Proceedings of the 2019 IEEE/ASME
International Conference on Advanced Intelligent Mechatronics
Hong Kong, China, July 8-12, 2019

Adaptive Incremental Nonlinear Dynamic Inversion Control Based on Neural Network for UAV Maneuver

Su Cao[1], Lincheng Shen[1], Renshan Zhang[1], Huangchao Yu[1*] and Xiangke Wang[1]

Abstract— **In this paper, a flight control strategy based on incremental nonlinear dynamic inversion (INDI) using properties of general flight control systems and nonlinear dynamic inversion by feeding back angular accelerations is presented. The INDI control law is based on a linearized approximation of incremental plant dynamics and reduces dependence on modeling accuracy. However, there is still an un-negligible nonlinear character of unmanned aerial vehicle (UAV) during high angle maneuvers. The main contributions of this article are 1) proposing an adaptive neural network compensation method : INDI with neural network(INDI_NN) to correctly consider the model uncertainties; 2) a quaternion based reference model which is suitable for vehicles to experience a high angle maneuver. The simulation results support the proposed control scheme in getting better tracking performance during large range of attitudes. Hence, the proposed control method makes INDI controller more practical and more suitable for high angle maneuver like Immelman Turn.**

I. INTRODUCTION

Currently, the maneuverability, fault tolerance and survivability of unmanned aerial vehicles(UAVs) provides a strong motivation for the development of low cost and stability augmented UAV. However, the limitations of the control system using current approaches are often unavoidable, such as the consumption of design and the robustness. Therefore, one of the objective of the UAV design is to overcome the shortage of robustness, which aims to implement a versatile control scheme using a primitive aerodynamic model. This paper presents a novel adaptable flight control scheme based on neural network, which provides a solution to eliminate the limitations mentioned above.

Since the linear control theory in flight control systems is always based on the approximation of nonlinear aircraft dynamics [1], [2], it does not exploit the physical ability for UAV to operate high angle maneuvers in aerobatics, obstacle avoidance or flight combat [3]. Therefore, the implementation of nonlinear control for UAV is considered as an effective way to enable robust control of the nonlinear system and several successful attempts have been made. Nonlinear dynamic inversion(NDI), which is designed to use an aerodynamic model to linearize the dynamics of UAV, has been utilized in many applications [4]–[6]. However, NDI suffers from the sensitivity to parameter uncertainties [7], which involves all of the UAV's forces and dynamics. since it is often impossible to obtain an accurate model, many attempts are focused on making the controller less dependent on the model.

In [8], a concept of feeding back angular accelerations is introduced. The rotational dynamic equations are transferred into an incremental form, resulting in incremental nonlinear dynamic inversion(INDI). Compared to NDI, INDI replace many part of the model with measured angular acceleration, and many un-modeled dynamics are compensated. However, there still remains some uncertainties which cannot be eliminated [9]. Neural Network(NN) have been proposed as an adaptive controller for high assurance system [10] and also for flight control of small UAV [11]. The learning capabilities of neural networks provide the property of real-time adaptivity during flight and reduce such effects which are often too hard to analytically calculated. In this paper, by combining a neural network adaptive controller with INDI controller, the problems related to model uncertainties and sensor measurement are solved.

The structures of this paper are as follows. First, a fixed-wing UAV model will be introduced in Sec. II, and a novel quaternion method based NDI will be discussed. Sec. III will deal with the analysis of INDI and the neural network based adaptive controller. Finally, in Sec. IV, the simulation setup is described and the results of simulation will be presented.

II. FIXED-WING UAV DYNAMICS

A. Model Definition

In this paper, the control strategy is tested in simulations of an F-16 like UAV, which has control surfaces including a elevators, a pair of rudders and ailerons. For this multiple-input-multiple-ouput (MIMO) system, the established attitude dynamics and kinematics model are as follows [12].

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x},\boldsymbol{u}) \\ \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}) \end{cases} \tag{1}$$

In Eq. (1), $\boldsymbol{x}$ is the state vector of UAV dynamics, which consists of attack angle α, side slip angle β, roll angle ϕ, pitch angle θ, yaw angle ψ and p, q, r, representing steady axis roll, pitch and yaw rate respectively. The control vector $\boldsymbol{u}$ contains three deflection angles of control surfaces: aileron δ_a, elevator δ_e and rudder δ_r. Because the main target of this paper is the attitude control, the output vector $\boldsymbol{y}$ only contains three Euler angles,

$$\boldsymbol{y} = [\phi, \theta, \psi]^T. \tag{2}$$

The rotational dynamics of a UAV with conventional control surfaces: ailerons δ_a, elevator δ_e and rudder δ_r is

This work is supported in part by Hunan Provincial Natural Science Foundation of China under Grant 2019JJ50717.

[1] Institute of Unmanned Systems, National University of Defense Technology, Changsha, Hunan, 410073, China
[*] Corresponding author: huangcha@ualberta.com

described in Eq. (3).

$$\begin{cases} \dot{p} = f_p(p,q,r) + g_p(p,q,r,\alpha,\beta,\delta_a,\delta_r,V) \\ \dot{q} = f_q(p,q,r) + g_q(p,q,r,\alpha,\beta,\delta_e,V) \\ \dot{r} = f_r(p,q,r) + g_r(p,q,r,\alpha,\beta,\delta_a,\delta_r,V) \end{cases} \tag{3}$$

where α is the attack angle, β is the sideslip angle and V is the velocity. The attitude motion equation is illustrated in Eq. (4).

$$\begin{cases} \dot{\phi} = f_\phi(p,q,r,\phi,\theta) \\ \dot{\theta} = f_\theta(q,r,\theta,\phi) \\ \dot{\psi} = f_\psi(q,r,\theta,\phi) \\ \dot{\alpha} = f_\alpha(p,q,r,\alpha,\beta,\mu,\delta_a,\delta_r) \\ \dot{\beta} = f_\beta(p,r,\alpha,\beta,\mu,\delta_a,\delta_r) \end{cases} \tag{4}$$

The derivative function f_p, f_q, f_r, f_ϕ, f_θ, f_ψ, f_α and f_β can be found in reference [13]–[15].

According to the time-scale separation assumption [16], the UAV dynamics can be separated into two parts due to different responding time: the fast(inner) states and slow(outer) states. the fast time-scale states are the axis rotational rates,

$$\boldsymbol{x}_{fast} = [p,q,r]^T, \tag{5}$$

and the control variables are the deflection angle of control surfaces,

$$\mathbf{u}_{fast} = [\delta_a,\delta_e,\delta_r]^T. \tag{6}$$

Moreover, the slow time scale states are given by Eq.(7), which are treated as constants in the slow time scale.

$$\boldsymbol{x}_{slow} = [\alpha,\beta,\phi,\theta,\psi]^T \tag{7}$$

B. Quaternion Method Based NDI

During the high maneuver flight of UAV, the Euler angles change drastically. Especially for the tumbling flight, the aircraft will fly across the singular point ($\theta = \pm\frac{\pi}{2}$), which leads to an unstable solution of the equation of motion. Moreover, the input step signal can also lead to instability. Hence, we introduce a quaternion based method for the slow loop nonlinear dynamic inversion.

Recall the Eq. (2), the output derivative can be described as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{8}$$

The attitude kinematics model represented by Euler angles contains trigonometric functions, which are only suitable for small-angle maneuvering and stabilization control. In order to overcome the singularity of the Euler angle model, a quaternion method is introduced, which eliminates singularity points and avoids the pulse signal originated from step signal differentiation. So quaternion is widely used to deal with large-angle maneuvers.

First, introducing the rotational equation using quaternion,

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}. \tag{9}$$

Using the generalized inverse method, the quaternion command signals can be obtained as follows:

$$\begin{bmatrix} p_c \\ q_c \\ r_c \end{bmatrix} = \begin{bmatrix} 2q_0\dot{q}_1 - 2q_1\dot{q}_0 - 2q_2\dot{q}_3 + 2q_3\dot{q}_2 \\ 2q_0\dot{q}_2 - 2q_2\dot{q}_0 + 2q_1\dot{q}_3 - 2q_3\dot{q}_1 \\ 2q_0\dot{q}_3 - 2q_1\dot{q}_2 + 2q_2\dot{q}_1 - 2q_3\dot{q}_0 \end{bmatrix} = 2\boldsymbol{H}^T \begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix}, \tag{10}$$

where matrix $\boldsymbol{H}$ is

$$\boldsymbol{H}^T = \begin{bmatrix} -q_1 & q_0 & q_3 & -q_2 \\ -q_2 & -q_3 & q_0 & q_1 \\ -q_3 & q_2 & -q_1 & q_0 \end{bmatrix}^T. \tag{11}$$

In the Eq. (10), $\dot{q}_0$, $\dot{q}_1$, $\dot{q}_2$, $\dot{q}_3$ are calculated based on the quaternion command and the proportional gains, that is, approximately equal the time difference. Therefore, the desired angular rates are

$$\begin{bmatrix} p_c \\ q_c \\ r_c \end{bmatrix} = 2\boldsymbol{H}^T \begin{bmatrix} k_0 \cdot (q_{0_c} - \hat{q}_0) \\ k_1 \cdot (q_{1_c} - \hat{q}_1) \\ k_2 \cdot (q_{2_c} - \hat{q}_2) \\ k_3 \cdot (q_{3_c} - \hat{q}_3) \end{bmatrix}. \tag{12}$$

As shown in Eq. (12), the proportional gains (k_0,k_1,k_2,k_3) is usually set to 1 or 2 rad/s [10]. $\hat{q}_0$, $\hat{q}_1$, $\hat{q}_2$, $\hat{q}_3$ are the output of the feedback filter. And q_{0_c}, q_{1_c}, q_{2_c}, q_{3_c} are the quaternion command calculated from the reference model.

III. Adaptive Controller Design

The overall INDI_NN control structure is illustrated in Fig. 1. The system contains several parts includes the control command, NDI controller for slow loop, INDI controller for fast loop, the adaptive controller and the plant.

A. Incremental Nonlinear Dynamic Inversion

The equation of motion is nonlinear, input non-affine,

$$\begin{aligned} \dot{\boldsymbol{\omega}} &= \boldsymbol{f}(\boldsymbol{\omega},\boldsymbol{\delta}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{\omega}) \end{aligned} \tag{13}$$

Where $\boldsymbol{\omega}$ is the state vector and $\boldsymbol{\delta}$ is the deflection command. By assuming that the deflection of control surface is achieved without delay, the output is differentiated as follows [16]

$$\begin{aligned} \dot{\boldsymbol{y}} &= \dot{\boldsymbol{\omega}}_0 + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{\omega}} d\boldsymbol{\omega} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{\delta}} d\boldsymbol{\delta} \\ &= \dot{\boldsymbol{\omega}}_0 + \boldsymbol{B}(\boldsymbol{\omega}_0,\boldsymbol{\delta}_0) d\boldsymbol{\delta} \end{aligned} \tag{14}$$

Eq. (14) is based on the linear approximation around $\boldsymbol{\delta}_0$ and $\boldsymbol{\omega}_0$, with the assumption that $d\boldsymbol{\omega}$ is much smaller than $d\boldsymbol{\delta}$ and can be neglected [13]. Replacing the angular acceleration $\dot{\boldsymbol{y}} = \dot{\boldsymbol{\omega}}$ by the desired pseudo-control signal $\dot{\boldsymbol{v}}_{des}$, the control increment $d\boldsymbol{\delta}$ can be derived:

$$d\boldsymbol{\delta} = \boldsymbol{B}^{-1}(\boldsymbol{\omega}_0,\boldsymbol{\delta}_0)(\dot{\boldsymbol{v}}_{des} - \dot{\boldsymbol{\omega}}_0). \tag{15}$$

In Eq. (15), where $\dot{\boldsymbol{\omega}}_0$ is the estimated output derivative and $\boldsymbol{B}^{-1}$ is the general inverse of input matrix $\boldsymbol{B}$. consider the discrete time approximation of the derivative: $\dot{\omega}_0 = (\omega_0 - \omega_0 z^{-1}) T_s^{-1}$, where T_s is the sample time.

Note that the output $\dot{\boldsymbol{\omega}}_0$ represent the current estimated angular acceleration, which is measured by differentiating

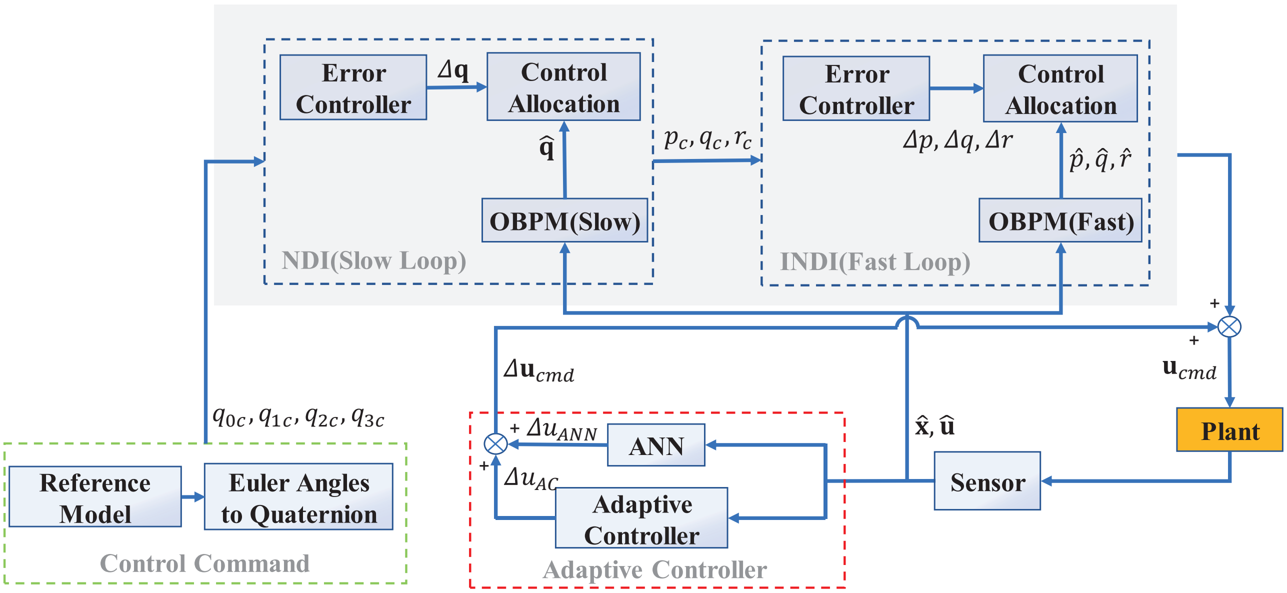

Fig. 1. Closed-Loop System of Adaptive INDI_NN Controller

the angular rates. From [17], the response speed grows when the angular acceleration is measured faster.

Theorem 1: Considering the system described in Eq.(1) and the ouput control δ defined below, the system illustrated in Fig.1 is locally uniformly ultimately bounded when the adaptive controller is not considered.

$$\boldsymbol{\delta} = \boldsymbol{\delta}_0 + d\boldsymbol{\delta} = \delta_0 + \boldsymbol{B}^{-1}(\boldsymbol{\omega}_0, \boldsymbol{\delta}_0)(\dot{\boldsymbol{v}}_{des} - \dot{\boldsymbol{\omega}}_0) \tag{16}$$

The theorem is based on the assumption that the actuator of UAV is a first-order system $A(z)$, and the sensors a second-order filter $H(z)$. The proof of this theorem is here.

Proof: First, according to ref. [17], [18], the transfer function of the system is shown in Eq. (17).

$$TF_{\omega_{err}\to\omega} = \mathrm{I}\left(1 - z^{-1}A(z)H(z)\right)^{-1}A(z) \tag{17}$$

Considering Eq. (12), the transfer function of NDI system $\boldsymbol{TF}_{\boldsymbol{q}_c\to\boldsymbol{v}}$ only contains the diagonal matrices [19]. By neglecting the filter $H(z)$, the transfer function of the INDI & NDI $\boldsymbol{TF}_{\boldsymbol{q}_c\to\boldsymbol{q}}$ is $IA(z)$. This means that the closed-loop transfer function of the entire system is the actuator dynamics $A(z)$. As long as the actuator dynamics are stable, the transfer function will go to zero over time. ∎

B. Neural Network Compensation

In this paper, standard feed-forward networks are implemented to cancel the modeling error, The output of Neural Network is

$$\boldsymbol{y}_{nn} = \boldsymbol{w}^{(1)^T}\boldsymbol{\theta}\left(\boldsymbol{w}^{(2)^T}\boldsymbol{x}\right), \tag{18}$$

with,

$$\boldsymbol{w}^{(i)} = \begin{bmatrix} \boldsymbol{\vartheta}_1 & \boldsymbol{w}^{(i)}{}_{11} & \boldsymbol{w}^{(i)}{}_{12} & \cdots \\ \boldsymbol{\vartheta}_2 & \boldsymbol{w}^{(i)}{}_{21} & \boldsymbol{w}^{(i)}{}_{22} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}^T, i = 1, 2 \tag{19}$$

$$\mathbf{x}_{nn} = [1, q_0, \ldots, q_3, p, q, r, q_{0d}, \ldots, q_{3d}, p_d, q_d, r_d]^T, \tag{20}$$

where $\boldsymbol{\vartheta}_i$ is the bias, q_{id}, p_d, q_d, r_d are the pseudo-control inputs and q_i, p, q, r is the estimated states.

The adaptive laws of neural network weights $\dot{\boldsymbol{w}}^{(1)}$ and $\dot{\boldsymbol{w}}^{(2)}$ are

$$\dot{\boldsymbol{w}}^{(1)} = \gamma^{(1)} \cdot \left[\boldsymbol{x}_{nn}\boldsymbol{w}^{(2)}\boldsymbol{\theta}'\left(\boldsymbol{w}^{(1)}\boldsymbol{x}_{nn}\right) - \lambda\boldsymbol{w}^{(1)^T}\right] \tag{21}$$

$$\begin{aligned}\dot{\boldsymbol{w}}^{(2)} = \gamma^{(2)} \cdot \Big[&\boldsymbol{\theta}\left(\boldsymbol{w}^{(1)}\boldsymbol{x}_{nn}\right) - \lambda\boldsymbol{w}^{(2)^T} \\ &-\boldsymbol{\theta}'\left(\boldsymbol{w}^{(1)}\boldsymbol{x}_{nn}\right)\boldsymbol{w}^{(1)}\boldsymbol{x}_{nn}\Big].\end{aligned} \tag{22}$$

where $\gamma^{(1)}$ and $\gamma^{(2)}$ are positive matrices representing the so called learning rates of each neural layer and λ is a positive constant of e-modification [20]. λ determines the upper bounds of the neural network weights $\boldsymbol{w}^{(1)}$ and $\boldsymbol{w}^{(2)}$. The Lyapunov function of the stability prove can be selected as

$$\begin{aligned}V(\boldsymbol{x}_{nn}) = &\tfrac{1}{2}\tilde{\boldsymbol{x}}_1^T\tilde{\boldsymbol{x}}_1 + \tfrac{1}{2}\tilde{\boldsymbol{x}}_2^T\tilde{\boldsymbol{x}}_2 + \tfrac{1}{2\gamma^{(1)}}\left[\tilde{\boldsymbol{w}}^{(1)^T}\tilde{\boldsymbol{w}}^{(1)}\right] \\ &+\tfrac{1}{2\gamma^{(2)}}\left[\tilde{\boldsymbol{w}}^{(2)^T}\tilde{\boldsymbol{w}}^{(2)}\right].\end{aligned} \tag{23}$$

From [21], if $\gamma^{(1)}$, $\gamma^{(2)}$ and λ are chosen to fulfill constraints.

The derivative of the chosen Lyapunov function satisfy

$$\dot{V}(\boldsymbol{x}_{nn}) \leq -2\mu V(\boldsymbol{x}_{nn}) + C. \tag{24}$$

In Eq. (24), μ and C are bounded constants. Hence, when $V(\boldsymbol{x}_{nn}) \geq C/2\mu$, the error states $\tilde{\boldsymbol{x}}_1$ and $\tilde{\boldsymbol{x}}_2$ and the weight estimation error are therefore bounded.

C. Adaptive Controller

Introducing the robustifying term $\boldsymbol{v}$ and ξ:

$$\boldsymbol{v} = -\frac{z_2\xi}{\|z_2\|\xi + \varepsilon}\xi \tag{25}$$

$$\xi = k_v\left(\boldsymbol{Z_M} + \left\|\hat{\boldsymbol{Z}}\right\|_F\right)(\|\boldsymbol{x_1}\| + \|\boldsymbol{x_2}\|), \tag{26}$$

where $\boldsymbol{Z}$ equals $diag[w_1, w_2]$ and k_v is the proportional gain.

The final control input $\boldsymbol{u}$ is defined as follows:

$$\boldsymbol{u} = \boldsymbol{u}_{INDI} + \boldsymbol{w}^{(1)^T} f\left(\boldsymbol{w}^{(2)^T}\boldsymbol{x}\right) + \boldsymbol{v}. \tag{27}$$

According to [21] and Theorem 1, the tracking errors and the parameter estimation error of the neural networks converge to a compact set. Furthermore, the size of the set is adjustable by tuning the design parameters [21].

IV. Results and Discussion

A. Case Studies

In this paper, A high angle maneuver is tested and the robust performance of three control system is compared in simulations using Simulink. The first one is NDI, second is INDI and the third one is adaptive INDI_NN. In NDI and INDI case, the fast loop applies NDI and INDI controller respectively, and the adaptive controller is removed. Actually NDI and INDI all can be implemented in UAV with the sensors such as IMU and airspeed tube. Since the quaternion method is utilized, the sensor of attack/sideslip angle is no longer needed. The model mismatch of UAV dynamics are considered in the simulation.

Proposed control strategies next section are subjected to a high angle maneuver flight test. In this paper, we choose a classic flight maneuver: the Immelman Turn. The reference command of attitude is based on the data obtained from the manually controlled flight trajectory. This maneuver contains a rapid and simultaneous pitch-up, roll and yaw motion. The flight speed maintains at $100[m/s]$ and the limitation of δ_a, δ_e and δr are $[-21.5°, 21.5°]$, $[-25°, 25°]$ and $[-30°, 30°]$ separately. The structure of neural network is chosen as a single hidden-layer one, and the number of nodes is 10. The adaptive parameters are chosen as: $\gamma^{(1)} = 1.75$, $\gamma^{(2)} = 0.17$ and $\lambda = 7.65$

B. Results

The simulation results of slow-loop performance subject to modeling mismatch are shown in Fig. 2-4. The response to the uncertainties described in Table I are presented.

In order to compare the performance of the three control methods in Fig. 2-4, the tracking error norms L_2 are used.

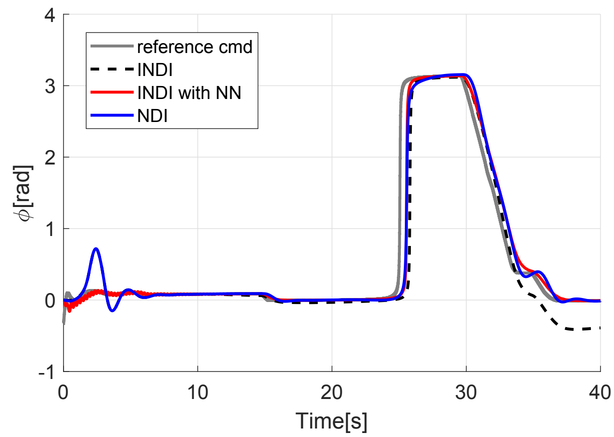

Fig. 2. Roll angle(ϕ) tracking results

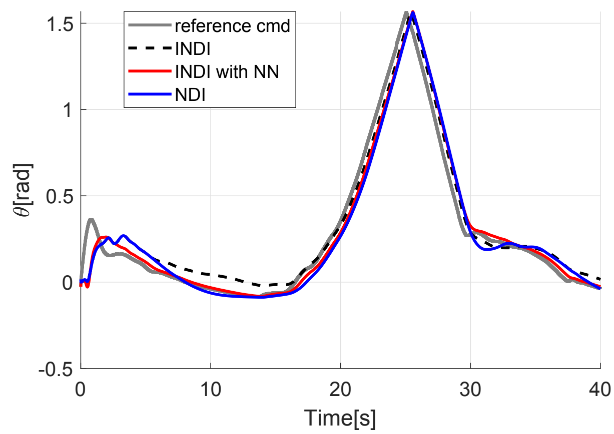

Fig. 3. Pitch angle(θ) tracking results

The normalized L_2 norm of the tracking error $\boldsymbol{e}$ is

$$\|\boldsymbol{e}\|_{L_2} = \frac{\|\boldsymbol{e}\|_{L_2}}{\|\boldsymbol{v}_{ref}\|_{L_2}}. \tag{28}$$

The Euler angles tracking error comparison between NDI, INDI and INDI_NN for UAV model case described in Table I is presented in Table II. It is observed from these simulation results that for considered UAV model with uncertainties, INDI_NN adapter controller provides better tracking performance than those of INDI and NDI. From Fig.4, the response of the INDI_NN does not show any sensitivity to the modeling mismatch between OBPM and plant, and the results are nearly identical to those of the reference command. However, conventional NDI and INDI suffer from the model mismatch. The responses for each realization are unsatisfactory, with non-negligible oscillation in NDI and deviation in INDI realization. Since the INDI controller uses angular acceleration, it performs a faster and more stable response than those of NDI. Note the beginning of the response of INDI_NN controller shows a noticeable oscillation which is caused by the mismatch between neural-network and ideal mismatch part. After a short time of

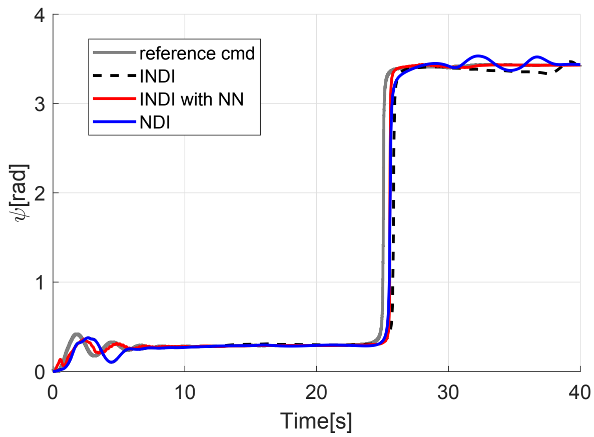

Fig. 4. Yaw angle(ψ) tracking results

TABLE I
THE MODEL MISMATCH SET IN TEST CASE

Parameters	Mismatch
Aerodynamic Model	
Cl_0, Cm_0, Cn_0, $Cl_{p,q,r}$, $Cm_{p,q,r}$, $Cn_{p,q,r}$	-20%
$\delta Cl_{LEF,\delta_a,\delta r}$, $\delta Cn_{LEF,\delta_a,\delta r}$	$+20\%$
Center of Gravity, Mass	-20%
Moment of Inertia	
I_{xx}, I_{yy}, I_{zz}	$+20\%$
I_{xz}, I_{yz}, I_{xz}	-20%

network learning, the weight matrices $\boldsymbol{w}^{(1)}$ and $\boldsymbol{w}^{(2)}$ converge and the oscillation vanishes.

TABLE II
L_2 NORMALIZED TRACKING ERROR

Test Case	NDI	INDI	INDI with NN
ϕ	0.2749	0.2710	0.2660
θ	0.2118	0.1750	0.1546
ψ	0.1971	0.1769	0.1461

In Fig.5-Fig.7, the responses of fast-loop p,q,r using three controllers mentioned above are presented respectively. These plots are only presented for the response applying reference command in INDI realization, because their characteristics vary analogously in the Immelman Turn. From Fig.3(b) and (c), the INDI and INDI_NN controller response to the reference command with negligible deviation, but the performance of NDI controller is unsatisfactory. Since NDI controller only uses the angular velocity $\boldsymbol{\omega}$, the response time is much longer and large overshoot is unavoidable. Although the model mismatch does not affect the INDI controller in p and q axis, it causes a un-negligible deviation in r. The result shows that the INDI controller cannot eliminate the modeling mismatch only with angular accelerations. By employing neural network and adaptive item, the response of INDI_NN controller is identical to reference command after convergence, which ensures the slow-loop performance in UAV maneuver.

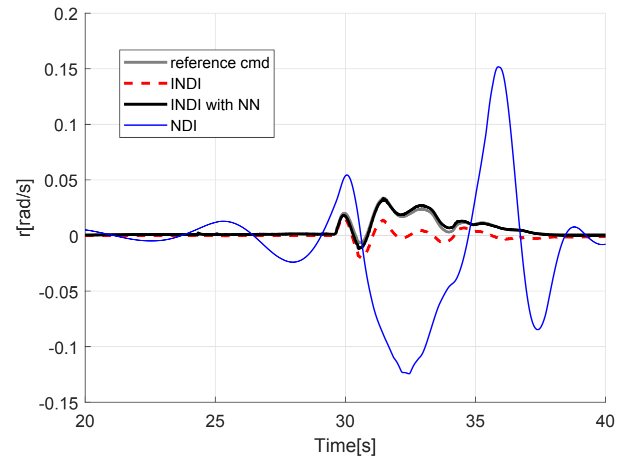

Fig. 5. Yaw rate tracking results

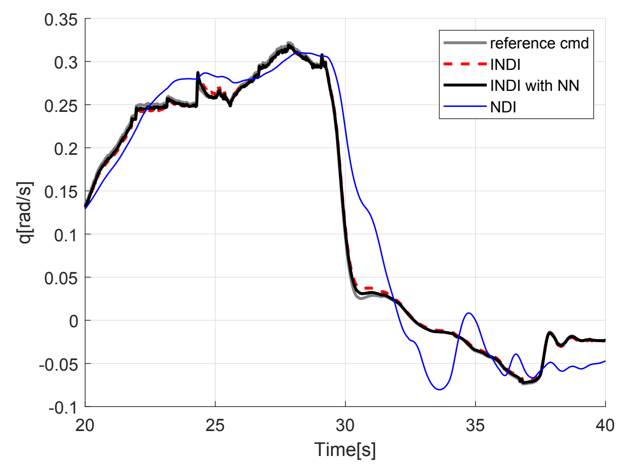

Fig. 6. Pitch rate tracking results

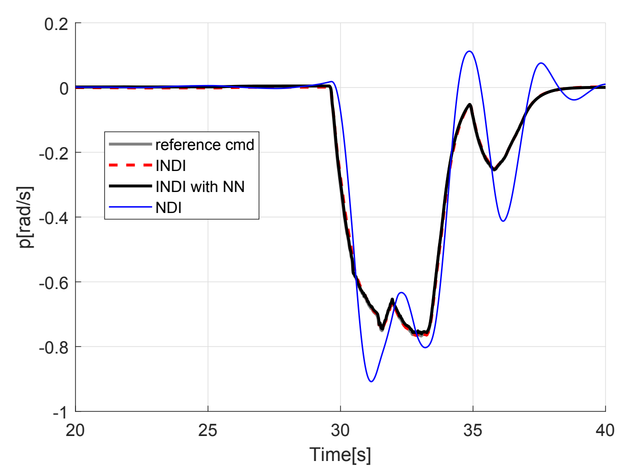

Fig. 7. Roll rate tracking results

V. Conclusion

A neural-network based adaptive incremental nonlinear dynamic inversion(INDI) control system for UAV maneuver has been described in this paper. The reference control has been developed using a quaternion based method, which avoids the singular problem in slow-loop NDI controller design. By adding a neural network based adaptive controller, the INDI_NN controller introduced performs an improved robust capability compared with regular NDI and INDI controllers for unmodeled UAV dynamics. The simulation shows good tracking performance and result in a versatile controller.

References

[1] Brian Stevens, Frank Lewis, and Eric N. Johnson. *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems, Third Edition*. 2015.

[2] Stephen Boyd and Craig Barratt. Linear controller design: limits of performance. Technical report, Stanford University Stanford United States, 1991.

[3] Luat T Nguyen, Marilyn E Ogburn, William P Gilbert, Kemper S Kibler, Philip W Brown, and Perry L Deal. Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability.[f-16]. 1979.

[4] William MacKunis, Zachary D Wilcox, M Kent Kaiser, and Warren E Dixon. Global adaptive output feedback tracking control of an unmanned aerial vehicle. *IEEE Transactions on Control Systems Technology*, 18(6):1390–1397, 2010.

[5] Jean-Jacques E Slotine, Weiping Li, et al. *Applied nonlinear control*, volume 199. Prentice hall Englewood Cliffs, NJ, 1991.

[6] Jacob Reiner, Gary J Balas, and William L Garrard. Flight control design using robust dynamic inversion and time-scale separation. *Automatica*, 32(11):1493–1504, 1996.

[7] RR Da Costa, QP Chu, and JA Mulder. Reentry flight controller design using nonlinear dynamic inversion. *Journal of Spacecraft and Rockets*, 40(1):64–71, 2003.

[8] Barton Bacon and Aaron Ostroff. Reconfigurable flight control using nonlinear dynamic inversion with a special accelerometer implementation. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 4565, 2000.

[9] Aaron J Ostroff and Barton J Bacon. Enhanced ndi strategies for reconfigurable flight control. In *American Control Conference, 2002. Proceedings of the 2002*, volume 5, pages 3631–3636. IEEE, 2002.

[10] Johann M Ph Schumann and Yan Liu. *Applications of neural networks in high assurance systems*, volume 268. Springer, 2010.

[11] Travis Dierks and Sarangapani Jagannathan. Output feedback control of a quadrotor uav using neural networks. *IEEE transactions on neural networks*, 21(1):50–66, 2010.

[12] Lars Sonneveldt. Nonlinear f-16 model description. *Delft University of Technology, Netherlands*, 2006.

[13] S Sieberling, QP Chu, and JA Mulder. Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction. *Journal of guidance, control, and dynamics*, 33(6):1732–1742, 2010.

[14] Gabriele Di Francesco and Massimiliano Mattei. Modeling and incremental nonlinear dynamic inversion control of a novel unmanned tiltrotor. *Journal of Aircraft*, 53(1):73–86, 2015.

[15] Ronald van't Veld, Erik-Jan Van Kampen, and Q Ping Chu. Stability and robustness analysis and improvements for incremental nonlinear dynamic inversion control. In *2018 AIAA Guidance, Navigation, and Control Conference*, page 1127, 2018.

[16] Pranav Bhardwaj, Venkata Sravan Akkinapalli, Jiannan Zhang, Saurabh Saboo, and Florian Holzapfel. Adaptive augmentation of incremental nonlinear dynamic inversion controller for an extended f-16 model. In *AIAA Scitech 2019 Forum*, page 1923, 2019.

[17] Pranav Bhardwaj, Stefan A Raab, Jiannan Zhang, and Florian Holzapfel. Integrated reference model for a tilt-rotor vertical take-off and landing transition uav. In *2018 Applied Aerodynamics Conference*, page 3479, 2018.

[18] Xuerui Wang, Erik-Jan Van Kampen, Q Ping Chu, and Peng Lu. Stability analysis for incremental nonlinear dynamic inversion control. In *2018 AIAA Guidance, Navigation, and Control Conference*, page 1115, 2018.

[19] Ewoud JJ Smeur, Qiping Chu, and Guido CHE de Croon. Adaptive incremental nonlinear dynamic inversion for attitude control of micro air vehicles. *Journal of Guidance, Control, and Dynamics*, 38(12):450–461, 2015.

[20] Thomas Krüger, Philipp Schnetter, Robin Placzek, and Peter Vörsmann. Fault-tolerant nonlinear adaptive flight control using sliding mode online learning. *Neural Networks*, 32:267 – 274, 2012. Selected Papers from IJCNN 2011.

[21] Taeyoung Lee and Youdan Kim. Nonlinear adaptive flight control using backstepping and neural networks controller. *Journal of Guidance, Control, and Dynamics*, 24(4):675–682, 2001.