde{Q}} &= f_q + g_q\delta_e + d_q - \dot{Q}_r\end{aligned} \tag{40}$$

Since $\dot{\gamma}_r, \dot{\theta}_r, \dot{Q}_r$ can be obtained now, they can be eliminated. Different from (28), the virtual control laws and real control input are now designed as

$$\begin{aligned}\gamma_c &= \left(-k_h\tilde{h} + \dot{h}_r\right)/V\\ \theta_c &= \left(-k_\gamma\tilde{\gamma} - f_\gamma - \hat{d}_\gamma + \dot{\gamma}_r\right)/g_\gamma\\ Q_c &= -k_\theta\tilde{\theta} - g_\gamma\tilde{\gamma} + \dot{\theta}_r\\ \delta_e &= \left(-k_q\tilde{Q} - \tilde{\theta} - f_q - \hat{d}_q + \dot{Q}_r\right)/g_q\end{aligned} \tag{41}$$

Substituting (41) into (40), the resulted closed loop system is

$$\begin{aligned}\dot{\tilde{h}} &= -k_h\tilde{h} + V\tilde{\gamma} + Ve_\gamma\\ \dot{\tilde{\gamma}} &= -k_\gamma\tilde{\gamma} + g_\gamma\tilde{\theta} + \tilde{d}_\gamma + g_\gamma e_\theta\\ \dot{\tilde{\theta}} &= -k_\theta\tilde{\theta} - g_\gamma\tilde{\gamma} + \tilde{Q} + e_q\\ \dot{\tilde{Q}} &= -k_q\tilde{Q} - \tilde{\theta} + \tilde{d}_q\end{aligned} \tag{42}$$

The adaptive law is designed as same as (30).

Next we will analyze the stability of the closed loop system. The analysis will be conducted for constant commands and varying commands respectively.

Like Assumption 3 and motivated by [16], the following assumption is made:

Assumption 4. For constant commands, $\dot{\gamma}_c$, $\dot{\theta}_c$, $\dot{Q}_c$ are regarded as zero. And for varying commands, $\dot{\gamma}_c$, $\dot{\theta}_c$, $\dot{Q}_c$ are bounded.

According to (38), (39), the virtual control estimation error dynamics can be written as

$$\begin{aligned}\dot{e}_\gamma &= -\lambda_\gamma e_\gamma - \dot{\gamma}_c\\ \dot{e}_\theta &= -\lambda_\theta e_\theta - \dot{\theta}_c\\ \dot{e}_q &= -\lambda_q e_q - \dot{Q}_c\end{aligned} \tag{43}$$

It is obvious that each equation of (43) is input-to-state sable.

When tracking constant commands. Under Assumption 4, $\dot{\gamma}_c = 0$, $\dot{\theta}_c = 0$, $\dot{Q}_c = 0$, thus we have $\lim_{t\to\infty} e_\gamma = 0$, $\lim_{t\to\infty} e_\theta = 0$, $\lim_{t\to\infty} e_q = 0$ from (43). So when tracking constant commands, the closed loop system (42) equals to the nominal system (32), (33) and the asymptotic stable conclusion holds.

When tracking varying commands. With $e_2 = \left[\tilde{\gamma}, \tilde{\theta}, \tilde{Q}, \tilde{d}_\gamma, \tilde{d}_q\right]^T$, the closed loop system (42) and (31) can be written as follows:

$$\begin{aligned}\dot{\tilde{h}} &= -k_h\tilde{h} + V\tilde{\gamma} + Ve_\gamma\\ \dot{e}_2 &= A_2 e_2 + B_2^2 u_2^2\end{aligned} \tag{44}$$

where A_2 is the same as (34) and $u_2^2 = \left[g_\gamma e_\theta, e_q, 0, \dot{d}_\gamma, \dot{d}_q\right]^T$, $B_2^2 \in R^{5\times 5}$ is an identity matrix. Since $-k_h$ and A_2 are both Hurwitz, it is input-to-state stable from u_2^2 to e_2 and from $V\tilde{\gamma} + Ve_\gamma$ to $\tilde{h}$. According to Assumption 4, $\dot{\gamma}_c$, $\dot{\theta}_c$, $\dot{Q}_c$ are bounded, which indicates that e_γ, e_θ, e_q are bounded from (43). Thus u_2^2 is bounded, which indicates that e_2 is bounded from (44). So $\tilde{\gamma}$ is bounded, and $V\tilde{\gamma} + Ve_\gamma$ is bounded which indicates that $\tilde{h}$ is bounded from (44). Therefore the bounded stable conclusion holds.

6. Adaptive backstepping control design based on SCOM

Adaptive backstepping [32,34,35] is a powerful tool in dealing with nonlinear high-order systems with matched and mismatched uncertainties. As described in Section 4, the altitude loop is simply an integral chain with matched and mismatched uncertainties in the SCOM. Thus adaptive backstepping can be conveniently used.

The schematic diagram of the control architecture is shown in Fig. 4. The velocity loop is designed as same as Section 5.1 and the virtual control γ_r is designed the same as (28). Next we will concentrates on designing controller for the integral chain parts (14) by using adaptive backstepping method.

In the process of adaptive backstepping control, we need to design a virtual control law and an adaptive law in each step. The design follows a recursive procedure as shown in Fig. 5, where the derivative of the current virtual control which includes the current adaptive law will be used in calculating the next virtual control,

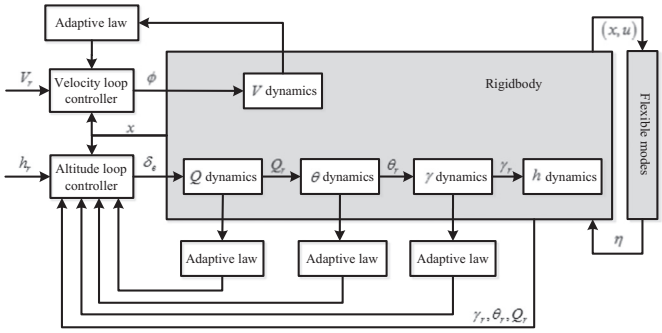

Fig. 4. Schematic diagram of the control architecture.

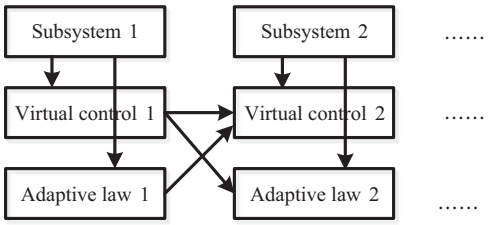

Fig. 5. Recursive procedure of adaptive backstepping.

and the current virtual control will be used in designing the next adaptive law.

First, consider the flight path angle tracking error dynamics

$$\dot{\tilde{\gamma}} = \tilde{\theta} + \theta_r + d_{\gamma 1} \tag{45}$$

Design the virtual control θ_r as

$$\theta_r = -k_\gamma\tilde{\gamma} - \hat{d}_{\gamma 1} \tag{46}$$

with $k_\gamma > 0$. Substituting (46) into (45) yields

$$\dot{\tilde{\gamma}} = -k_\gamma\tilde{\gamma} + \tilde{\theta} + \tilde{d}_{\gamma 1} \tag{47}$$

where $\tilde{d}_{\gamma 1} = d_{\gamma 1} - \hat{d}_{\gamma 1}$. Design the update law of $\hat{d}_{\gamma 1}$ as

$$\dot{\hat{d}}_{\gamma 1} = b_\gamma\tilde{\gamma} \tag{48}$$

with $b_\gamma > 0$. So the derivative of θ_r can be calculated as

$$\begin{aligned}\dot{\theta}_r &= -k_\gamma\dot{\tilde{\gamma}} - \dot{\hat{d}}_{\gamma 1}\\ &= -k_\gamma\left(\theta + d_{\gamma 1}\right) - b_\gamma\tilde{\gamma}\end{aligned} \tag{49}$$

Next consider the pitch angle tracking error dynamics

$$\begin{aligned}\dot{\tilde{\theta}} &= \dot{\theta} - \dot{\theta}_r\\ &= \tilde{Q} + Q_r + k_\gamma\theta + b_\gamma\tilde{\gamma} + d_{\theta 1}\end{aligned} \tag{50}$$

where

$$d_{\theta 1} = k_\gamma d_{\gamma 1} \tag{51}$$

is viewed as the lumped uncertainty in $\dot{\tilde{\theta}}$ for convenience. Design the virtual control Q_r as

$$Q_r = -k_\theta\tilde{\theta} - k_\gamma\theta - b_\gamma\tilde{\gamma} - \hat{d}_{\theta 1} - \tilde{\gamma} \tag{52}$$

with $k_\theta > 0$. Substituting (52) into (50) yields

$$\dot{\tilde{\theta}} = -k_\theta\tilde{\theta} + \tilde{Q} + \tilde{d}_{\theta 1} - \tilde{\gamma} \tag{53}$$

where $\tilde{d}_{\theta 1} = d_{\theta 1} - \hat{d}_{\theta 1}$. Design the update law of $\hat{d}_{\theta 1}$ as

$$\dot{\hat{d}}_{\theta 1} = b_\theta \tilde{\theta} \tag{54}$$

with $b_\theta > 0$. So the derivative of Q_r can be calculated as

$$\begin{aligned}\dot{Q}_r &= -k_\theta \dot{\tilde{\theta}} - k_\gamma \dot{\theta} - b_\gamma \dot{\tilde{\gamma}} - \dot{\hat{d}}_{\theta 1} - \dot{\bar{\gamma}} \\ &= -k_\theta \dot{\tilde{\theta}} - k_\gamma \dot{\theta} - \left(b_\gamma + 1\right)\dot{\tilde{\gamma}} - \dot{\hat{d}}_{\theta 1} \\ &= -k_\theta\left(Q + k_\gamma \theta + b_\gamma \tilde{\gamma} + d_{\theta 1}\right) - k_\gamma Q - \left(b_\gamma + 1\right)\left(\theta + d_{\gamma 1}\right) - b_\theta \tilde{\theta}\end{aligned} \tag{55}$$

Finally, consider the pitch rate tracking error dynamics

$$\begin{aligned}\dot{\tilde{Q}} &= \dot{Q} - \dot{Q}_r \\ &= f_q + g_q \delta_e + k_\theta\left(Q + k_\gamma \theta + b_\gamma \tilde{\gamma}\right) + k_\gamma Q + \left(b_\gamma + 1\right)\theta + b_\theta \tilde{\theta} + d_{q1}\end{aligned} \tag{56}$$

where

$$d_{q1} = d_q + k_\theta d_{\theta 1} + \left(b_\gamma + 1\right) d_{\gamma 1} \tag{57}$$

is viewed as the lumped uncertainty in $\dot{\tilde{Q}}$ for convenience.

Now the actual control intput δ_e appears in $\dot{\tilde{Q}}$. The final control law is designed as

$$\begin{aligned}\delta_e = g_q^{-1}\Big[&-k_q \tilde{Q} - f_q - k_\theta\left(Q + k_\gamma \theta + b_\gamma \tilde{\gamma}\right) - k_\gamma Q - \left(b_\gamma + 1\right)\theta \\ &- b_\theta \tilde{\theta} - \hat{d}_{q1} - \bar{\theta}\Big]\end{aligned} \tag{58}$$

with $k_q > 0$. Substituting (58) into (56) yields

$$\dot{\tilde{Q}} = -k_q \tilde{Q} + \tilde{d}_{q1} - \tilde{\theta} \tag{59}$$

where $\tilde{d}_{q1} = d_{q1} - \hat{d}_{q1}$. Design the update law of $\hat{d}_{q1}$ as

$$\dot{\hat{d}}_{q1} = b_q \tilde{Q} \tag{60}$$

with $b_q > 0$. Denote $e_3 = \left[\tilde{\gamma}, \tilde{\theta}, \tilde{Q}, \tilde{d}_{\gamma 1}, \tilde{d}_{\theta 1}, \tilde{d}_{q1}\right]^T$, $u_3 = \left[0, 0, 0, \dot{d}_{\gamma 1}, \dot{d}_{\theta 1}, \dot{d}_{q1}\right]^T$, then the closed loop system of altitude loop can be written as follows according to (47), (48), (53), (54), (59), (60).

$$\dot{e}_3 = A_3 e_3 + B_3 u_3 \tag{61}$$

where

$$A_3 = \begin{bmatrix} -k_\gamma & 1 & 0 & 1 & 0 & 0 \\ -1 & -k_\theta & 1 & 0 & 1 & 0 \\ 0 & -1 & -k_q & 0 & 0 & 1 \\ -b_\gamma & 0 & 0 & 0 & 0 & 0 \\ 0 & -b_\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & -b_q & 0 & 0 & 0 \end{bmatrix} \tag{62}$$

and $B_3 \in R^{6\times 6}$ is an identity matrix.

Since $u_3 = 0$ when tracking constant commands, we denote the nominal closed loop system as

$$\dot{e}_3 = A_3 e_3 \tag{63}$$

Select the candidate Lyapunov function as

$$W_3 = \frac{1}{2}\tilde{\gamma}^2 + \frac{1}{2}\tilde{\theta}^2 + \frac{1}{2}\tilde{Q}^2 + \frac{1}{2b_\gamma}\tilde{d}_{\gamma 1}^2 + \frac{1}{2b_\theta}\tilde{d}_{\theta 1}^2 + \frac{1}{2b_q}\tilde{d}_{q1}^2 \tag{64}$$

Differentiating both sides of (66) along (64) yields

$$\dot{W}_3 = -k_\gamma \tilde{\gamma}^2 - k_\theta \tilde{\theta}^2 - k_q \tilde{Q}^2 \le 0 \tag{65}$$

Since $\dot{W}_3$ is negative semidefinite, LaSalle's Theorem will be used to analyze the stability. Solving $\dot{W}_3 = 0$ yields $\tilde{\gamma} = 0$, $\tilde{\theta} = 0$, $\tilde{Q} = 0$. If $\tilde{\gamma} \equiv 0$, $\tilde{\theta} \equiv 0$, $\tilde{Q} \equiv 0$, then $\dot{\tilde{\gamma}} = 0$, $\dot{\tilde{\theta}} = 0$, $\dot{\tilde{Q}} = 0$. According to (63), we can obtain $\tilde{d}_{\gamma 1} = 0$, $\tilde{d}_{\theta 1} = 0$, $\tilde{d}_{q1} = 0$. So the largest invariant set is $\left\{e_3 | e_3 = 0\right\}$. According to Lemma 1, we have $\lim_{t\to\infty} e_3 = 0$. Combine (32), we have $\lim_{t\to\infty} \tilde{h} = 0$. Therefore, the nominal closed loop system of altitude loop is asymptotic stable. According to Lyapunov's First Theorem, it can be known that A_3 is Hurwitz. So system (61) is input-to-state stable from u_3 to e_3.

When tracking varying commands, $\dot{d}_{\gamma 1}$, $\dot{d}_q$ are bounded according to Assumption 1 and Assumption 2, then $\dot{d}_{\theta 1}$, $\dot{d}_{q1}$ are also bounded from (51) and (57). So u_3 is bounded which indicates that e_3 is bounded from (61), thus $\tilde{\gamma}$ is bounded. And (32) is input-to-state stable as well, so $\tilde{h}$ is bounded.

7. Simulations

To illustrate the effectiveness of the proposed methods, we have done some simulations in MATLAB/Simulink environment. The simulations focus on adaptive backstepping method, while approximate backstepping and dynamic surface will be used as comparisons.

7.1. Maneuver simulation

First, we will design the commands V_r, h_r. To maintain satisfactory performance of the scramjet engine, the dynamic pressure is expected to keep constant [28], i.e., $\dot{\bar{q}} = 0$. Because $\bar{q} = \rho V^2/2$, we have

$$\dot{\bar{q}} = \left(\dot{\rho} V^2 + 2\rho V \dot{V}\right)/2 = V\left(\dot{\rho} V + 2\rho \dot{V}\right)/2 = 0 \tag{66}$$

That is

$$\dot{\rho} V + 2\rho \dot{V} = 0 \tag{67}$$

And because $\rho = \rho_0 \exp\left[-\left(h - h_0\right)/h_s\right]$, it follows that

$$\dot{\rho} = -\rho \dot{h}/h_s \tag{68}$$

Substituting (68) into (67) yields

$$\dot{h} = 2h_s \dot{V}/V \tag{69}$$

So if V_r is given, then h_r is determined by

$$\dot{h}_r = 2h_s \dot{V}_r/V_r \tag{70}$$

The commands are designed as: In the first 50 s, the vehicle flights with constant commands $V_r = 8000ft/s$, $h_r = 80000ft$. Then the velocity rises to 10,000 ft/s, the trajectory of V_r is obtained by filtering the step signal with a third order low pass filter $1/\left(8s + 1\right)^3$. And the corresponding trajectory of h_r is obtained according to (70). The simulation time is 200 s and the simulation step is 0.01 s. The simulation results of adaptive backstepping are shown in Figs. 6–11 and a comparison between approximate backstepping, dynamic surface, and adaptive backstepping is shown in Fig. 12.

Fig. 6 shows the curves of velocity and altitude and their tracking errors. From the top part we can see that the real trajectories of V, h coincide with the reference trajectories V_r, h_r. The bottom part shows the tracking errors. In the time interval 0–50 s and 120–200 s, the commands V_r, h_r are constants, and we can see that the tracking errors converge to zero. Moreover, in the time interval 50–120 s, the commands V_r, h_r are varying and the tracking errors remain small. Therefore the control objective proposed in Section 2 is well realized and the results coincide well with the analysis conclusions in Section 6. It demonstrates the effectiveness of model simplification and adaptive backstepping control method.

Fig. 7 shows the dynamic pressure throughout the flight, from which we can see that the dynamic pressure changes very little

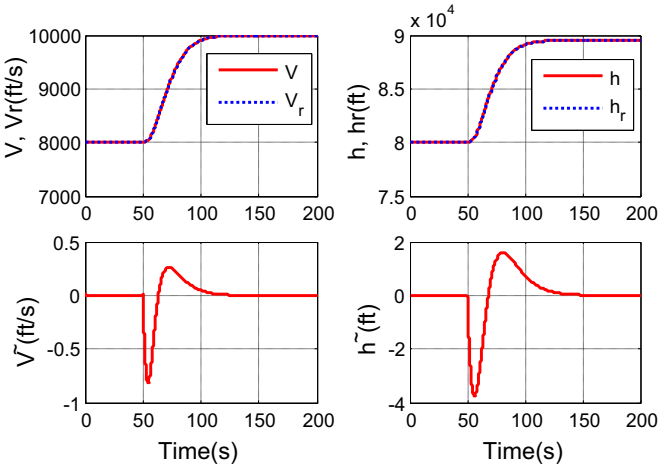

Fig. 6. Tracking results of velocity and altitude.

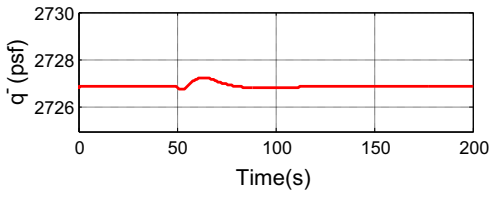

Fig. 7. Dynamic pressure.

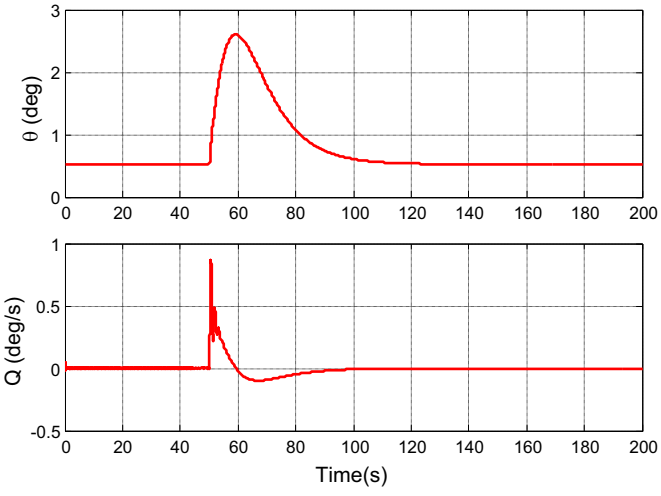

Fig. 8. Internal dynamics: pitch angle and pitch rate.

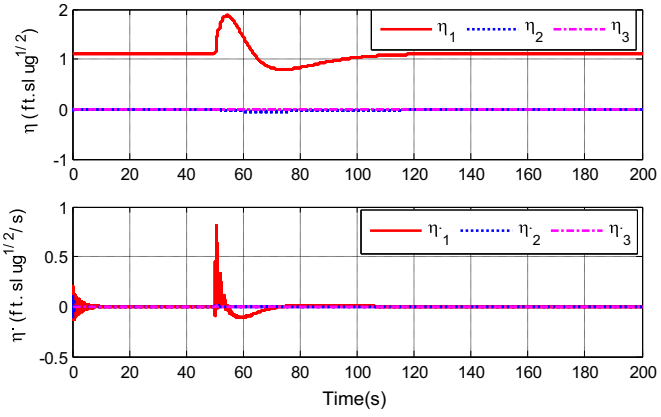

Fig. 9. Flexible modes.

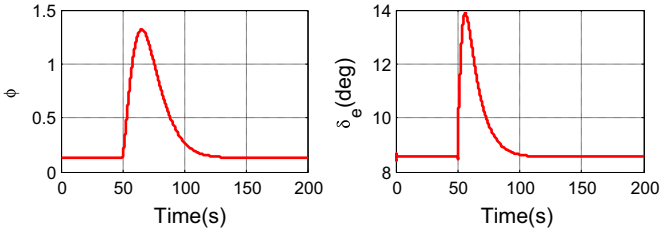

Fig. 10. Control inputs: fuel air ratio and elevator angular deflection.

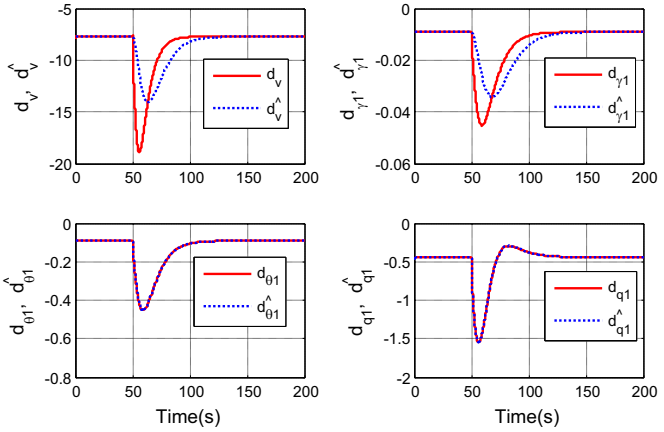

Fig. 11. Uncertainties and their estimated values.

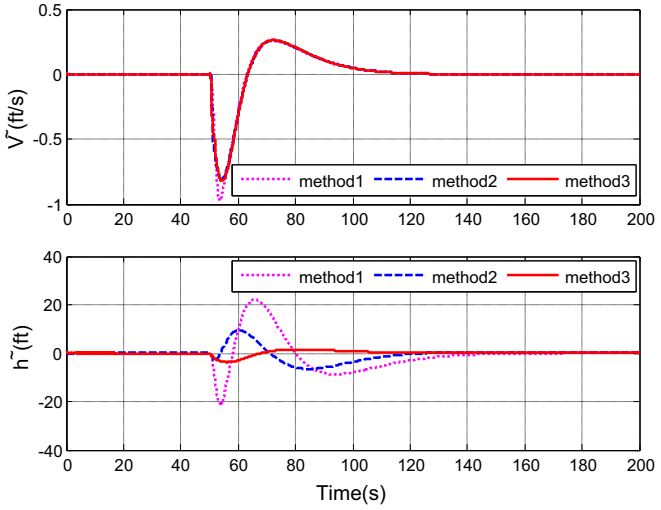

Fig. 12. Tracking errors comparison: method1 – approximate backstepping, method 2 – dynamic surface, method3 – adaptive backstepping.

and always stays in the range [2726,2728] psf. So the goal of keeping the dynamic pressure constant is achieved.

Fig. 8 shows the simulation results of the internal dynamics θ, Q. We can see that both of them remain bounded throughout the flight. This demonstrates that the nonminimum phase problem is solved successfully by using extended control loop, which guarantees both output tracking and internal stability.

Fig. 9 shows the simulation results of the flexible modes. We can see that all the flexible modes remain bounded. When vehicle climbs at the 50th s, the flexible modes are excitated but soon tend to be stable.

Fig. 10 shows the simulation results of the control inputs. We can see that both the fuel air ratio ϕ and the elevator angular deflection δ_e change smoothly. This indicates that the controller has a good engineering practicality.

Fig. 11 shows the estimated values of the uncertainties $\hat{d}_V$, $\hat{d}_{\gamma 1}$, $\hat{d}_{\theta 1}$, $\hat{d}_{q1}$. Since the uncertainties actually compose of system states, their true values d_V, $d_{\gamma 1}$, $d_{\theta 1}$, d_{q1} are known in simulation and are also shown in Fig. 11. On one hand, compared with Fig. 6, we can see that the uncertainties are constants when the commands keep unchanged. This validates that the assumptions we made are reasonable. On the other hand, we can see that the estimated values converge to the true values of the uncertainties when they turn to constants. This indicates that the adaptive laws work well.

Fig. 12 shows the comparison of the tracking errors for approximate backstepping, dynamic surface and adaptive backstepping. It can be seen that adaptive backstepping has the minimum tracking errors, especially in altitude tracking. It may because that adaptive backstepping method uses backstepping method directly rather than neglect (approximate backstepping) or estimate (dynamic surface) the command derivatives.

7.2. Monte-Carlo test

In the previous simulations, model parameter uncertainties are not considered. However, in reality, the model parameters are not exactly known. The values we used in the controller are their nominal values whereas their true values in the model may deviate from the nominal values within a certain range.

So we should do plenty of simulations to ensure the controller can maintain stable in a real flight. This is the so-called Monte-Carlo simulation, which can test the robustness of the controller. Here we consider 63 parameters of uncertainties as show in Table 1.

In each simulation, the controller is not changed while all parameters upon of the model are selected within ±10% range of their nominal values and obey uniform distribution. Take 1000 simulations using the same reference trajectories as before. The results are shown in Figs. 13–15. Each figure shows the tracking results of system outputs on the top and the control inputs on the bottom.

From Fig. 13 and Fig. 14, it can be seen that both approximate backstepping and dynamic surface lead to divergence of system outputs and control inputs. In contrast, adaptive backstepping exhibits good robustness. From Fig. 15 we can see that the tracking errors of the two outputs both remain small and converge to zero in the time interval 0–50 s and 120–200 s when the commands V_r, h_r are constants, and the control inputs remain bounded at the same time.

8. Conclusions

This paper investigates the nonminimum phase problem of hypersonic vehicle. We find out the relationship between nonminimum phase and backstepping. By extending the control loop to cover the internal dynamics, a stable nonlinear controller can be obtained through backstepping. Based on the extended control loop, the nonminimum phase problem of hypersonic vehicle is solved by model simplification and adaptive backstepping. Further work can be focused on extending this method to general nonminimum phase system and applied to other nonminimum phase systems. What's more, from the controller design process in this paper, we can see that the model is not the more accurate the

Table 1
Uncertain parameters of hypersonic vehicle.

Vehicle parameters	S, $\bar{c}$, m, I_{yy}
Lift coefficients	C_L^{α}, $C_L^{\delta_e}$, C_L^0, $C_L^{\eta 1}$, $C_L^{\eta 2}$, $C_L^{\eta 3}$
Drag coefficients	$C_D^{\alpha 2}$, C_D^{α}, $C_D^{\delta_e 2}$, $C_D^{\delta_e}$, C_D^0, $C_D^{\eta 1}$, $C_D^{\eta 2}$, $C_D^{\eta 3}$
Thrust coefficients	$C_T^{\phi\alpha 3}$, $C_T^{\phi\alpha 2}$, $C_T^{\phi\alpha}$, C_T^{ϕ}, C_T^3, C_T^2, C_T^1, C_T^0, $C_T^{\eta 1}$, $C_T^{\eta 2}$, $C_T^{\eta 3}$
Pitch moment coefficients	$C_M^{\alpha 2}$, C_M^{α}, C_M^0, $C_M^{\delta_e}$, $C_M^{\eta 1}$, $C_M^{\eta 2}$, $C_M^{\eta 3}$
Generalized forces coefficients	$N_1^{\alpha 2}$, N_1^{α}, N_1^0, $N_1^{\delta_e}$, $N_1^{\eta 1}$, $N_1^{\eta 2}$, $N_1^{\eta 3}$
	$N_2^{\alpha 2}$, N_2^{α}, N_2^0, $N_2^{\delta_e}$, $N_2^{\eta 1}$, $N_2^{\eta 2}$, $N_2^{\eta 3}$
	$N_3^{\alpha 2}$, N_3^{α}, N_3^0, $N_3^{\delta_e}$, $N_3^{\eta 1}$, $N_3^{\eta 2}$, $N_3^{\eta 3}$
Other parameters	z_T, ρ_0, g, ω_1, ω_2, ω_3

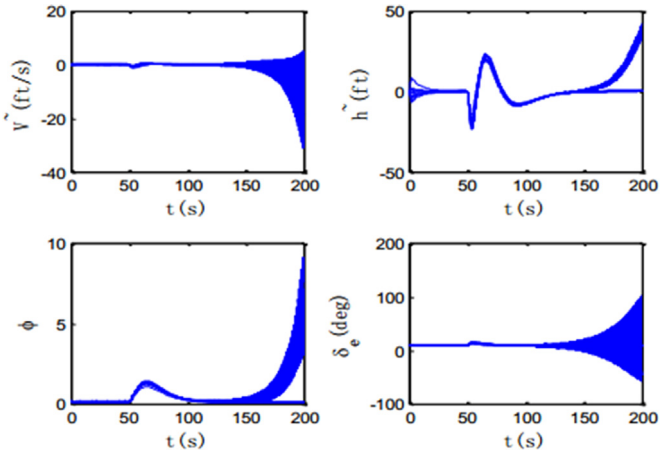

Fig. 13. Monte-Carlo simulation results of approximate backstepping.

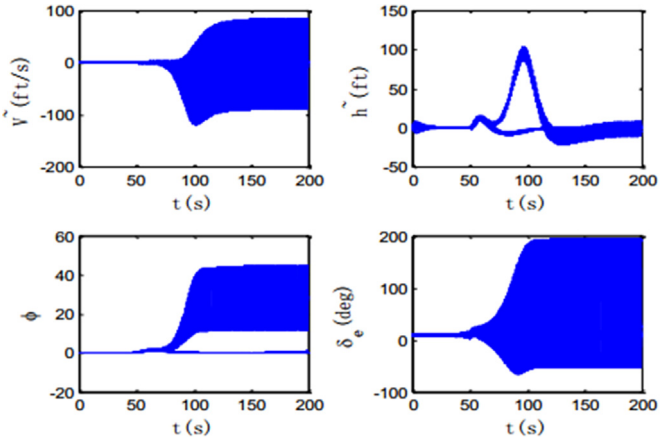

Fig. 14. Monte-Carlo simulation results of dynamic surface.

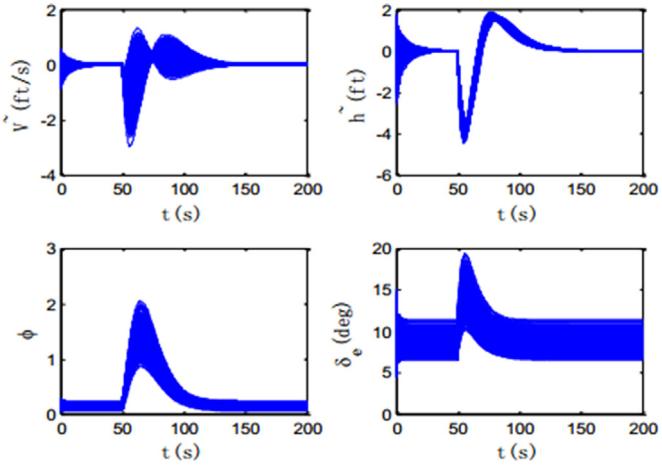

Fig. 15. Monte-Carlo simulation results of adaptive backstepping.

better for controller design. Sometimes it's necessary to simplify the original complex model into a simpler one so that a certain method can be applicable. Control-oriented modeling technique provides a useful tool for complex system control design, even when the system is nonminimum phase.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Nos. 61273092, 61673294 and 61503323), China Scholarship Council (No. 201606250160), China Postdoctoral Science Foundation (No. 2015M571282), Qinhuangdao Science and Technology Project (No. 201502A178) and Ministry of Education Equipment Development Fund (6141A02033311).

References

[1] Shkolnikov IA, Shtessel YB. Aircraft nonminimum phase control in dynamic sliding manifolds. J Guid Control Dyn 2001;24(3):566–72.
[2] Bolender M, Doman D. Flight path angle dynamics of air-breathing hypersonic vehicles. AIAA Guid Navig Control Conf Exhib 2006:6692.
[3] Stengel RF. Flight dynamics. New Jersey: Princeton University Press; 2015.
[4] Hag JB, Bernstein DS. Nonminimum-phase zeros-much to do about nothing-classical control-revisited part II. IEEE Control Syst Mag 2007;27(3):45–57.
[5] Tomlin C, Lygeros J, Benvenuti L, Sastry S. Output tracking for a non-minimum phase dynamic CTOL aircraft model. In: Proceedings of the 34th IEEE conference on decision and control. IEEE. Vol. 2; 1995, p. 1867–1872.
[6] Fiorentini L, Serrani A, Bolender MA, Doman DB. Nonlinear control of non-minimum phase hypersonic vehicle models. In: Proceedings of the Ameerican Control Conference (ACC'09). IEEE; 2009. p. 3160–3165.
[7] Isidori A. Nonlinear control systems. Heidelberg: Springer Science & Business Media; 2013.
[8] Fiorentini L, Serrani A. Adaptive restricted trajectory tracking for a non-minimum phase hypersonic vehicle model. Automatica 2012;48(7):1248–61.
[9] Qiu L, Davison EJ. Performance limitations of non-minimum phase systems in the servomechanism problem. Automatica 1993;29(2):337–49.
[10] Parker J, Serrani A, Yurkovich S, Bolender M, Doman D. Approximate feedback linearization of an air-breathing hypersonic vehicle. AIAA Guid Navig Control Conf Exhib 2006:6556.
[11] Parker JT, Serrani A, Yurkovich S, Bolender MA, Doman DB. Control-oriented modeling of an air-breathing hypersonic vehicle. J Guid Control Dyn 2007;30(3):856–69.
[12] Hauser J, Sastry S, Meyer G. Nonlinear control design for slightly non-minimum phase systems: application to V/STOL aircraft. Automatica 1992;28(4):665–79.
[13] Gopalswamy S, Karl Hedrick J. Tracking nonlinear non-minimum phase systems using sliding control. Int J Control 1993;57(5):1141–58.
[14] Zong Q, Wang F, Tian B, Wang J. Robust adaptive approximate backstepping control design for a flexible air-breathing hypersonic vehicle. J Aerosp Eng 2014;28(4):4014107.
[15] Xu B, Huang X, Wang D, Sun F. Dynamic surface control of constrained hypersonic flight models with parameter estimation and actuator compensation. Asian J Control 2014;16(1):162–74.
[16] Zong Q, Wang F, Tian B, Su R. Robust adaptive dynamic surface control design for a flexible air-breathing hypersonic vehicle with input constraints and uncertainty. Nonlinear Dyn 2014;78(1):289–315.
[17] Ji Y, Zong Q, Zhou H. Command filtered back-stepping control of a flexible air-breathing hypersonic flight vehicle. Proc Inst Mech Eng Part G J Aerosp Eng 2014;228(9):1617–26.
[18] Xu B, Wang S, Gao D, Zhang Y, Shi Z. Command filter based robust nonlinear control of hypersonic aircraft with magnitude constraints on states and actuators. J Intell Robot Syst 2014;73(1–4):233.
[19] Srinivasan B, Huguenin P, Bonvin D. Global stabilization of an inverted pendulum–control strategy and experimental verification. Automatica 2009;45(1):265–9.
[20] Benosman M, Le Vey G. Stable inversion of SISO nonminimum phase linear systems through output planning: an experimental application to the one-link flexible manipulator. IEEE Trans Control Syst Technol 2003;11(4):588–97.
[21] Consolini L, Tosques M. A minimum phase output in the exact tracking problem for the nonminimum phase underactuated surface ship. IEEE Trans Autom Control 2012;57(12):3174–80.
[22] Swaroop D, Hedrick JK, Yip PP, Gerdes JC. Dynamic surface control for a class of nonlinear systems. IEEE Trans Autom Control 2000;45(10):1893–9.
[23] Zong Q, Wang J, Tao Y. Adaptive high-order dynamic sliding mode control for a flexible air-breathing hypersonic vehicle. Int J Robust Nonlinear Control 2013;23(15):1718–36.
[24] Sun H, Li S, Yang J, Guo L. Non-linear disturbance observer-based back-stepping control for airbreathing hypersonic vehicles with mismatched disturbances. IET Control Theory Appl 2014;8(17):1852–65.
[25] Bolender MA, Doman DB. Nonlinear longitudinal dynamical model of an air-breathing hypersonic vehicle. J Spacecr Rocket 2007;44(2):374–87.
[26] Fiorentini L, Serrani A, Bolender MA, Doman DB. Nonlinear robust adaptive control of flexible air-breathing hypersonic vehicles. J Guid Control Dyn 2009;32(2):402–17.
[27] Sigthorsson D, Jankovsky P, Serrani A, Yurkovich S, Bolender M, Doman DB. Robust linear output feedback control of an airbreathing hypersonic vehicle. J Guid Control Dyn 2008;31(4):1052–66.
[28] Fiorentini L. Nonlinear adaptive controller design for air-breathing hypersonic vehicles; 2010.
[29] Fan H, Liu B, Wang W, Wen C. Adaptive fault-tolerant stabilization for nonlinear systems with Markovian jumping actuator failures and stochastic noises. Automatica 2015;51:200–9.
[30] Shen M, Ye D. Improved fuzzy control design for nonlinear Markovian-jump systems with incomplete transition descriptions. Fuzzy Sets Syst 2013;217:80–95.
[31] Shen M, Park JH, Ye D. A separated approach to control of Markov jump nonlinear systems with general transition probabilities. IEEE Trans Cybern 2016;46(9):2010–8.
[32] Krstic M, Kanellakopoulos I, Kokotovic PV. Nonlinear and adaptive control design.New Jersey: Wiley; 1995.
[33] Khalil HK. Nonlinear systems.New Jersey: Prentice-Hall; 1996.
[34] Zhao X, Shi P, Zheng X, Zhang J. Intelligent tracking control for a class of uncertain high-order nonlinear systems. IEEE Trans Neural Netw Learn Syst 2016;27(9):1976–82.
[35] Zhang J, Zhao X, Huang J. Synchronization control of neural networks with state-dependent coefficient matrices. IEEE Trans Neural Netw Learn Syst 2016;27(11):2440–7.