

Article

Attitude-Tracking Control for Over-Actuated Tailless UAVs at Cruise Using Adaptive Dynamic Programming

Zihou He , Jianbo Hu, Yingyang Wang , Jiping Cong , Yuan Bian and Linxiao Han

Equipment Management and Unmanned Aerial Vehicle Engineering College, Airforce Engineering University, Xi'an 710072, China

* Correspondence: hezihou9551@163.com (Z.H.); wangyingyang@outlook.com (Y.W.)

Abstract: Using adaptive dynamic programming (ADP), this paper presents a novel attitude-tracking scheme for over-actuated tailless unmanned aerial vehicles (UAVs) that integrates control and control allocation while accounting for nonlinearity and nonaffine control inputs. The proposed method uses the idea of nonlinear dynamic inversion to create an augmented system and converts the optimal tracking problem into an optimal regulation problem using a discounted performance function. Drawing inspiration from incremental control, this method achieves optimal tracking control for the nonaffine system by simply using a critic-only structure. Moreover, the unique design of the performance function ensures robustness against model uncertainties and external disturbances. The ADP method was found to outperform traditional control architectures that separate control and control allocation, achieving the same level of attitude-tracking performance through a more optimized approach. Furthermore, unlike many recent optimal controllers for nonaffine systems, our method does not require any model identifiers and demonstrates robustness. The superiority of the ADP-based approach is verified through two simulated scenarios, and its internal mechanism is further discussed. The theoretical analysis of robustness and stability is also provided.

Keywords: attitude tracking; tailless unmanned aerial vehicle; adaptive dynamic programming; UAV flight simulation; nonlinear flight control



Citation: He, Z.; Hu, J.; Wang, Y.; Cong, J.; Bian, Y.; Han, L.

Attitude-Tracking Control for Over-Actuated Tailless UAVs at Cruise Using Adaptive Dynamic Programming. *Drones* **2023**, *7*, 294. <https://doi.org/10.3390/drones7050294>

Academic Editors: Zeashan H. Khan, Imran Mir and Syed Tauqeer Ul Islam Rizvi

Received: 4 March 2023

Revised: 21 April 2023

Accepted: 24 April 2023

Published: 27 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The tailless unmanned aerial vehicle (UAV) has garnered immense attention due to its promising potential in both civil and military aviation. Its superior aerodynamic efficiency in comparison to traditional designs offers benefits such as improved voyage, carrying capacity, and stealth performance. This has led to the emergence of several tailless UAVs such as Boeing's X-45 A, X-45B/C, Lockheed Martin's RQ-170 Sentinel, BAE's Taranis, and Dassault's NEURON.

The Innovative Control Effector (ICE) aircraft [1,2], developed through research by Lockheed Martin, stands out among the latest tailless aerial vehicles. ICE is equipped with as many as 11 effectors relocated compactly on its main wing, which makes the system over-actuated. Such a unique layout was designed to investigate and measure the aerodynamics and performance of various low-observable tailless configurations using innovative control effectors. After decades of research, ICE has been found to have excellent maneuverability and stealth performance, making it a good choice for future UAV design.

The unique configuration of ICE allows its effectors to achieve goals beyond providing aerodynamic moments, such as minimizing drag or maximizing lift [3]. However, this configuration also poses challenges for control. Like other tailless vehicles, ICE suffers from problems such as poor static stability and coupling between longitudinal and lateral dynamics. Additionally, the redundant effectors of ICE require dealing with the control allocation problem, which involves selecting appropriate effectors and providing deflection commands to generate the required moments. However, the compact layout of ICE's

effectors results in strong coupling effects between them, causing the control inputs to appear nonlinear in the system, which means that the system is nonaffine. As a result, the control allocation of ICE is an extremely challenging task.

Since the proposal of the concept of ICE, researchers in the flight control field have been paying constant attention to it [4–6]. In 2017, Niestroy et al. [7] published detailed aerodynamic data for ICE, which enabled the construction of a highly precise control-oriented nonlinear model and the development of advanced control algorithms. Many researchers have developed different control algorithms for the nonlinear ICE model. In a recent study, He et al. [8] proposed an altitude tracker for ICE using the well-known decoupling conditions for nonaffine systems [9–14], while other researchers prefer incremental control methods.

The principle behind incremental control methods is timescale separation, which makes use of Taylor expansion. Incremental control methods can transform nonaffine systems into incremental affine forms. Therefore, the complexity of nonlinear optimization in CA can be avoided. This approach is highly effective for dealing with nonaffine systems [15]. Recently, there have been several advancements in incremental control for ICE. Stolk et al. [3] proposed a minimum drag CA method based on incremental nonlinear dynamic inversion. Matamoros [16] implemented an incremental nonlinear CA in ICE, resulting in improved tracking and CA performance. Sun et al. [17] improved the CA of ICE using hierarchical multi-objective optimization and adaptive incremental backstepping. Additionally, He et al. [14] extended the incremental control to the outer-loop control of ICE trajectory tracking using the pseudo-control hedging technique and relaxing the need for a timescale separation principle.

The reason incremental control is effective for nonaffine systems is it makes good use of the partial derivative of control inputs, $\frac{\partial f(x,u)}{\partial u}$. This is obtained through the digital differentiation of aerodynamic data. It is worth noting that obtaining the partial derivative of control inputs may be challenging in some systems. However, with the advancements in wind-tunnel tests, we can obtain more accurate and economic aerodynamic data for flight control. When combined with model identification techniques, incremental control has great potential for the future.

Most of the aforementioned control methods separate the command tracking and control allocation (CA) tasks. The command tracker provides the aerodynamic coefficients command τ_c to ensure that the reference signal x_c is accurately tracked, while the CA determines the specific effector deflection u based on the aerodynamic coefficients command and its objective function. This framework is highly convenient for incorporating established flight control algorithms, and the CA can be viewed as an optimization problem that can leverage the well-established optimization theory. As a result, this framework is preferred by most researchers.

However, there are some aspects of the above framework that could be improved. The obvious drawback is that the CA is designed to minimize the objective function consisting of the moments tracking error $\tau_e = \tau - \tau_c$ and the effector deflection u . From an input and output perspective, the moments tracking error is only an intermediate value, and what truly matters is the reference signal tracking error $x_e = x - x_c$. Therefore, the ideal objective function should take both x_e and the effector deflection into consideration, instead of just τ_e . Additionally, most existing flight control algorithms are based on Lyapunov theory and can only take into account the convergence of x_e . To make these algorithms compatible with the over-actuated UAV, the above framework must be adopted, and the second goal is left to the CA.

Meanwhile, the above framework takes two steps to give effector deflection, increasing the computational time. Hou et al. [18] introduced the recurrent neural network in CA and claimed that the recurrent neural network model could be solved in parallel to meet the real-time requirement. Still, this approach has only been validated in a linearized model, where the computational load is inherently small.

Hence, it is imperative to develop more reasonable and effective frameworks that abandon the meaningless intermediate values, therefore ensuring convergence of x_e and achieving the second goal in one step. Optimal control is a promising option in this regard. Unlike the objective function of CA that only considers τ_e and the second goal, the performance function of optimal control can incorporate the tracking error and any other desired second goals. This means that the command tracking and CA can be described using a single equation. However, solving the nonlinear Hamilton–Jacobi–Bellman (HJB) equation remains a formidable challenge.

Adaptive dynamic programming (ADP) provides new ideas for solving the nonlinear HJB equation. ADP is a heuristic algorithm for solving optimal control. Compared with other heuristic algorithms, such as reinforcement learning, ADP is supported by optimal control theory, so it shows better convergence and stability and is more suitable for flight control. The first application of ADP in optimal control could be seen in a study by Werbos [19]. ADP's basic idea is to use sampling data to drive a neural network to approximate the optimal value function. In this way, ADP turns the backward-in-time dynamic programming process into a forward-in-time manner and greatly expands the application of optimal control. For theoretical studies of ADP, Wei and Liu [20] give the stability analysis of policy iterative ADP, and the stability proof of value iterative ADP is given by Al-Tamimi and Lewis [21]. Moreover, the researchers also proposed different frameworks of ADP, such as heuristic dynamic programming [22], dual heuristic programming [23], and globalized dual heuristic programming [24]. These studies lay the foundation of ADP, and a more detailed review of recent studies on ADP can be found in the paper by Liu et al. [25]

Model identification is a commonly adopted technique in recent applications of ADP in practical systems [26,27]. Model identification is an effective method for enhancing the robustness of ADP, but it requires introducing an identifier network. Compared to basic ADP, which uses only a critical network to approximate the value function, the incorporation of additional networks significantly increases the computational complexity. Therefore, approaches to alleviate the computational burden, such as the event-trigger technique [28], are necessary for these methods.

However, it is often overlooked that the optimal control itself could be robust with an appropriate design of performance function. This idea is illustrated in a book by Lin [29], and the author systematically discusses how to handle disturbance and model uncertainty in an optimal control way. This way, the ADP could enjoy robustness while avoiding heavy computational burdens. However, the author also points out that it is still an open question how to apply similar approaches to a nonaffine system. Most ADPs are developed to address the optimal regulation problem, but for flight control, the optimal tracking control has a more practical use. With the development of the aviation industry, modern flight control is no longer satisfied with just ensuring flight stability. Many researchers [30–34] began to consider how to track the command signal optimally.

In the control field, optimal tracking control has attracted increased attention. One of the most common optimal tracking methods is the combination of feedforward control and feedback control [35–40]. The feedforward control is a traditional steady-state tracking controller to ensure the command reference signal is tracked. ADP is used in the feedback control to stabilize the transient error optimally. With the help of the traditional steady-state tracking controller, this optimal tracker shows good stability. Nevertheless, this optimal tracker is not suitable for ICE. Designing a traditional steady-state tracking controller for ICE has already been an arduous task, and the control allocation still needs to be considered in this process.

Some ADP-based optimal trackers do not rely on the feedforward control [41–43]. These studies applied a discounted performance function to ensure the boundness of the optimal value function in the infinite-time process and constructed an augmented system using the error dynamic and reference signal dynamic to transform the tracking problem into the regulation problem. However, the dynamic of the reference signal is unavailable in these methods, limiting the use of these ADPs.

From the other view, nonlinear dynamic inversion [44], as a tried-and-tested control method in the flight control area, constructs the dynamic of the desired signal using state value and command signal, which provides a new idea to overcome this drawback. Moreover, these ADPs also cannot address the nonaffine system, so they cannot be directly used for ICE.

To apply ADP in ICE, the nonaffine control input must be considered. Recent optimal trackers for nonaffine systems can be grouped into two types. One type is mainly for single-input systems, which decouple the nonaffine system into an affine system with model uncertainties [45–47], but it is not easy to extend such a method to the multi-input system. Using decoupling conditions, these methods show robustness, but neglecting model details also makes their optimization performance poor. The other type uses the other neural network, known as the actor network, to handle nonlinearity in control input and update the policy through gradient-base algorithm [48–50]. This method performs well but also needs more data and training, which undoubtedly increases the computational burden. Of course, some tricks commonly used in reinforcement learning [34] could also help improve the convergence rate of the method, but this also causes the lack of stability proofs.

Motivated by the aforementioned studies, this article proposes a critic-only ADP technique for the attitude tracking of ICE featured by nonaffine control inputs and redundant effectors. Through ADP, our approach integrates control and control allocation so that the same performance can be achieved at a cost less than conventional methods. By the idea of nonlinear dynamic inversion, an augmented system is constructed. The optimal tracking problem is transformed into an optimal regulation problem with discounted performance function, and the command dynamic is avoided. Inspired by the successful use of $\frac{\partial f(x,u)}{\partial u}$ in incremental control, we introduce $\frac{\partial f(x,u)}{\partial u}$ into APD, letting our method handle the nonaffine system in a simple way. Moreover, this article proves that for the control of the nonaffine system, the robust tracking problem could be equivalent to the optimal tracking problem with an augmented cost. This provides another way to improve the robustness of ADP, and complex model identification methods can be avoided.

The rest of the paper is arranged as follows: Section 2 introduces the aerodynamic model of the UAV. Section 3 gives the problem formulation and shows at the theoretical level that the optimal control with a specially designed performance function is equivalent to robust control. Section 4 presents the control scheme and stability analysis. Section 5 presents two simulations that validate the superiority of our method over the conventional approach and demonstrate its robustness, respectively. Finally, Section 6 gives the conclusion and the outlook for the next steps of research.

2. Model Description

This section introduces the ICE model. The basic parameters of ICE can be found in Table 1 [7], while more detailed information on the modeling of effectors can be found in Chapter 3 of [3]. Due to space constraints, this information will not be repeated here.

Figure 1 displays the layout of ICE, which features a high-sweep, tailless flying wing with a leading-edge sweep of 65 deg and 25 deg chevron shaping on the trailing edge. ICE is equipped with 13 independent effectors, including two pairs of leading-edge flaps (LEF), a pair of spoiler slot deflectors (SSD), a pair of all-moving tips (AMT), a pair of elevon (ELE), a pair of ganged pitch flaps (PF), and multi-axis thrust vectoring (MTV). Since this paper is focused on the cruising stage, MTV will not be taken into account.

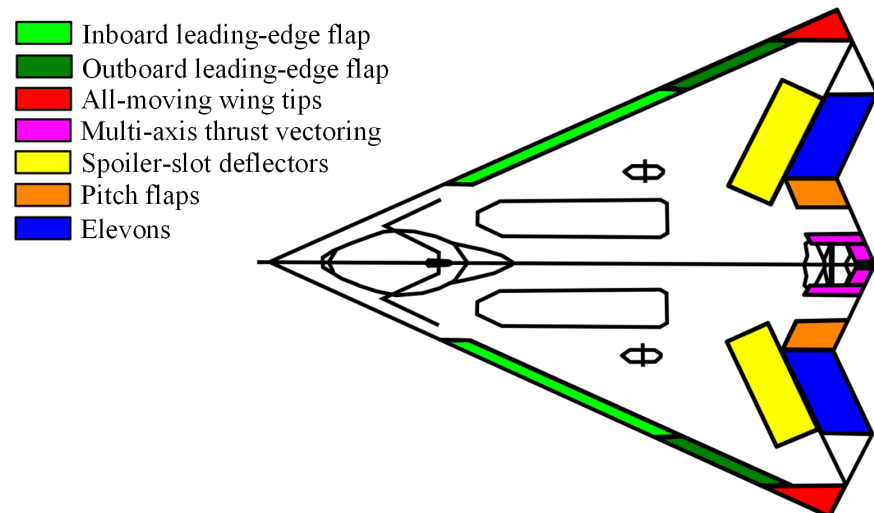
The deflection ranges of the effectors are, inboard LEF: 0–40 deg, outboard LEF: ± 40 deg, ELE: ± 30 deg, PF: ± 30 deg, AMT: ± 60 deg, SSD: 0–60 deg. The rate limits on the leading-edge devices are 40 deg/s and on all the other surfaces 150 deg/s.

The modeling of the UAV is based on the following two assumptions: 1st, the UAV flies in the atmosphere, and the atmosphere is incompressible; 2nd, the UAV's body is rigid. Please note that only the body of the UAV is considered a rigid body, but the effectors are deformable.

Table 1. The basic parameters of ICE.

Parameter	Nomenclature	Value	Unit
b	Lateral-directional reference length, span	37.50	ft
$\bar{c}$	Mean aerodynamic chord	28.75	ft
m	Weight	32,750	LBF
I_{yy}	Pitch Moment of Inertia	78,451	slug·ft ²
I_{xz}	Cross Product of Inertia	−525	slug·ft ²
I_{xx}	Roll Moment of Inertia	35,479	slug·ft ²
I_{zz}	Yaw Moment of Inertia	110,627	slug·ft ²
S	Reference area	808.60	ft ²
X_{th}	Moment arm for thrust vectoring	18.75	ft
X_{cg}	Gravity center	38.84% $\bar{c}$	
X_{ac}	Aerodynamic center	38.00% $\bar{c}$	

Remark 1. In previous studies [8,44,51], the MTV was used solely during the UAV's vigorous maneuvers or when other effectors were saturated. However, this paper proposes a cruise-oriented approach where ADP enables a superior trade-off between effector deflection and tracking error. This effectively eliminates the need for MTV, ensuring that effector saturation is avoided.

**Figure 1.** The layout of ICE.

The motion equation of 6-DOF UAV model is given below [14,18,52], the nomenclature of the variables in the following equation can be found in Table 2.

$$\begin{bmatrix} \dot{V} \\ \dot{\chi} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} \frac{1}{M} & 0 & 0 \\ 0 & \frac{1}{MVc_\gamma} & 0 \\ 0 & 0 & \frac{-1}{MV} \end{bmatrix} \cdot [T_{ve}G_f + F] \quad (1)$$

where V , γ , and χ are airspeed, flight path angle, and ground tracking angle, respectively. $s_\star$ and $c_\star$ represent $\sin \star$ and $\cos \star$. F donate the sum of aerodynamic force and thrust, which could be approximated through accelerometers, and $G_f = [0 \ 0 \ Mg]^T$ represents gravitational forces. Define μ , α , and β as the bank angle, angle of attack, and sideslip angle, then the dynamic of $[\mu \ \alpha \ \beta]$ is:

$$\begin{bmatrix} \dot{\mu} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} c_\alpha c_\beta & 0 & s_\alpha \\ s_\beta & 1 & 0 \\ s_\alpha c_\beta & 0 & -c_\alpha \end{bmatrix}^{-1} \left(-T_{bv}^T \begin{bmatrix} -\dot{\chi} s_\gamma \\ \dot{\gamma} \\ \dot{\chi} c_\gamma \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) \quad (2)$$

where p, q , and r are the body-axis roll, pitch, and yaw rates. The expression T_{vb} is the transformation matrix from the body frame to the velocity frame, and T_{ve} is the transformation matrix from the earth frame to the velocity frame. These matrices are given in [53]:

$$T_{vb} = \begin{bmatrix} c_\alpha c_\beta & s_\beta & s_\alpha c_\beta \\ -c_\alpha s_\beta c_\mu + s_\alpha s_\mu & c_\beta c_\mu & -s_\alpha s_\beta c_\mu - c_\alpha s_\mu \\ -c_\alpha s_\beta s_\mu - s_\alpha c_\mu & c_\beta s_\mu & -s_\alpha s_\beta s_\mu + c_\alpha c_\mu \end{bmatrix} \quad (3)$$

$$T_{ve} = \begin{bmatrix} c_\chi c_\gamma & s_\chi c_\gamma & s_\gamma \\ -s_\chi & c_\chi & 0 \\ c_\chi s_\gamma & s_\chi s_\gamma & c_\gamma \end{bmatrix} \quad (4)$$

The dynamic of $[p \ q \ r]$ is:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = J^{-1} \left(M_a - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times J \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) \quad (5)$$

where J is rotary inertia, defined as:

$$J = \begin{bmatrix} I_{xx} & 0 & I_{xz} \\ 0 & I_{yy} & 0 \\ I_{zx} & 0 & I_{zz} \end{bmatrix} \quad (6)$$

and $M_a = [l \ m \ n]$ is the aerodynamic moment, defined by:

$$M_a = \begin{bmatrix} l \\ m \\ n \end{bmatrix} = \bar{q} S \begin{bmatrix} b \cdot C_l \\ \bar{c} \cdot C_m \\ b \cdot C_n \end{bmatrix} = \bar{q} S \begin{bmatrix} b \cdot [(C_{l,base}(\alpha, \beta, V) + \sum_{i=1}^j C_{l,i}(\alpha, \beta, \delta))] \\ \bar{c} \cdot [(C_{m,base}(\alpha, \beta, V) + \sum_{i=1}^j C_{m,i}(\alpha, \beta, \delta))] \\ b \cdot [(C_{n,base}(\alpha, \beta, V) + \sum_{i=1}^j C_{n,i}(\alpha, \beta, \delta))] \end{bmatrix} \quad (7)$$

where the $\bar{q}$ is the dynamic pressure, b is span, $\bar{c}$ is the mean aerodynamic chord, $\delta \in \delta_{max}$ represents the deflection of effectors, δ_{max} is the deflection range of effectors [3], $C_{.,base}(\alpha, \beta, V)$ and $\sum_{i=1}^j C_{.,i}(\alpha, \beta, \delta)$ are the aerodynamic coefficients generated by the body and control surfaces.

Table 2. Nomenclature of variables.

Parameter	Nomenclature	Unit
F	= sum of aerodynamic force and thrust	LBS
l, m, n	= aerodynamic rolling, pitching, and yaw moment	LBS-ft
p, q, r	= body-axis roll, pitch, and yaw rate	rad/s
V	= airspeed	ft/s
γ	= flight path angle	rad
χ	= flight path sideslip angle	rad
α	= angle of attack	rad
β	= sideslip angle	rad
μ	= bank angle about the velocity vector	rad

The coordinate system involved in the above kinetic equations is shown in Figure 2. Equation (1) is defined in the tangent-plane coordinate system, which is aligned as a geographic system but has its origin fixed at a point of interest on the spheroid; Equation (2) is defined in the wind-axes system, and Equation (5) is defined in the body-fixed coordinate system. The relationship between the wind-axes system and the body-fixed coordinate system is shown in Figure 1. The origin of both is at the UAV's center of gravity, but the X-axis of the body-fixed coordinate system points in the direction of the nose, and the X-axis of the wind-axes system points in the direction of the relative wind [53].

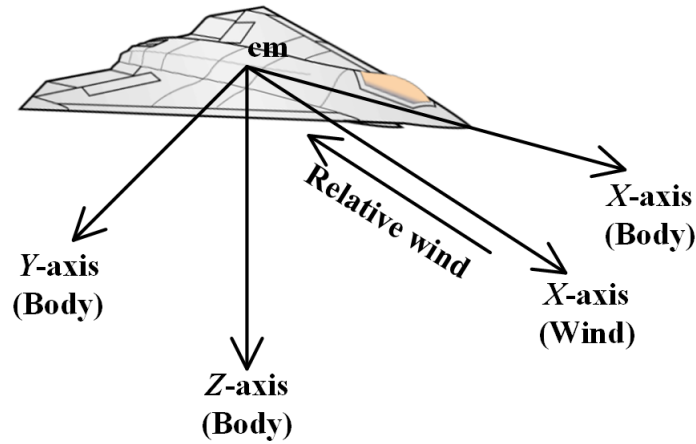


Figure 2. Explanation of wind-axes system and body-fixed coordinate system.

To facilitate the design of the attitude tracker, we first construct a control-oriented model that considers disturbances and model uncertainty caused by inaccurate aerodynamic parameters. Define $x_1 = [\mu \ \alpha \ \beta]$, $x_2 = [p \ q \ r]$, then

$$\begin{cases} \dot{x}_1 = f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 = f_2(x_2) + g_2(x_1, \delta) + d_t \end{cases} \quad (8)$$

where

$$f_1(x_1) = \begin{bmatrix} c_\alpha c_\beta & 0 & s_\alpha \\ s_\beta & 1 & 0 \\ s_\alpha c_\beta & 0 & -c_\alpha \end{bmatrix}^{-1} \cdot -T_{bv}^T \begin{bmatrix} -\dot{\chi} s_\gamma \\ \dot{\gamma} \\ \dot{\chi} c_\gamma \end{bmatrix} \quad (9)$$

$$g_1(x_1) = \begin{bmatrix} c_\alpha c_\beta & 0 & s_\alpha \\ s_\beta & 1 & 0 \\ s_\alpha c_\beta & 0 & -c_\alpha \end{bmatrix}^{-1} \quad (10)$$

$$f_2(x_2) = J^{-1}x_2 \times Jx_2 \quad (11)$$

$$g_2(x_1, \delta) = J^{-1}M_a - g_{2,e}(x_1, \delta) \quad (12)$$

$$d_t = g_{2,e}(x_1, \delta) + d \quad (13)$$

where d_t stand for total uncertainty, $g_{2,e}$ represents the control effectiveness that aerodynamic data fail to curve, and d represents the external disturbances.

3. Problem Formulation

The control structure is shown in Figure 3, donating $\cdot^c$ as the command signal. The control system consists of two parts, attitude control using NDI and angular rate control using ADP. The attitude control is to give the proper angular command so that the UAV can track the attitude command, assuming that the derivative of the attitude command is known, and gives the angular rate command as [44]:

$$x_2^c = g_1^{-1}(x_1) \cdot (\dot{x}_1^d - f_1(x_1)) \quad (14)$$

where $x_2^c = [p_c, q_c, r_c]^T$.

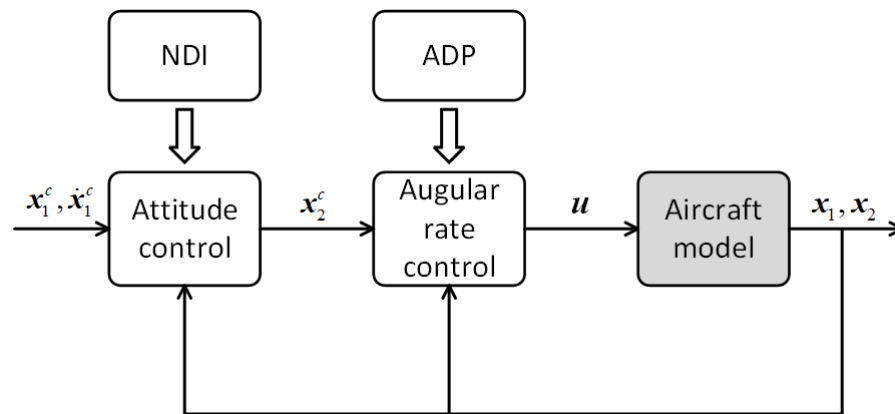


Figure 3. Control structure.

The goal of angular rate control is to give effectors' deflection so that the angular rate command can be tracked optimally according to the performance function. First, we constructed an augmented system:

$$\dot{x}_2 = f_2(x_2) + g_2(x_1, \delta) \quad (15)$$

$$\dot{x}_2^d = \lambda(x_2^c - x_2) \quad (16)$$

where λ is positively defined, and x_2^d represents the desired angular rate.

Define $X(t) = [x_2^T, x_2^{dT}]^T$, and rewrite the augmented system in the compact form:

$$\dot{X} = F(X) + G(X, \delta) + C + D \quad (17)$$

$$\text{where } F(X) = \begin{bmatrix} f_2(x_2) \\ -\lambda x_2 \end{bmatrix}, G(X) = \begin{bmatrix} g_2(x_1, \delta) \\ 0_{3 \times 1} \end{bmatrix}, C = \begin{bmatrix} 0_{3 \times 1} \\ \lambda x_2^c \end{bmatrix}, D = \begin{bmatrix} d_t \\ 0_{3 \times 1} \end{bmatrix}.$$

Remark 2. From Equation (14), it could be known that it is difficult to obtain the dynamic of x_2^c because of its dependence on the second order derivative of x_1^d . In many control methods, such as incremental control methods [14], $\dot{x}_1^c$ is readily available, but $\dot{x}_2^c$ is usually obtained through digital differentiation, which is not only sensitive to noise but also increases the computational complexity. Therefore, in ADP, the dynamic of x_2^d is constructed using NDI, and the use of $\dot{x}_1^d$ could be avoided in this way.

To facilitate the following analysis, the following assumptions are given:

Assumption 1. The total model uncertainty is bounded, there exist $D_{max} > 0$, let $\|D\| < D_{max}$.

Assumption 2. $\frac{\partial g_2(x_1, \delta)}{\partial \delta} \in \mathbb{R}_{3 \times 11}$ is line nonsingular, i.e., for $\|D\| < D_{max}$, there exist $|\Delta\delta| < \bar{\delta}$, $\bar{\delta} > 0$ such that $\frac{\partial g_2(x_1, \delta)}{\partial \delta} \Delta\delta = D$.

Remark 3. From Equation (8), it is obvious that the total model uncertainties can be equated to time-varying aerodynamic moment disturbances. To put it bluntly, Assumption 2 means that such disturbances are contained in the attainable moments set of ICE, and considering the redundant effectors that ICE is equipped with, its attainable moments set [54] has already been greatly expanded. From Equation (13), it also could be found that $g_2(x_1, \delta)$ could only reflect the aerodynamic characteristic of ICE to a certain degree. Except for the accuracy loss in the wind-tunnel test, the raw aerodynamic data also must be well-tailored to make it more suitable for flight control design, and the aerodynamic that $g_2(x_1, \delta)$ failed to reflect are seen as model uncertainties. Therefore, it is reasonable to say that $g_2(x_1, \delta)$ has the properties stated in Assumption 2.

Specifically, the angular rate control is to ensure the x_2 tracks the x_2^d by minimizing the following performance function. Consider that when the command signal does not converge to 0, the control input also tends not to be 0. To ensure the boundness of the performance function, the following discounted performance function is introduced:

$$J(X, \delta) = \int_t^\infty e^{-v(\tau-t)} (\bar{\delta}^T R \bar{\delta} + e_2^T Q e_2 + \delta^T R \delta) d\tau \quad (18)$$

where $e_2 = x_2 - x_2^d$, Q and R are positively defined matrices, $v > 0$ is the discounted factor. Compared to the conventional performance function, the upper bound of model uncertainties $\bar{\delta}$ is introduced into the performance function.

Remark 4. As a tried-and-tested control algorithm, NDI has been used in flight control for decades. In NDI, by feeding back the error between x_2 and x_2^d , the tracking of x_2^d could be achieved. Therefore, in the above performance function design, we also use the feedback of x_2^d instead of x_2^e .

Since the exact value of model uncertainties is unavailable, only the optimal tracker for the following nominal system that excludes the model uncertainties can be obtained:

$$\dot{X} = F(X) + G(X, \delta) + C \quad (19)$$

However, the optimal tracker designed for the nominal system using the performance function in Equation (18) is capable of handling model uncertainties. This point will be elaborated on later, and the optimal tracker for the nominal system is derived as follows:

Define $Q_T = [I_{3 \times 3}, -I_{3 \times 3}]^T Q [I_{3 \times 3}, -I_{3 \times 3}]$, then the discounted performance function could be modified as:

$$J(X, \delta) = \int_t^\infty e^{-v(\tau-t)} (\bar{\delta}^T R \bar{\delta} + X^T Q_T X + \delta^T R \delta) d\tau \quad (20)$$

Then define the optimal value function:

$$V(X) = \min J(X, \delta) \quad (21)$$

Remark 5. It is worth noting that the constraints imposed by the effector are not taken into account when solving the optimal control problem. However, it is possible to effectively prevent effector saturation by adjusting the weights in the performance function. This approach is widely adopted in the solution of optimization problems.

Differentiating Equation (20) and noting Equation (21) give the following Hamilton–Jacobi–Bellman (HJB) equation:

$$H(V, \delta) \triangleq \bar{\delta}^T R \bar{\delta} + X^T Q_T X + \delta^T R \delta - vV + V_X^T (F(X) + G(X, \delta) + C) = 0 \quad (22)$$

where $V_X = \frac{\partial V}{\partial X}$. Applying stationarity condition $\partial H(V, \delta) / \partial \delta = 0$, then we have the optimal tracker:

$$\delta^* = -\frac{1}{2} R^{-1} G_\delta^T V_X \quad (23)$$

where $G_\delta = \frac{\partial G(X, \delta)}{\partial \delta}$. If the HJB equation is solved, the optimal control can be obtained. Therefore, the main issue of this paper is to solve the HJB equation using ADP.

Remark 6. From Equation (23), it can be found that the only difference between the proposed optimal control for multi-input nonaffine systems and traditional optimal control lies in the use of G_δ^T , which is obtained through digital differentiation of aerodynamic data. In the theoretical study of the optimal control of nonaffine systems [49,50,55–57], the nonaffine part of the system is usually treated as completely unknown. Therefore, complex model identification methods are

needed in these studies. However, for realistic control systems, even if the accurate analytical model is not available, the data-based model can be built by observing the input and output of the system. With the improvement of wind-tunnel tests for UAV modeling, the aerodynamic data we obtain is more accurate than ever. The aforementioned theoretical studies, however, have not made sufficient use of these data. This is undoubtedly a huge waste. In this paper, the wind-tunnel data are used to assist ADP. By introducing D_{max} into the performance function, the ADP shows robustness against uncertainties in wind-tunnel data. Compared with online model identification, many more computational resources could be saved in this way.

The following lemma shows that with the performance function Equation (20), the optimal tracker in Equation (23) shows robustness against model uncertainties.

Lemma 1. Assume that the optimal control δ^* of the nominal system (19) with performance function (20) exist, δ^* could also make the system with model uncertainties (17) asymptotically stable.

Proof. To facilitate the proof, an auxiliary system is proposed:

$$\begin{cases} \dot{X} = F(X) + G(X, \delta) + C + D \\ \dot{a} = -\frac{1}{2}va \end{cases} \quad (24)$$

It is obvious that the subsystem $\dot{a} = -\frac{1}{2}va$ is asymptotically stable. Therefore, as long as the auxiliary system is proven to be asymptotically stable, Lemma 1 holds. Consider the optimal value function in Equation (21), for all $X \neq 0$, $V > 0$ and $V = 0$ only when $X = 0$. Therefore, choose the Lyapunov function as $V_R = a^2V$, and the time derivation of V_R is:

$$\begin{aligned} \dot{V}_R &= a^2\dot{V} - a^2vV \\ &= a^2[V_X^T(F(X) + G(X, \delta) + C) + V_X^TD - vV] \end{aligned}$$

according to the HJB equation and stationarity condition, we have:

$$\dot{V}_R = a^2[-\bar{\delta}^TR\bar{\delta} - X^TQ_TX - \delta^TR\delta + vV - 2\delta^TR^{\frac{1}{2}} \cdot R^{\frac{1}{2}}G_\delta^+D - vV]$$

where the G_δ^+ is the Moore-Penrose inverse of G_δ . According to Assumption 2 $G_\delta^+D = \Delta\delta$, such that:

$$\begin{aligned} \dot{V}_R &= a^2[-\bar{\delta}^TR\bar{\delta} + \Delta\delta^TR\Delta\delta - X^TQ_TX - \delta^TR\delta - 2\delta^TR^{\frac{1}{2}} \cdot R^{\frac{1}{2}}\Delta\delta - \Delta\delta^TR\Delta\delta] \\ &= a^2[-\bar{\delta}^TR\bar{\delta} + \Delta\delta^TR\Delta\delta - X^TQ_TX - (R^{\frac{1}{2}}\delta + R^{\frac{1}{2}}\Delta\delta)^T(R^{\frac{1}{2}}\delta + R^{\frac{1}{2}}\Delta\delta)] \end{aligned} \quad (25)$$

according to Assumption 2, $|\Delta\delta| < \bar{\delta}$ and R is positively definite, $-\bar{\delta}^TR\bar{\delta} + \Delta\delta^TR\Delta\delta < 0$. Therefore, it is clear that $\dot{V}_R$ is negative definite, the auxiliary system is asymptotically stable, and Lemma 1 is proven.

4. Main Result

As mentioned above, the main issue of this paper is solving the HJB equation and obtaining the optimal value function. The following single-layer neural network(NN) is applied to approximate the optimal value function:

$$V(X) = W^T\Psi(X) + \varsigma \quad (26)$$

where $\Psi(X) \in \mathbb{R}^l$ is the activation function vector, l is the number of neurons, ς is the approximation error, and the derivative to state of $V(X)$ is:

$$V_X = \nabla\Psi W + \nabla\varsigma \quad (27)$$

where $W \in \mathbb{R}^l$ is unknown ideal weights, $\nabla\Psi = \frac{\partial\Psi^T}{\partial X}$, $\nabla\varsigma = \frac{\partial\varsigma}{\partial X}$.

Suppose that the optimal value function is continuous and is defined on a bounded closed interval, according to the Weierstrass approximation theorem [58]. As l increases, the optimal value function can be uniformly approximated by the NN with arbitrarily high preciseness, which means that ς and $\nabla\varsigma$ can be arbitrarily small. In practice, we use a critical NN to approximate the optimal value function:

$$\hat{V}(\mathbf{X}) = \hat{\mathbf{W}}^T \boldsymbol{\Psi}(\mathbf{X}) \quad (28)$$

where $\hat{\mathbf{W}}$ is the estimation of unknown weights $\mathbf{W}$. Then the near-optimal control $\hat{\delta}$ can be obtained:

$$\hat{\delta} = -\frac{1}{2} \mathbf{R}^{-1} \mathbf{G}_{\delta}^T \nabla \boldsymbol{\Psi} \hat{\mathbf{W}} \quad (29)$$

In this paper, ADP updates the critical NN online using sampling data. As more sampling data ADP obtains, $\hat{\mathbf{W}}$ would approach $\mathbf{W}$ gradually. The rest of this section contains two parts. The first part introduces the update law of critical NN. The second part is stability analysis, in which we will discuss why the $\hat{\mathbf{W}}$ could approach $\mathbf{W}$ and why the system stays stable.

4.1. Update Law for Critical NN

The goal of the update scheme is to minimize $\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}}$ the estimation error of unknown NN weights. To derive the update law, substitute Equations (26) and (27) into HJB Equation (22), we have:

$$\begin{aligned} 0 = H(\mathbf{V}, \delta) &= \bar{\delta}^T \mathbf{R} \bar{\delta} + \mathbf{X}^T \mathbf{Q}_T \mathbf{X} + \delta^T \mathbf{R} \delta - v(\mathbf{W}^T \boldsymbol{\Psi}(\mathbf{X}) + \varsigma) \\ &\quad + (\mathbf{W}^T \nabla \boldsymbol{\Psi}^T + \nabla \varsigma^T)(\mathbf{F}(\mathbf{X}) + \mathbf{G}(\mathbf{X}, \delta) + \mathbf{C}) \\ &= \Lambda + \hat{\mathbf{W}}^T \mathbf{Y} + \tilde{\mathbf{W}}^T \mathbf{Y} + \varsigma_H \end{aligned} \quad (30)$$

where

$$\Lambda = \bar{\delta}^T \mathbf{R} \bar{\delta} + \mathbf{X}^T \mathbf{Q}_T \mathbf{X} + \delta^T \mathbf{R} \delta \quad (31)$$

$$\mathbf{Y} = -v \boldsymbol{\Psi}(\mathbf{X}) + \nabla \boldsymbol{\Psi}^T (\mathbf{F}(\mathbf{X}) + \mathbf{G}(\mathbf{X}, \delta) + \mathbf{C}) \quad (32)$$

$$\varsigma_H = -v \varsigma + \nabla \varsigma^T (\mathbf{F}(\mathbf{X}) + \mathbf{G}(\mathbf{X}, \delta) + \mathbf{C}) \quad (33)$$

From Equation (30), both the estimation error $\tilde{\mathbf{W}}$ and the weights of critical NN $\hat{\mathbf{W}}$ appear linearly. As discussed in Section 2, ς_H is bounded and tends to be zero as l increases. Moreover, the rest of variables $\mathbf{Y}$, Λ are accessible. This allows us to design the update law that minimizes the estimation error. Therefore, we define the following filter:

$$\begin{cases} \dot{\mathbf{P}} = -k\mathbf{P} + \mathbf{Y}\mathbf{Y}^T, \mathbf{P}(0) = 0 \\ \dot{\mathbf{Q}} = -k\mathbf{Q} + \mathbf{Y}\Lambda, \mathbf{Q}(0) = 0 \end{cases} \quad (34)$$

where $k \in \mathbb{R}^+$. The solution of the filter is:

$$\begin{cases} \mathbf{P} = \int_0^t e^{-k(t-\tau)} \mathbf{Y}\mathbf{Y}^T d\tau \\ \mathbf{Q} = \int_0^t e^{-k(t-\tau)} \mathbf{Y}\Lambda d\tau \end{cases} \quad (35)$$

According to Equations (30) and (35), we have:

$$0 = \mathbf{Q} + \mathbf{P}\hat{\mathbf{W}} + \mathbf{P}\tilde{\mathbf{W}} + \mu \quad (36)$$

where $\mu = \int_0^t e^{-k(t-\tau)} Y_{\zeta_H} d\tau$. According to the above analysis, with sufficient neurons, $\zeta \rightarrow 0$ and $\nabla \zeta \rightarrow 0$. Therefore, it is reasonable to find that μ is bounded, i.e., $|\mu| < \bar{\mu}$, where $\bar{\mu} \in \mathbb{R}^+$.

Define an auxiliary vector $\mathbf{M} \in \mathbb{R}^l$:

$$\mathbf{M} = \mathbf{Q} + \mathbf{P}\hat{\mathbf{W}} \quad (37)$$

According to Equation (36), $\mathbf{M}$ could be seen as the estimation error of NN weights:

$$\mathbf{M} = -\mathbf{P}\tilde{\mathbf{W}} - \mu \quad (38)$$

Hence, we could obtain the online update law of $\hat{\mathbf{W}}$ as:

$$\dot{\hat{\mathbf{W}}} = -\mathbf{K}\mathbf{M} \quad (39)$$

where $\mathbf{K}$ is constant positively defined matrix.

The update law in this paper is fundamentally different from some of the current methods [41,42] that employ the gradient-based algorithm to minimize the bellman error and only could ensure ultimately uniform boundlessness (UUB). The update law used here intends to use measurable state values and the weights of critical NN to represent the unknown estimation error of NN weights. In this way, the estimation error of NN weights can be ensured to be asymptotically convergent, and good convergence makes this update rate more suitable for online control systems. In what follows, we will show that $\hat{\mathbf{W}}$ could converge to the domain of $\mathbf{W}$.

Remark 7. The idea of this paper is to use the optimal tracker of nominal systems with modified performance functions to handle the systems with disturbances and model uncertainties. This idea can be well applied in linear systems [29] since the optimal control for linear nominal systems is easy to obtain by solving the Riccati equation. For the nonlinear system, however, it needs to solve the HJB equation to obtain the optimal control. For algorithms such as ADP that solve the HJB equation online, it gives the approximate solution of the HJB equation based on the sampled data. Still, for the systems that suffer from disturbance, it is impossible to measure the state value of the nominal system. According to Equations (31), (32) and (34), the information that update law uses include the quadratic function of effector deflection $\delta^T \mathbf{R} \delta$, the nominal system dynamic $\mathbf{F}(\mathbf{X}) + \mathbf{G}(\mathbf{X}, \delta) + \mathbf{C}$, and the quadratic function of state value $\mathbf{X}^T \mathbf{Q}_T \mathbf{X}$. The $\delta^T \mathbf{R} \delta$ and $\mathbf{F}(\mathbf{X}) + \mathbf{G}(\mathbf{X}, \delta) + \mathbf{C}$ are directly accessible. Only the $\mathbf{X}^T \mathbf{Q}_T \mathbf{X}$ is affected by disturbances and may influence the update law.

To address this problem, the filter system in Equation (34) is used here. Instead of using these values directly, the values used by the update law are processed by the filter system. In this way, the influence of disturbance on $\mathbf{X}^T \mathbf{Q}_T \mathbf{X}$ could be mitigated to a certain extent and make the update law more applicable. To further illustrate this opinion, the following example is introduced:

For the system with disturbance:

$$\begin{cases} \dot{x}_1 = -x_1 + x_2 + d \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = -x_2 \end{cases} \quad (40)$$

where d is the disturbance.

Figure 4 shows the unfiltered value of x_1^2 , and the value of x_1^2 filtered by $\frac{1}{0.8s+1}$. The real system is affected by disturbance, while the nominal system is not. From Figure 4a, there is a distinct difference between the unfiltered x_1^2 of the real system and the nominal system. But by filtering the two signals, the difference between the two becomes significantly smaller, as shown in Figure 4b. By choosing the parameters of the filter wisely, the filtered x_1^2 of the real system could approximate the x_1^2 of the nominal system pretty well.

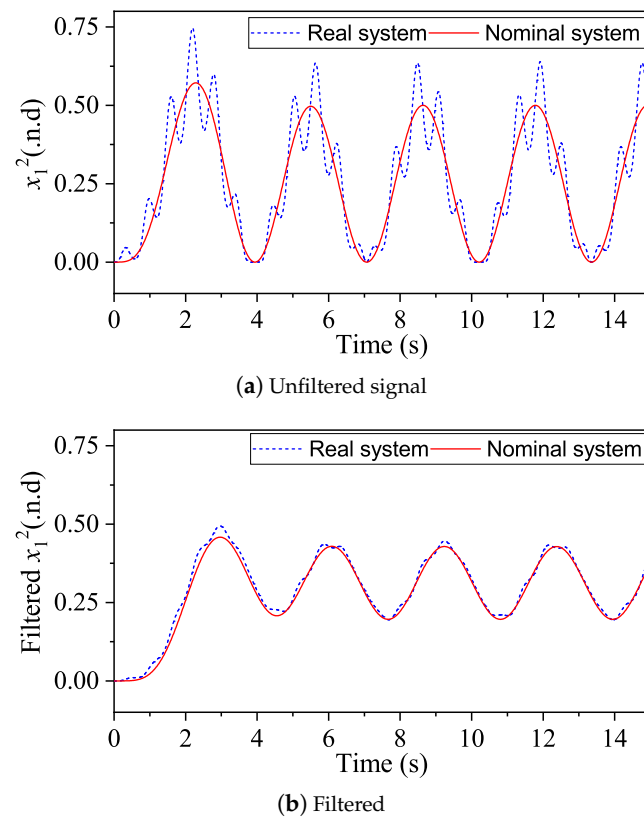


Figure 4. Effectiveness of the filter.

It should be noted that since the filter used in this paper is only the simplest first-order low-pass filter, this method could achieve quite a good result when the frequency of disturbances is higher than the frequency of state. Considering that high-frequency disturbance is a common kind of disturbance in control systems, the applicability of this method is acceptable. To cope with more complex situations, more targeted filters can be used according to the knowledge of the disturbance.

4.2. Stability Analysis

To prove that our approach can solve the optimal control online while making the tracking error converge, the following stability analysis is performed. Accordingly, modify the nominal system (24) in the following closed-loop form:

$$\begin{cases} \dot{X} = F(X) + G(X, \delta^*) + C + G(X, \hat{\delta}) - G(X, \delta^*) \\ \dot{a} = -\frac{1}{2}va \end{cases} \quad (41)$$

Assumption 3. There exist constants $\bar{f}, \bar{g} \in \mathbb{R}^+$ such that $\|F(X)\| \leq \bar{f}\|X\|$, $\|G(X, \delta)\| \leq \bar{g}\|X\|$.

Assumption 4. The $G(X, \delta)$ is Lipschitz continuous with respect to δ , i.e., $\|G(X, \delta_1) - G(X, \delta_2)\| \leq L\|\delta_1 - \delta_2\|$, where $L \in \mathbb{R}^+$.

Assumption 5. The command signal is bounded, i.e., there exists a constant $\bar{c} \in \mathbb{R}^+$ such that $\|C\| \leq \bar{c}$.

Assumption 6. The activation function vector $\Psi(X)$ and its derivative $\nabla\Psi$ is bounded.

Lemma 2. For the PE condition is satisfied for the regressor Y of NN, the optimal control and the update law could stabilize the tracking error of the nominal system (19), and the near-optimal

control converges to a bounded neighborhood around optimal control, i.e., $\mathbf{X}^T \mathbf{Q}_T \mathbf{X} \leq \zeta_X$ and $\|\hat{\delta} - \delta^*\| \leq \zeta_\delta$, where $\zeta_X, \zeta_\delta \in \mathbb{R}^+$.

Proof. The Lyapunov function is constructed as:

$$J = J_1 + J_2 \quad (42)$$

where $J_1 = \frac{1}{2} \tilde{\mathbf{W}}^T \mathbf{K}^{-1} \tilde{\mathbf{W}}$, $J_2 = \zeta_1 \mathbf{X}^T \mathbf{Q}_T \mathbf{X} + \zeta_2 a^2 V$.

According to the Lemma 1 in [37], if the PE condition is satisfied for the regressor $\mathbf{Y}$ of NN, then $\mathbf{P}$ is positively defined, and since the inequality $2ab \leq \eta a^2 + \frac{b^2}{\eta}$, the derivative of J_1 is:

$$\begin{aligned} \dot{J}_1 &= -\tilde{\mathbf{W}}^T \mathbf{K} \tilde{\mathbf{W}} + \tilde{\mathbf{W}}^T \boldsymbol{\mu} \\ &\leq -(\lambda_{\min}(\mathbf{P}) - \eta_1) \|\tilde{\mathbf{W}}\|^2 + \frac{1}{\eta_1} \|\boldsymbol{\mu}\|^2 \\ &\leq -(\lambda_{\min}(\mathbf{P}) - \eta_1) \|\tilde{\mathbf{W}}\|^2 + \frac{1}{\eta_1} \bar{\mu}^2 \end{aligned} \quad (43)$$

where $\lambda_{\min}(\mathbf{P})$ represents minimum eigenvalue of $\mathbf{P}$, and $\eta_1 \in \mathbb{R}^+$. According to Equations (22) and (24), the derivative of J_2 is:

$$\begin{aligned} \dot{J}_2 &= 2\zeta_1 \mathbf{X}^T \mathbf{Q}_T \dot{\mathbf{X}} - va^2 V + a^2 \dot{V} \\ &= 2\zeta_1 \mathbf{X}^T \mathbf{Q}_T [\mathbf{F}(\mathbf{X}) + \mathbf{G}(\mathbf{X}, \delta^*) + \mathbf{C} + \mathbf{G}(\mathbf{X}, \hat{\delta}) - \mathbf{G}(\mathbf{X}, \delta^*)] - va^2 V \\ &\quad + a^2 (-\bar{\delta}^T \mathbf{R} \bar{\delta} - \mathbf{X}^T \mathbf{Q}_T \mathbf{X} - \delta^T \mathbf{R} \delta + vV) \\ &= 2\zeta_1 \mathbf{X}^T \mathbf{Q}_T \mathbf{F}(\mathbf{X}) + 2\zeta_1 \mathbf{X}^T \mathbf{Q}_T \mathbf{G}(\mathbf{X}, \delta^*) + 2\zeta_1 \mathbf{X}^T \mathbf{Q}_T \mathbf{C} + 2\zeta_1 \mathbf{X}^T \mathbf{Q}_T (\mathbf{G}(\mathbf{X}, \hat{\delta}) - \mathbf{G}(\mathbf{X}, \delta^*)) \\ &\quad - \zeta_2 a^2 \bar{\delta}^T \mathbf{R} \bar{\delta} - \zeta_2 a^2 \mathbf{X}^T \mathbf{Q}_T \mathbf{X} - \zeta_2 a^2 \delta^T \mathbf{R} \delta \\ &\leq \zeta_1 (\eta_2 \|\mathbf{X}^T \mathbf{Q}_T\|^2 + \frac{1}{\eta_2} \|\mathbf{F}(\mathbf{X})\|^2) + \zeta_1 (\eta_3 \|\mathbf{X}^T \mathbf{Q}_T\|^2 + \frac{1}{\eta_3} \|\mathbf{G}(\mathbf{X}, \delta^*)\|^2) \\ &\quad + \zeta_1 (\eta_4 \|\mathbf{X}^T \mathbf{Q}_T\|^2 + \frac{1}{\eta_4} \|\mathbf{C}\|^2) \\ &\quad + \zeta_1 (\eta_5 \|\mathbf{X}^T \mathbf{Q}_T\|^2 + \frac{1}{\eta_5} \|\mathbf{G}(\mathbf{X}, \delta^*) - \mathbf{G}(\mathbf{X}, \hat{\delta})\|^2) \\ &\quad - \zeta_2 a^2 \bar{\delta}^T \mathbf{R} \bar{\delta} - \zeta_2 a^2 \lambda_{\min}(\mathbf{Q}_T) \|\mathbf{X}\|^2 - \zeta_2 a^2 \lambda_{\min}(\mathbf{R}) \|\delta^*\|^2 \end{aligned} \quad (44)$$

According to Assumptions 3–5, we have:

$$\begin{aligned} \dot{J}_2 &\leq [(\zeta_1 \eta_2 + \zeta_1 \eta_3 + \zeta_1 \eta_4 + \zeta_1 \eta_5) \lambda_{\max}^2(\mathbf{Q}_T) + \frac{\zeta_1}{\eta_2} \bar{f}^2 + \frac{\zeta_1}{\eta_3} \bar{g}^2 - \zeta_2 a^2 \lambda_{\min}(\mathbf{Q}_T)] \|\mathbf{X}\|^2 \\ &\quad + \frac{\zeta_1}{\eta_5} L^2 b_w \|\tilde{\mathbf{W}}\|^2 + \frac{\zeta_1}{\eta_5} L^2 b_\varsigma \|\nabla \varsigma\|^2 - \zeta_2 a^2 \bar{\delta}^T \mathbf{R} \bar{\delta} - \zeta_2 a^2 \lambda_{\min}(\mathbf{R}) \|\delta^*\|^2 \end{aligned} \quad (45)$$

where $b_w = \|\frac{1}{2} \mathbf{R}^{-1} \mathbf{G}_\delta^T \nabla \Psi\|$, $b_\varsigma = \|\frac{1}{2} \mathbf{R}^{-1} \mathbf{G}_\delta^T\|$ according to Assumptions 4 and 6, b_w and b_ς are bounded.

Therefore, the derivative of J is:

$$\dot{J} = \dot{J}_1 + \dot{J}_2 \leq -\aleph_1 \|\mathbf{X}\| - \aleph_2 \|\tilde{\mathbf{W}}\| - \aleph_3 \|\delta^*\| + \aleph_4 \quad (46)$$

where

$$\begin{aligned} \aleph_1 &= -(\zeta_1 \eta_2 + \zeta_1 \eta_3 + \zeta_1 \eta_4 + \zeta_1 \eta_5) \lambda_{\max}^2(\mathbf{Q}_T) - \frac{\zeta_1}{\eta_2} \bar{f}^2 - \frac{\zeta_1}{\eta_3} \bar{g}^2 + \zeta_2 a^2 \lambda_{\min}(\mathbf{Q}_T) \\ \aleph_2 &= \lambda_{\min}(\mathbf{P}) - \eta_1 - \frac{\zeta_1}{\eta_5} L^2 b_w \\ \aleph_3 &= \zeta_2 a^2 \lambda_{\min}(\mathbf{R}) \\ \aleph_4 &= +\frac{1}{\eta_1} \bar{\mu}^2 + \frac{\zeta_1}{\eta_5} L^2 b_\varsigma \|\nabla \varsigma\|^2 - \zeta_2 a^2 \bar{\delta}^T \mathbf{R} \bar{\delta} \end{aligned} \quad (47)$$

By designing the parameters wisely, it could be ensured that $B_1, B_2, B_3 > 0$. B_4 is mainly influenced by critical NN's estimation error, which would converge to zero as the number of neurons increases.

According to Equation (47), if the following inequalities hold, $\dot{J}_2$ would be negative defined:

$$\begin{aligned} -\aleph_1 \|X\| + \aleph_4 &< 0 \\ -\aleph_2 \|\tilde{W}\| + \aleph_4 &< 0 \\ -\aleph_3 \|\delta^*\| + \aleph_4 &< 0 \end{aligned} \quad (48)$$

Therefore, according to Lyapunov theory, the closed-loop system is stable, and the weights error of NN $\tilde{W}$ is bounded, consider the estimation error of critical NN is also bounded, the near-optimal control will converge to a bounded neighborhood of optimal control. $\square$

5. Simulation Verification

This section presents two representative simulations to illustrate the effectiveness of the ADP-based integrated-control-and-control-allocation scheme. The simulations are conducted using fixed-step ode4(Runge–Kutta) solver. The fixed-step size is 0.01 s. The block diagram of the 6-DOF UAV simulation model can be found in Figure 10 of the report given by Niestroy [7]. Simulation 1 compares our control scheme with conventional incremental dynamic inversion and pseudo-inverse control allocation(INDIPI) to verify the optimality of our method. Simulation 2 aims to test the robustness of the proposed control scheme. It is assumed that the leading-edge actuators are represented by the transfer function $\frac{(18)(100)}{(s+18)(s+100)}$ while all the other actuators, including thrust vectoring, as $\frac{(40)(100)}{(s+40)(s+100)}$.

The initial condition of the UAV is $V(0) = 1240$ ft/s, $\chi(0) = 0$, $\gamma(0) = 0$, $\beta(0) = -0.0196$ deg, $\alpha(0) = 3.759$ deg, $\mu(0) = 0.304$ deg, $p(0) = 0$, $q(0) = 0$, $r(0) = 0$, and the initial height is $H(0) = 10,000$ ft. The angle of attack command α_c is generated by a band-limited white noise pass the second order filter $\frac{0.5}{5s^2+2s+0.5}$, the bank angle signal command $\mu_c = 6 \sin(0.5t)$, and the sideslip angle command $\beta_c = 0$.

The control parameters are set as $\lambda = 3 \cdot I_{3 \times 3}$, $R = I_{11 \times 11}$, $Q = 5 \cdot I_{3 \times 3}$, $K = 500 \cdot I_{7 \times 7}$, $v = 1$. Define $e_1 = p - p_c$, $e_2 = q - q_c$, $e_3 = r - r_c$, then the activation function vector are designed as $\Psi(X) = [e_1^2, e_2^2, e_3^2, e_1^2 e_2, e_1 e_2 e_3, e_3^2 e_2, e_2^2 e_3]^T$, with initial weights $\hat{W}(0) = [1000, 1000, 1500, 0, 0, 0, 0]^T$.

Remark 8. We designed the activation function vector as above because it would be easier to find the initial admissible control policy [59]. Clearly, there exists $\frac{\partial e_1^2}{\partial p} = 2e_1$. In this sense, the initial control is equivalent to proportional control. Compared with dealing with complex nonlinear feedback, finding an admissible proportional control law is much easier.

Model uncertainty exists in both simulations. As mentioned above, model uncertainty mainly comes from inaccurate aerodynamic data, which are used to obtain G_δ . Since the aerodynamic data provided by Niestroy is a series of discrete points, it takes interpolation so that these data are of practical use. In both simulations, different interpolation methods are used for controller design and model construction to simulate that controller cannot access accurate aerodynamic data. Specifically, the cubic spline is applied for model construction, and linear interpolation is used in controller design.

Remark 9. The cubic spline is applied for model construction because the actual aerodynamic data should be continuous and smooth. Meanwhile, using linear interpolation in control could save online computational load. As mentioned in [14], different interpolation methods can cause errors of up to 30%. Take the aerodynamic data of a set of all-moving wingtips as an example, as shown in

Figure 5, from which it can be founded that the slopes of the tangents of linear interpolation and cubic spline, i.e., $\tan \tau_2$ and $\tan \tau_1$, are different.

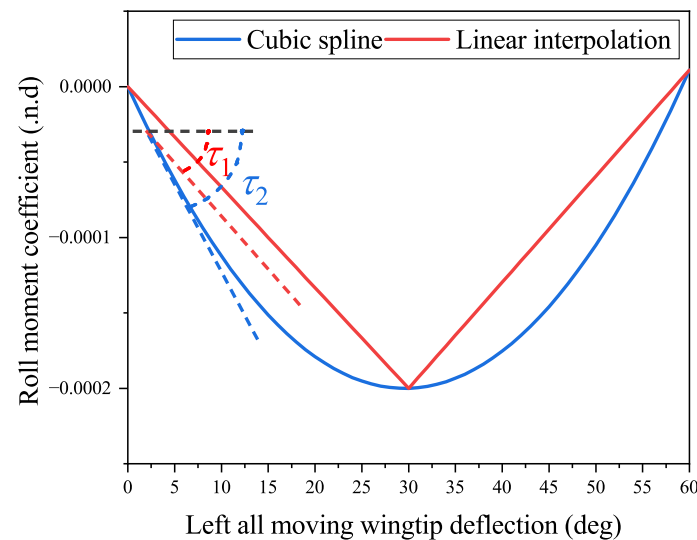


Figure 5. Moment coefficients vs. left all-moving tips deflection.

5.1. Simulation 1

In this simulation, the ADP-based control scheme is compared with INDIPI [60]. Specifically, our approach and INDIPI are to track the same attitude command, and the performance of both is judged according to the tracking performance, flight quality, and control input. Considering that inaccurate control effectiveness could cause INDIPI to lose stability and this simulation is not to compare the robustness of INDIPI and ADP-based method, INDIPI could obtain accurate model information in this simulation, and model uncertainties only influence our method. Set $\bar{\delta}$ as 11-dimensional vector $[5.6, 5.6, \dots, 5.6]^T$. Please note that the same NDI scheme is applied in attitude control of the ADP-based method and INDIPI.

The result is shown in Figures 6–17. First, from Figure 17, ADP-based method shows good convergence. Moreover, the proposed method outperforms INDIPI in three ways: superior flight quality, intelligence, and better effector deflection pattern.

From Figures 9–12, the flight quality under ADP control is better than that of INDIPI. From Figure 9, the p signal under ADP control is steadier. After the adjustment period before 5 s, the p signal under ADP control keeps steady, while the p signal fluctuates under INDIPI control, such as around 38 s and 44 s. From Figure 10, q chattered all the time under the control of INDIPI, such chattering also can be found in effector deflection as shown in Figure 16, and this can cause fatigue of the effector, which is very dangerous in reality, while the effector deflection under ADP control is more fluent. From Figure 11, it can be found that there is a sudden change in r signal under INDIPI control at 20 s, 29 s, 36 s, and 44 s. It also could be seen that the r signal under ADP control also appears to fluctuate, but, differently from the sudden change under INDIPI control exhibited all the time, it can be found that the fluctuation under ADP control is becoming lighter.

According to the above description, the control performance of our method is better compared to INDIPI. However, our method goes far beyond that. With the help of ADP, our method shows intelligence, i.e., it could improve its policy according to its experience.

The specific manifestations of the ADP-based method's intelligence are p signal under ADP control only fluctuates once around 5 s. After that, it is always very smooth, and, compared with the sudden change in r under INDIPI control exhibited all the time, the fluctuation of r signal under ADP control becomes lighter as the control system runs.

A more extreme example is introduced to further illustrate the intelligence of ADP, as shown in Figure 14, which shows the p signal under the proportional control that adopts the initial weights of critical NN. Comparing Figures 9 and 14 it can be found that no

matter whether under ADP control or proportional control, fluctuation occurred around 5 s. However, such fluctuation only occurs once under ADP control; for proportional control, such fluctuation occurs repeatedly and eventually leads to losing control. From Figure 17, it can be seen that the critical NN weights undergo a large adjustment at 5 s, after which there is no more fluctuation similar to at 5 s. ADP could learn from such fluctuation; therefore, the subsequent policy is more suitable for flight control with a broader flight envelope.

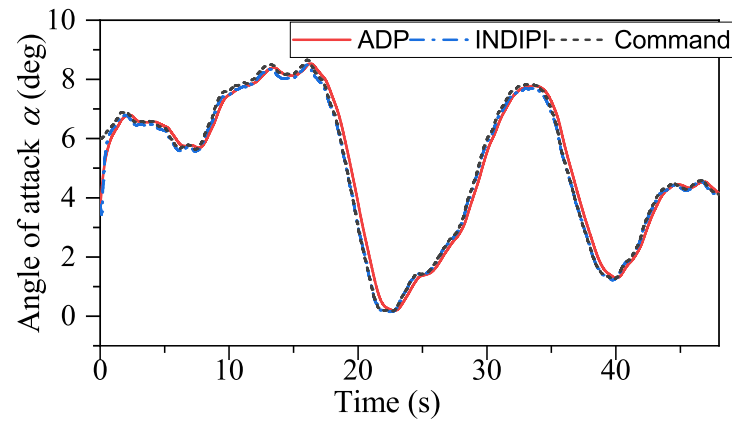


Figure 6. Angle of attack.

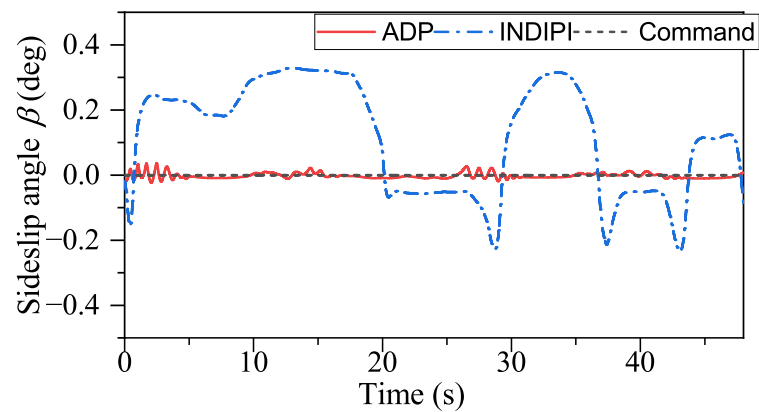


Figure 7. Sideslip angle.

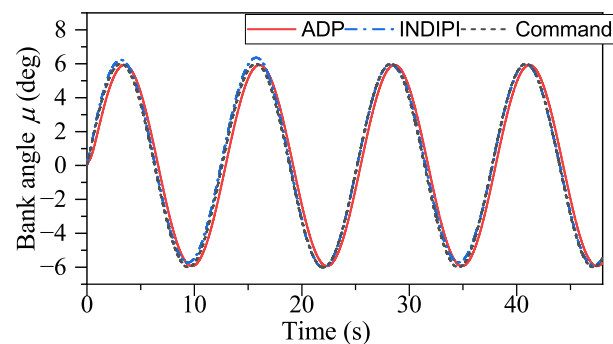


Figure 8. Bank angle.

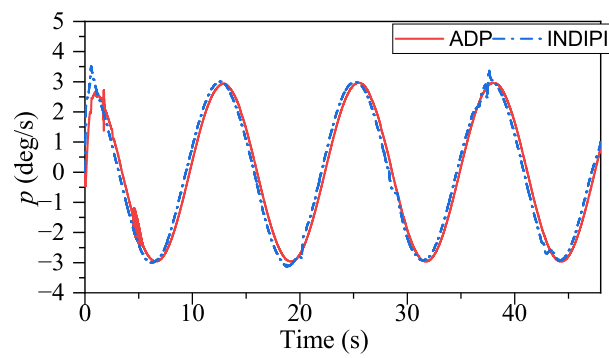


Figure 9. Body-axis roll rate.

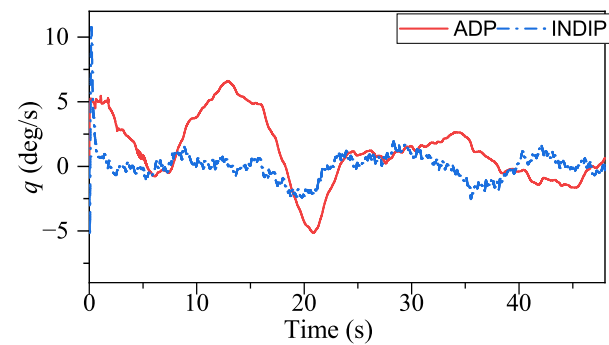


Figure 10. Body-axis pitch rate.

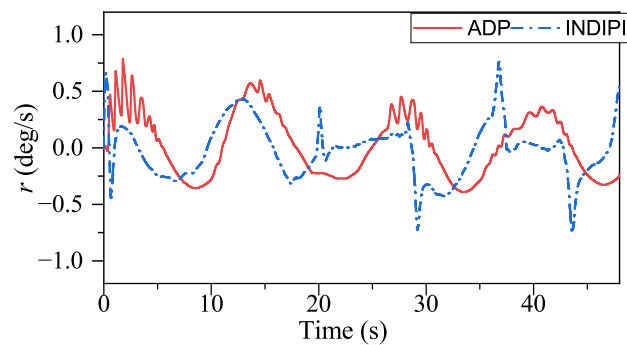


Figure 11. Body-axis yaw rate.

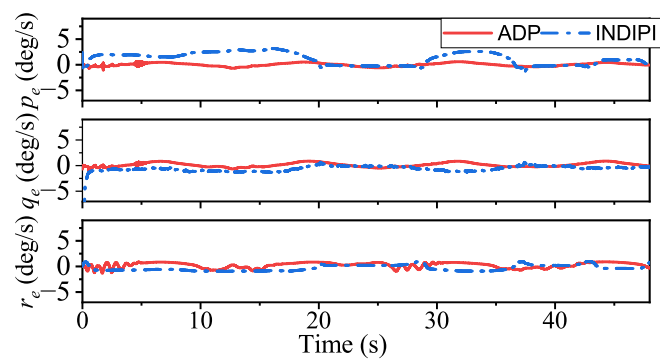


Figure 12. Tracking error of x_2^c .

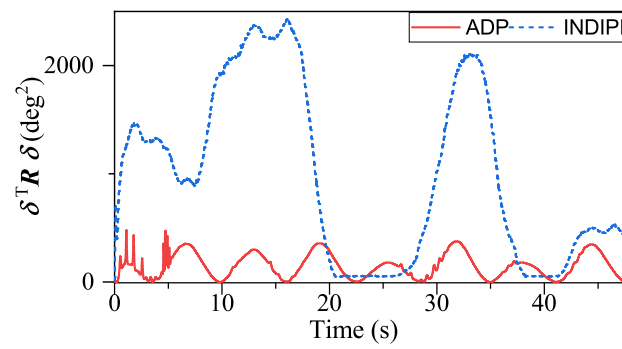


Figure 13. Weighted quadratic sum of effector deflection.

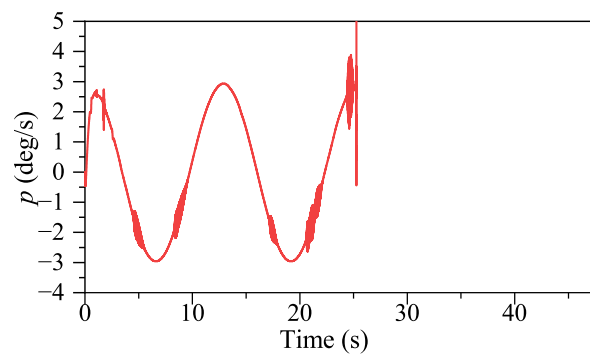


Figure 14. p under proportional control.

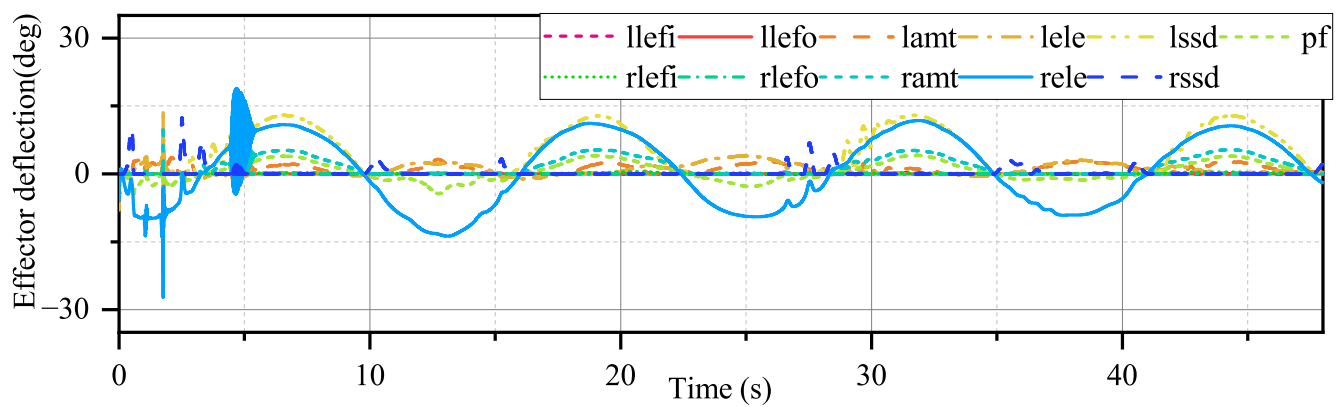


Figure 15. Effector deflection of ADP.

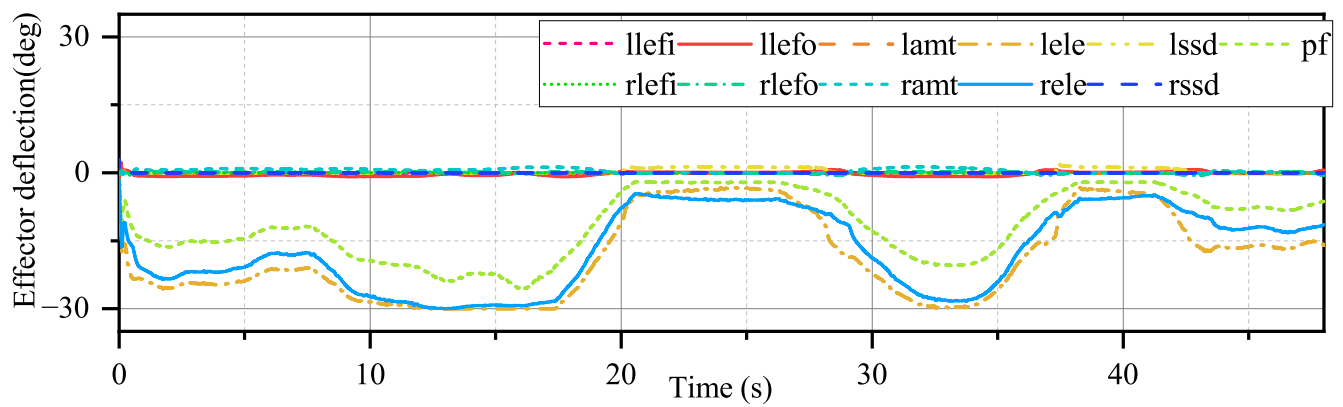


Figure 16. Effector deflection of INDIPI.

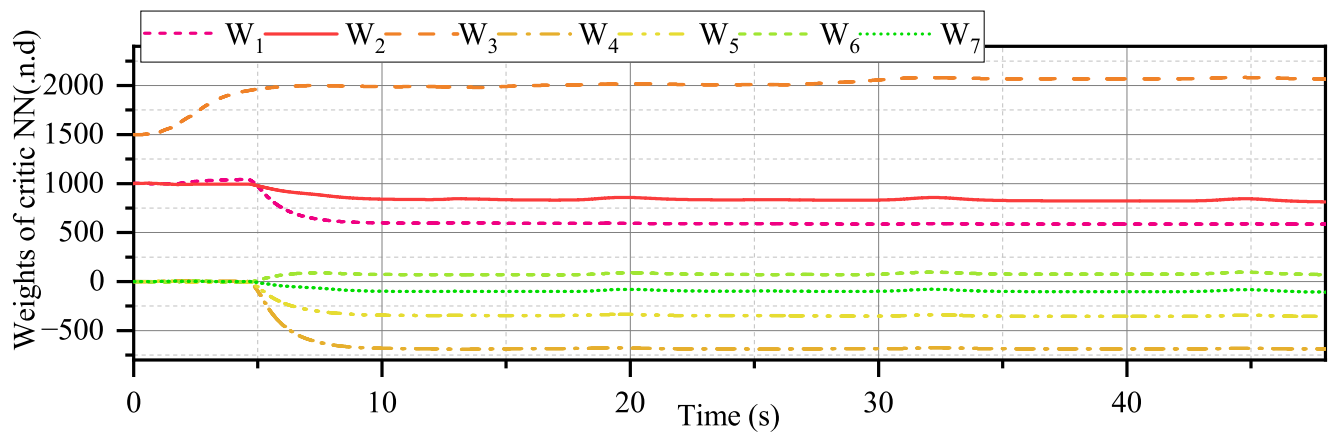


Figure 17. Convergence of the critical NN weights.

From Figures 15 and 16, the effector deflection generated by INDIPI and ADP shows different patterns. In Figure 16, it can be seen that only three effectors participate in the process under INDIPI control, even though some effectors appear to be saturated. However, for the ADP, under the modulation of the performance function, more effectors participate in the control process, and the effector deflection amplitude is significantly smaller than that of the INDIPI. From Figure 13, it also can be seen that the weighted quadratic sum of effector deflection given by ADP is much smaller than INDIPI.

Overall, integrated-design ADP's performance is better than conventional INDIPI's. Compared with INDIPI, ADP allows for a trade-off between tracking performance and effector deflection. The performance function dominates such a trade-off, so ADP would not waste too many resources to pursue tiny improvements in tracking performance. Coupling with its learning mechanism, ADP can achieve the same tracking performance as INDIPI in an optimal manner.

5.2. Simulation 2

Simulation 2 discusses the robustness of the proposed method and the effect of $\bar{\delta}$. The UAV suffers from aforementioned model uncertainties and external disturbances $d = [0.06 \sin(20t), 0.04 \cos(20t), 0.03 \sin(20t)]^T$. The UAV is to follow the same attitude command as Simulation 1 and set $\bar{\delta}$ as $[17.8, 17.8, \dots, 17.8]^T$, the result is shown in Figures 18–25.

With the help of ADP, the tracking performance of our method in the presence of external disturbances is unaffected, as shown in Figures 18–20. However, some small chattering can be observed in the angular rate signal, as depicted in Figures 21–23, which is typical for a UAV subject to external disturbance. Nevertheless, this does not compromise the stability of the closed-loop system. Figure 24 demonstrates that more effectors are involved in controlling external disturbances. Most importantly, the convergence of critical NN weights remains satisfactory, as demonstrated by Figure 25.

From the stability analysis, it can be found that our method's robustness comes from $\bar{\delta}$. In the following, the performance of different $\bar{\delta}$ is tested.

We begin by testing $\bar{\delta} = [5.6, 5.6, \dots, 5.6]^T$. Due to space limitations, we only present the convergence of the critical NN weights in Figure 26. As shown in Figure 26, the convergence of the critical NN in this result is initially similar to Simulation 1. However, the critical NN weights do not ultimately converge due to external disturbances.

From a theoretical perspective, Lemma 1 can explain the non-convergence of the algorithm, as the closed-loop system's stability can only be guaranteed when $\bar{\delta}$ is of sufficient magnitude.

From the other point of view, if the UAV experiences intense external disturbances, the initial sampled data may not provide enough information for ADP to update critical NN weights. This is particularly true when the UAV is required to track random commands, as

the old critical neural network weights may not be equipped to handle new, unforeseen scenarios. As a result, the NN weights may take longer to converge, making it difficult to maintain control of the UAV. In this sense, $\bar{\delta}^T R \bar{\delta}$ not only acts as a means of compensation for external disturbances that may initially affect performance but can also be seen as an estimation of the potential impact of such disturbances on performance function. This helps the ADP better understand the current situation, allowing the weights of the critical NN to converge more quickly to a stable value.

Unlike the affine systems, where the upper bound on the effect of external disturbances on the performance function is easily ascertained [29], for the nonaffine system, the design of $\bar{\delta}$ is more rely on the experience. Still, considering that $\bar{\delta}$ has the actual physical definition, it would not be too hard to find a proper $\bar{\delta}$.

The above result shows that $\bar{\delta}$ must exceed the upper limit of external disturbance effects so that ADP can show robustness.

However, $\bar{\delta}$ should not exceed a reasonable value either. To illustrate this point, we conducted a convergence test of the critical neural network with $\bar{\delta} = [56.4, 56.4, \dots, 56.4]^T$, which is very large. The result is depicted in Figure 27. It can be observed that the system experiences a significant shift in critical NN weights, leading to a collapse within 3 s. From the analysis of Equation (34), having a too large $\bar{\delta}$ can result in P not being positively definite. More bluntly, too large $\bar{\delta}$ could dominate the dynamics of critical neural network weights, causing the ignored of sampled data that aid in policy improvement.

In conclusion, Simulation 2 demonstrates that our method can effectively withstand model uncertainties and external disturbances, given the appropriate selection of $\bar{\delta}$.

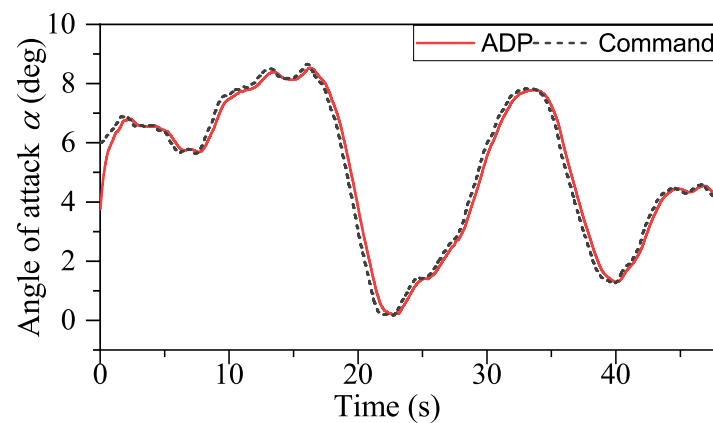


Figure 18. Angle of attack.

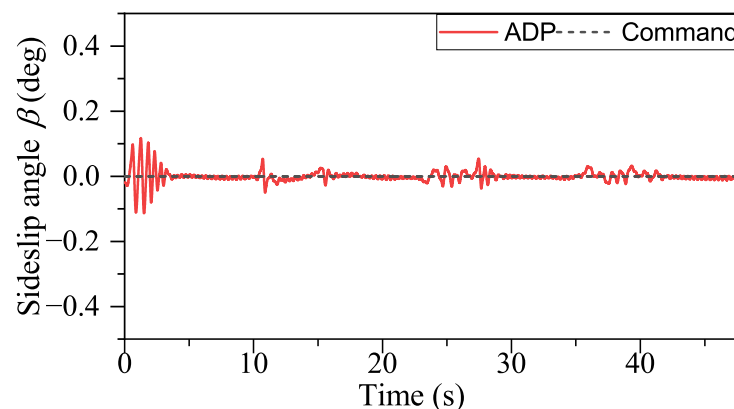


Figure 19. Sideslip angle.

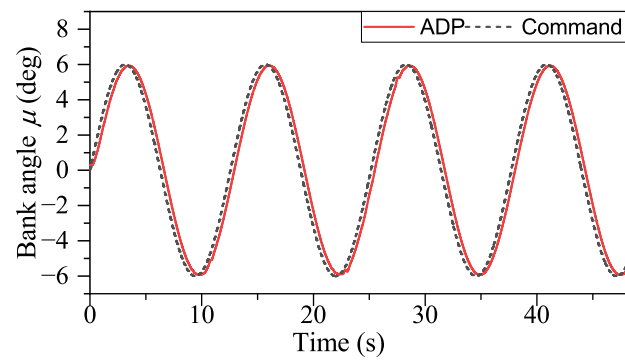


Figure 20. Bank angle.

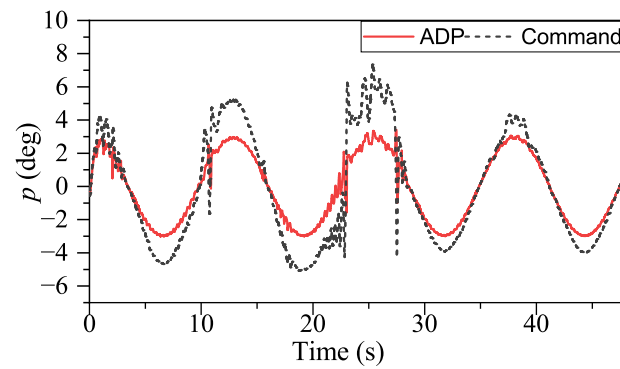


Figure 21. Body-axis roll rate.

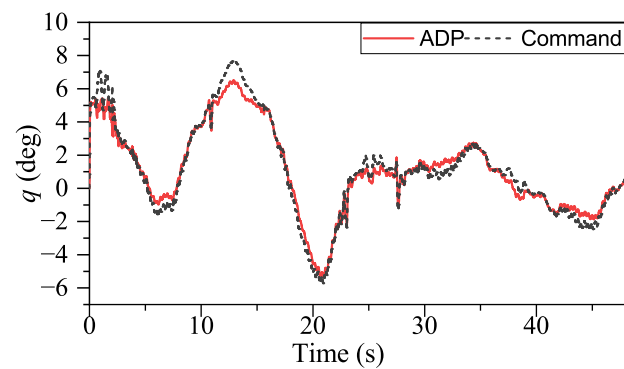


Figure 22. Body-axis pitch rate.

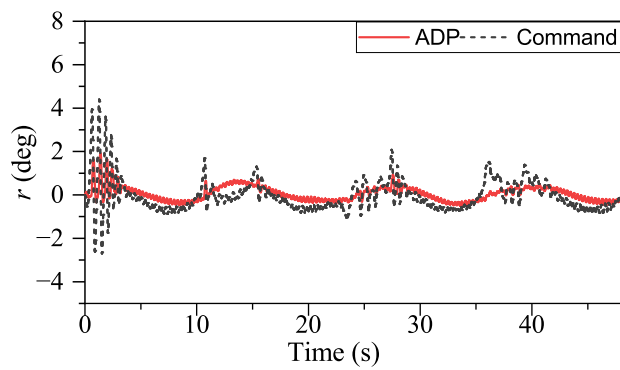


Figure 23. Body-axis yaw rate.

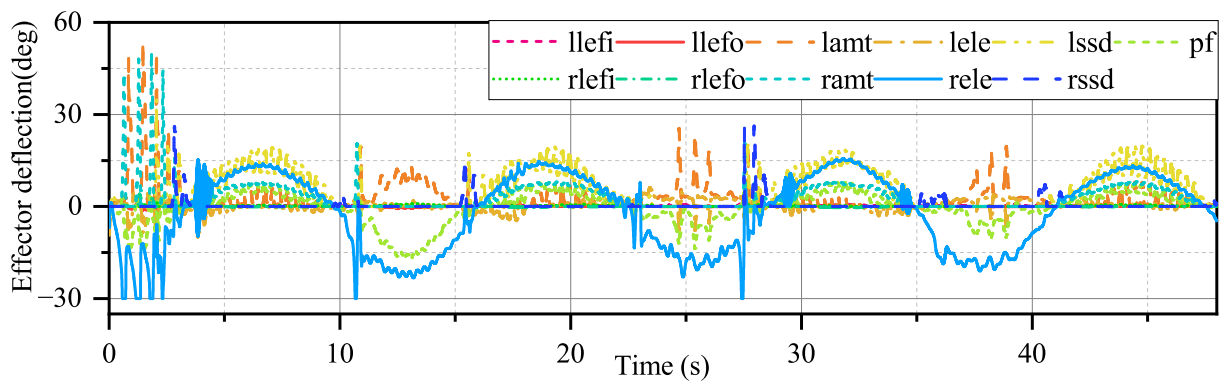
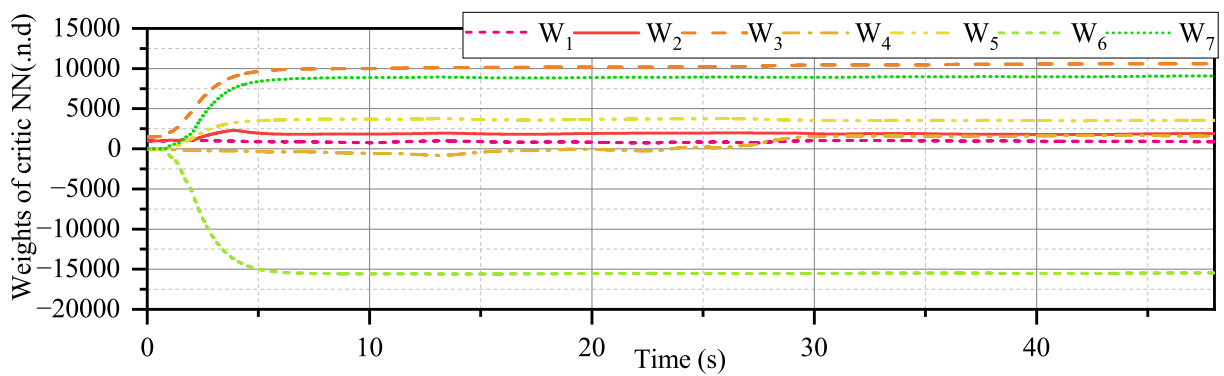
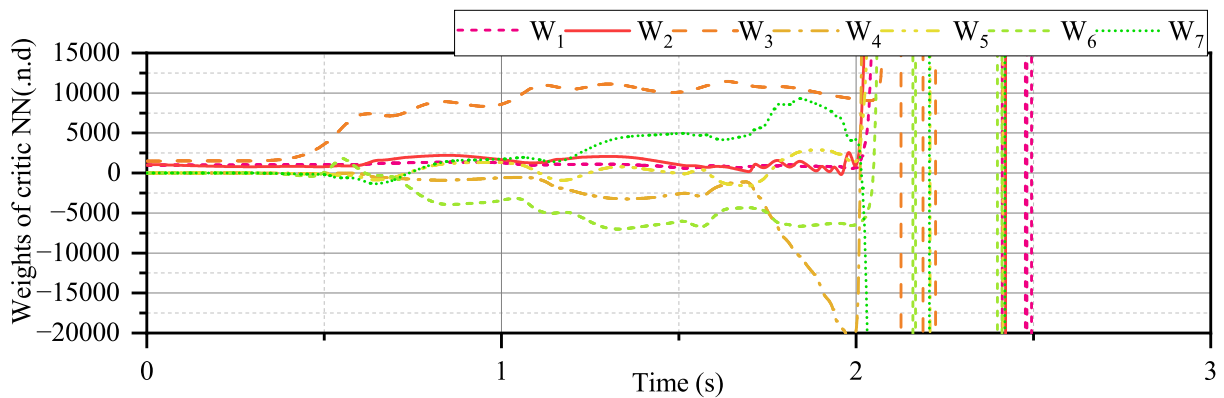
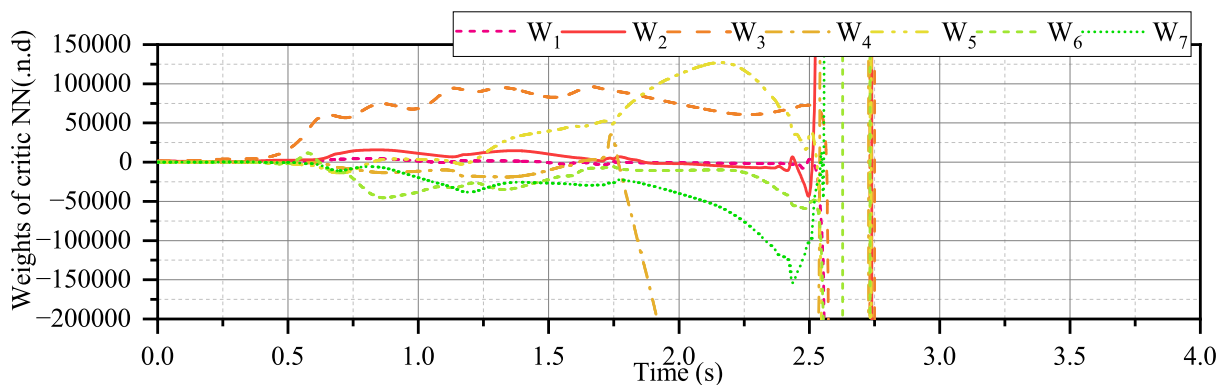


Figure 24. Effector deflection.

Figure 25. Convergence of the critical NN weights when $\bar{\delta} = [17.8, 17.8, \dots, 17.8]^T$.Figure 26. Convergence of the critical NN weights when $\bar{\delta} = [5.6, 5.6, \dots, 5.6]^T$.Figure 27. Convergence of the critical NN weights when $\bar{\delta} = [56.4, 56.4, \dots, 56.4]^T$.

6. Conclusions and Outlook

The proposed method uses ADP to integrate control and control allocation, resulting in superior performance compared to conventional methods. Without using any model identification techniques, the ADP-based method exhibits strong convergence and robustness in the face of external disturbance and model uncertainty. Additionally, it presents a novel approach to flight control for over-actuated UAVs with nonaffine control inputs. From a control-theory perspective, the paper presents a straightforward yet efficient optimal tracking method for nonaffine systems, with theoretical evidence verifying its robustness. Specifically, this study has two key advantages in comparison to existing research. First, our method achieves better performance than traditional control architectures that separate control and control allocation by using a more optimized approach. Second, unlike many current optimal controllers for nonaffine systems, our method remains robust and does not depend on any model identifiers.

The proposed method has certain limitations that require attention. First, this method is only aimed at the cruise stage, as the nonlinear characteristics of the aircraft during this phase are not as prominent, and the optimal value function is relatively simple and can be well-fitted by a polynomial network. However, if large maneuvering flight is required, a more complex network structure needs to be introduced. This inevitably requires an improvement in the weight update rate to ensure system stability. Second, selecting the initial value for the critical network can be challenging when a complex network is used since the convergence of this method relies on the proper choice of the network's initial value. Thirdly, the design of the $\bar{\delta}$ is still heavily reliant on empirical knowledge. As demonstrated in the simulation section, a $\bar{\delta}$ that is too small may weaken robustness, while too large $\bar{\delta}$ may harm the closed-loop stability. Lastly, there is a dearth of real-world validation of this method. The external perturbations applied in the simulation offer only a limited exhibition of robustness and stability since the external interferences experienced by a UAV, in reality, are much more complex.

The next-step studies should focus on the following aspects: First, more complex neural networks can be introduced to further approximate the value function and handle more complex situations. Second, it would be very worthy work to introduce some intelligent algorithms to help design the performance function. Thirdly, only linear filters, as shown in Equation (34), are used in this paper, making our method better when facing high-frequency disturbances. In future studies, more advanced filters could be introduced to improve the performance of the ADP-based method when facing various disturbances. More importantly, it would be very expected that the performance of our method can be validated in real flight experiments.

Author Contributions: Conceptualization, Z.H. and Y.W.; Methodology, J.H.; Software, Z.H.; Validation, Y.B. and J.C.; Resources, J.H.; Data curation, Y.W.; Writing—original draft preparation, Z.H.; Writing—review and editing, Y.B. and L.H.; Visualization, Z.H.; Supervision, J.H.; Project administration, Z.H.; Funding acquisition, Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China [grant number 62103439], China Postdoctoral Science Foundation [grant number 2020M683716], and the Natural Science Basic Research Program of Shaanxi Province [grant number 2021JQ-364].

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dorsett, K.M.; Fears, S.P.; Houlden, H.P. *Innovative Control Effectors (ICE) Phase III*; Final Report for March 1996–July 1997, WL-TR-3059, August 1997; Wright Laboratory, Air Force Material Command: Dayton, OH, USA, 1997.
2. Dorsett, K.M.; Mehl, D. *Innovative Control Effectors (ICE)*; Technical Report ADB212813, Lockheed Martin, Wright Laboratory: Dayton, OH, USA, 1996.

3. Stolk, A. Minimum Drag Control Allocation for the Innovative Control Effector Aircraft. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2017; p. 142.
4. Yao, L.; You, S.; Xiaodong, L.; Bo, G. Control allocation for a class of morphing aircraft with integer constraints based on Levy flight. *J. Syst. Eng. Electron.* **2020**, *31*, 826–840. [\[CrossRef\]](#)
5. Dong, C.; Lu, Y.; Wang, Q. Tracking Control Based on Control Allocation with an Innovative Control Effector Aircraft Application. *Math. Probl. Eng.* **2016**, *2016*, 5037678. [\[CrossRef\]](#)
6. Weilai, J.; Chaoyang, D.; Tong, W.; Qing, W. Fault tolerant control based on control allocation for morphing aircraft model. *J. Beijing Univ. Aeronaut. Astronaut.* **2014**, *40*, 355.
7. Niestroy, M.A.; Dorsett, K.M.; Markstein, K. A tailless fighter aircraft model for control-related research and development. In Proceedings of the AIAA Modeling and Simulation Technologies Conference, Grapevine, TX, USA, 9–13 January 2017. [\[CrossRef\]](#)
8. He, Z.; Hu, J.; Wang, Y.; Cong, J.; Han, L.; Su, M. Sample entropy based prescribed performance control for tailless aircraft. *ISA Trans.* **2022**, *131*, 349–366. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Wu, L.; Park, J.H.; Xie, X.; Ren, Y.; Yang, Z. Distributed adaptive neural network consensus for a class of uncertain nonaffine nonlinear multi-agent systems. *Nonlinear Dyn.* **2020**, *100*, 1243–1255. [\[CrossRef\]](#)
10. Wang, Y.; Hu, J.; Zheng, Y. Improved decentralized prescribed performance control for non-affine large-scale systems with uncertain actuator nonlinearity. *J. Frankl. Inst.* **2019**, *356*, 7091–7111. [\[CrossRef\]](#)
11. Bechlioulis, C.P.; Rovithakis, G.A. A low-complexity global approximation-free control scheme with prescribed performance for unknown pure feedback systems. *Automatica* **2014**, *50*, 1217–1226. [\[CrossRef\]](#)
12. Wang, Y.; Hu, J.; Wang, J.; Xing, X. Adaptive neural novel prescribed performance control for non-affine pure-feedback systems with input saturation. *Nonlinear Dyn.* **2018**, *93*, 1241–1259. [\[CrossRef\]](#)
13. Wang, Y.; Hu, J.; Li, J.; Liu, B. Improved prescribed performance control for nonaffine pure-feedback systems with input saturation. *Int. J. Robust Nonlinear Control* **2019**, *29*, 1769–1788. [\[CrossRef\]](#)
14. He, Z.; Hu, J.; Wang, Y.; Cong, J.; Han, L.; Su, M. Incremental Backstepping Sliding-Mode Trajectory Control for Tailless Aircraft with Stability Enhancer. *Aerospace* **2022**, *9*, 352. [\[CrossRef\]](#)
15. Reiner, J.; Balas, G.J.; Garrard, W.L. Flight control design using robust dynamic inversion and time-scale separation. *Automatica* **1996**, *32*, 1493–1504. [\[CrossRef\]](#)
16. Matamoros, I. Nonlinear Control Allocation for a High-Performance Tailless Aircraft with Innovative Control Effectors—An Incremental Robust Approach. Master's Thesis, Delft University of Technology, Delft, The Netherlands, 2017; p. 107.
17. Sun, L.; Zhou, Q.; Jia, B.; Tan, W.; Li, H. Effective control allocation using hierarchical multi-objective optimization for multi-phase flight. *Chin. J. Aeronaut.* **2020**, *33*, 2002–2013. [\[CrossRef\]](#)
18. Hou, Y.; Lv, M.; Liang, X.; Yang, A. Fuzzy adaptive fixed-time fault-tolerant attitude tracking control for tailless flying wing aircrafts. *Aerosp. Sci. Technol.* **2022**, *130*, 107950. [\[CrossRef\]](#)
19. Werbos, P.J. Building and Understanding Adaptive Systems: A Statistical/Numerical Approach to Factory Automation and Brain Research. *IEEE Trans. Syst. Man. Cybern.* **1987**, *17*, 7–20. [\[CrossRef\]](#)
20. Liu, D.; Wei, Q. Policy Iteration Adaptive Dynamic Programming Algorithm for Discrete-Time Nonlinear Systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 621–634. [\[CrossRef\]](#)
21. Al-Tamimi, A.; Lewis, F.L.; Abu-Khalaf, M. Discrete-Time Nonlinear HJB Solution Using Approximate Dynamic Programming: Convergence Proof. *IEEE Trans. Syst. Man. Cybern. Part B (Cybern.)* **2008**, *38*, 943–949. [\[CrossRef\]](#)
22. Si, J.; Wang, Y.T. Online learning control by association and reinforcement. In Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. Neural Computing: New Challenges and Perspectives for the New Millennium, Como, Italy, 27–27 July 2000; Volume 3, pp. 221–226.
23. Sun, B.; van Kampen, E.J. Intelligent adaptive optimal control using incremental model-based global dual heuristic programming subject to partial observability. *Appl. Soft Comput.* **2021**, *103*, 107153. [\[CrossRef\]](#)
24. Ye, J.; Bian, Y.; Xu, B.; Qin, Z.; Hu, M. Online Optimal Control of Discrete-Time Systems Based on Globalized Dual Heuristic Programming with Eligibility Traces. In Proceedings of the 2021 3rd International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China, 8–11 November 2021; pp. 1–6.
25. Liu, D.; Xue, S.; Zhao, B.; Luo, B.; Wei, Q. Adaptive Dynamic Programming for Control: A Survey and Recent Advances. *IEEE Trans. Syst. Man. Cybern. Syst.* **2021**, *51*, 142–160. [\[CrossRef\]](#)
26. El-Sousy, F.F.M.; Amin, M.M.; Al-Durra, A. Adaptive Optimal Tracking Control Via Actor-Critic-Identifier Based Adaptive Dynamic Programming for Permanent-Magnet Synchronous Motor Drive System. *IEEE Trans. Ind. Appl.* **2021**, *57*, 6577–6591. [\[CrossRef\]](#)
27. Dong, B.; Zhou, F.; Liu, K.; chun Li, Y. Decentralized robust optimal control for modular robot manipulators via critic-identifier structure-based adaptive dynamic programming. *Neural Comput. Appl.* **2018**, *32*, 3441–3458. [\[CrossRef\]](#)
28. Wang, Z.; Lee, J.; Sun, X.; Chai, Y.; Liu, Y. Self-Learning Optimal Control with Performance Analysis using Event-Triggered Adaptive Dynamic Programming. In Proceedings of the 5th International Conference on Crowd Science and Engineering, Jinan, China, 16–18 October 2021.
29. Lin, F.Y. *Robust Control Design: An Optimal Control Approach*; John Wiley and Sons Ltd: Chichester, UK, 2007.
30. Wan, S.; Chang, X.; Li, Q.; Yan, J. Finite-Horizon Optimal Tracking Guidance for Aircraft Based on Approximate Dynamic Programming. *Math. Probl. Eng.* **2019**, *2019*, 8649781. [\[CrossRef\]](#)

31. Nobleheart, W.; Lakshmikanth, G.S.; Chakravarthy, A.; Steck, J.E. Single Network Adaptive Critic (SNAC) Architecture for Optimal Tracking Control of a Morphing Aircraft during a Pull-up Maneuver. In Proceedings of the AIAA Guidance, Navigation, and Control (GNC) Conference, Boston, MA, USA, 19–22 August 2013.
32. Sun, T.; Sun, X.; Sun, A.Y. Optimal Output Tracking of Aircraft Engine Systems: A Data-Driven Adaptive Performance Seeking Control. *IEEE Trans. Circuits Syst. II Express Briefs* **2022**, *69*, 1467–1471. [\[CrossRef\]](#)
33. Du, C.; Li, F.; Yang, C.; Shi, Y.; Liao, L.; Gui, W. Multiphase-Based Optimal Slip Ratio Tracking Control of Aircraft Antiskid Braking System via Second-Order Sliding-Mode Approach. *IEEE/ASME Trans. Mechatronics* **2022**, *27*, 823–833. [\[CrossRef\]](#)
34. Zhou, Y.; van Kampen, E.J.; Chu, Q. Incremental model based online heuristic dynamic programming for nonlinear adaptive tracking control with partial observability. *Aerosp. Sci. Technol.* **2020**, *105*, 106013. [\[CrossRef\]](#)
35. Mannava, A.; Balakrishnan, S.N.; Tang, L.; Landers, R.G. Optimal Tracking Control of Motion Systems. *IEEE Trans. Control Syst. Technol.* **2012**, *20*, 1548–1558. [\[CrossRef\]](#)
36. Nodland, D.; Zargarzadeh, H.; Jagannathan, S. Neural Network-Based Optimal Adaptive Output Feedback Control of a Helicopter UAV. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 1061–1073. [\[CrossRef\]](#)
37. Na, J.; Herrmann, G. Control with Simplified Dual Approximation Structure for Continuous-time Unknown Nonlinear. *IEEE/CAA J. Autom. Sin.* **2014**, *1*, 412–422.
38. Wang, Q.; Gong, L.; Dong, C.; Zhong, K. Morphing aircraft control based on switched nonlinear systems and adaptive dynamic programming. *Aerosp. Sci. Technol.* **2019**, *93*, 105325. [\[CrossRef\]](#)
39. Li, H.; Sun, L.; Tan, W.; Liu, X.; Dang, W. Incremental Dual Heuristic Dynamic Programming Based Hybrid Approach for Multi-Channel Control of Unstable Tailless Aircraft. *IEEE Access* **2022**, *10*, 31677–31691. [\[CrossRef\]](#)
40. Xu, N.; Niu, B.; Wang, H.; Huo, X.; Zhao, X. Single-network ADP for solving optimal event-triggered tracking control problem of completely unknown nonlinear systems. *Int. J. Intell. Syst.* **2021**, *36*, 4795–4815. [\[CrossRef\]](#)
41. Xue, S.; Luo, B.; Liu, D.; Gao, Y. Event-Triggered ADP for Tracking Control of Partially Unknown Constrained Uncertain Systems. *IEEE Trans. Cybern.* **2021**, *52*, 9001–9012. [\[CrossRef\]](#) [\[PubMed\]](#)
42. Xue, S.; Luo, B.; Liu, D.; Gao, Y. Adaptive dynamic programming-based event-triggered optimal tracking control. *Int. J. Robust Nonlinear Control* **2021**, *31*, 7480–7497. [\[CrossRef\]](#)
43. Zhao, J.; Na, J.; Gao, G. Neurocomputing Robust tracking control of uncertain nonlinear systems with adaptive dynamic programming. *Neurocomputing* **2022**, *471*, 21–30. [\[CrossRef\]](#)
44. Snell, S.A.; Enns, D.F.; Garrard, W.L. Nonlinear inversion flight control for a supermaneuverable aircraft. *J. Guid. Control. Dyn.* **1992**, *15*, 976–984. [\[CrossRef\]](#)
45. Ding, L.; Li, S.; Gao, H.; Liu, Y.; Huang, L.; Deng, Z. Adaptive Neural Network-Based Finite-Time Online Optimal Tracking Control of the Nonlinear System With Dead Zone. *IEEE Trans. Cybern.* **2021**, *51*, 382–392. [\[CrossRef\]](#)
46. Wang, N.; Gao, Y.; Zhao, H.; Ahn, C.K. Reinforcement Learning-Based Optimal Tracking Control of an Unknown Unmanned Surface Vehicle. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 3034–3045. [\[CrossRef\]](#)
47. Liu, Y.; Li, S.; Tong, S.; Chen, C.L.P. Adaptive Reinforcement Learning Control Based on Neural Approximation for Nonlinear Discrete-Time Systems with Unknown Nonaffine Dead-Zone Input. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 295–305. [\[CrossRef\]](#)
48. Farzanegan, B.; Suratgar, A.A.; Menhaj, M.B.; Zamani, M. Distributed optimal control for continuous-time nonaffine nonlinear interconnected systems. *Int. J. Control* **2021**, *95*, 3462–3476. [\[CrossRef\]](#)
49. Duan, J.; Liu, Z.; Li, S.E.; Sun, Q.; Jia, Z.; Cheng, B. Adaptive dynamic programming for nonaffine nonlinear optimal control problem with state constraints. *Neurocomputing* **2022**, *484*, 128–141. [\[CrossRef\]](#)
50. Ha, M.; Wang, D.; Liu, D. Data-based nonaffine optimal tracking control using iterative DHP approach. *IFAC-PapersOnLine* **2020**, *53*, 4246–4251. [\[CrossRef\]](#)
51. Bodson, M. Evaluation of optimization methods for control allocation. *J. Guid. Control Dyn.* **2001**, *25*, 703–711. [\[CrossRef\]](#)
52. Singh, S.N.; Steinberg, M.L.; Page, A.B. Nonlinear adaptive and sliding mode flight path control of F/A-18 model. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1250–1262. [\[CrossRef\]](#)
53. Stevens, B.L.; Lewis, F.L.; Eric, E.N.J. *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*, 3rd ed.; John Wiley and Sons, Inc.: Hoboken, NJ, USA, 2016.
54. Durham, W.C.; Bordignon, K.A.; Beck, R. *Aircraft Control Allocation*; Registered office John Wiley and Sons Ltd: Chichester, UK, 2017.
55. Farzanegan, B.; Zamani, M.; Suratgar, A.A.; Menhaj, M.B. A neuro-observer-based optimal control for nonaffine nonlinear systems with control input saturations. *Control Theory Technol.* **2021**, *19*, 283–294. [\[CrossRef\]](#)
56. Song, R.; Wei, Q.; Li, Q. *Adaptive Dynamic Programming: Single and Multiple Controllers*; Science Press: Beijing, China, 2019; p. 278.
57. Wang, D.; Zhao, M.; Ha, M.; Ren, J. Neural optimal tracking control of constrained nonaffine systems with a wastewater treatment application. *Neural Netw.* **2021**, *143*, 121–132. [\[CrossRef\]](#)
58. Stone, M.H. The Generalized Weierstrass Approximation Theorem. *Math. Mag.* **1948**, *21*, 167. [\[CrossRef\]](#)

59. Abu-Khalaf, M.; Lewis, F.L. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach. *Automatica* **2005**, *41*, 779–791. [[CrossRef](#)]
60. Cong, J.; Hu, J.; Wang, Y.; He, Z.; Han, L.; Su, M. Fault-Tolerant Attitude Control Incorporating Reconfiguration Control Allocation for Supersonic Tailless Aircraft. *Aerospace* **2023**, *10*, 241. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.