Attitude Adjustment of Quadrotor Aircraft Platform via a Data-Driven Model Free Adaptive Control Cascaded with Intelligent PID

Shida Liu, Zhongsheng Hou, Jian Zheng

Advanced Control Systems Laboratory, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, 100044, China

E-mail: zhshhou@bjtu.edu.cn

Abstract: In this work, a novel cascade model free adaptive control method (MFAC) with an intelligent PID (i-PID) algorithm is proposed for the attitude adjustment of a multi-input multi-output (MIMO) quadrotor aircraft. Since the quadrotor system may be open-loop unstable, it is first stabilized using i-PID controllers to ensure its controllability. Then the closed-loop quadrotor system controlled via i-PID controllers is controlled by using MFAC. In this way, the MIMO aircraft system is decoupled via a novel concept called Pseudo Jacobi Matrix (PJM), such that the conventional complex modelling and decoupling process can be avoided. The comparison experimental results implemented on a practical quadrotor aircraft verify the effectiveness of proposed approach.

Key Words: Quadrotor Aircraft; Data-Driven Control; Model Free Adaptive Control (MFAC); Intelligent PID Control

1 INTRODUCTION

In recent years, due to the rapid development of the aircraft industry, there exist increasing studies concerning with quadrotor aircrafts. In particular, the attitude adjustment for a quadrotor aircraft is widely researched until now.

Currently, the existing control methods for the attitude adjustment of a quadrotor include linear quadratic regulators (LQR) [1], sliding mode control methods [2], backstepping control approaches [3] and model predictive attitude control approaches [4], etc.. For these methods, the controllers are must designed relying on accurate mathematical models of corresponding quadrotors.

The quadrotor aircraft, however, is a typically complex multi-input and multi-output (MIMO) system with strong coupling and nonlinearity, making the mathematical model of the quadrotor aircraft become difficult to establish. For this reason, the controller designed by directly using the input-output (I/O) data is a novel way for addressing this problem, since the I/O data of the system contain all information concerning the actual dynamics.

To date, PID methods are deeply applied for the quadrotor aircrafts control due to its simple structure [5, 6]. All PID controllers, however, may require parameters retuning because the control process usually changes under large operating ranges. Moreover, some identification procedures should be needed in order to obtain the controller parameters, which is quite inconvenient in most cases.

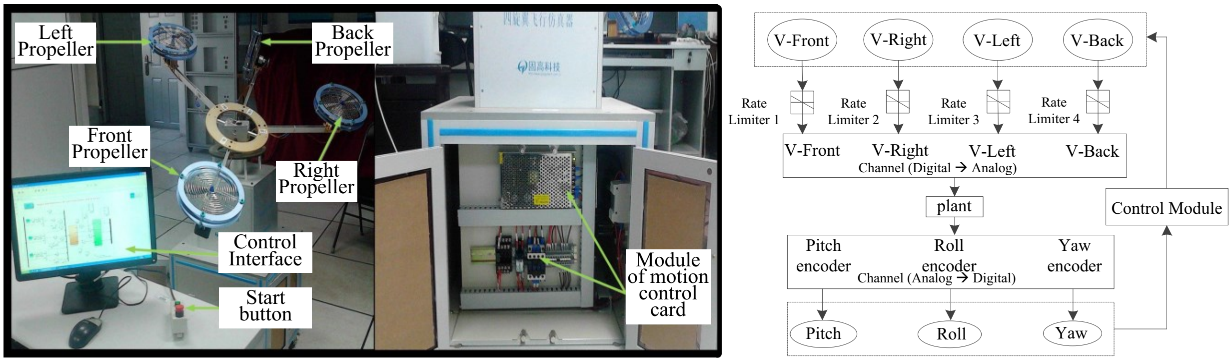
Recently, an intelligent PID (i-PID) control algorithm is proposed and widely applied [7]. Compared with conventional PID methods, the i-PID controller is designed based on the framework of differential algebra and its parameters tuning become quite straightforward even for highly nonlinear or time varying systems.

However, the i-PID controller is originally proposed for single-input ingle-output (SISO) system. For MIMO systems, although i-PID can be implemented, it is lack of rigorous theoretical guidance for application.

As another typical data-driven control algorithm, model free adaptive control (MFAC) [8, 9] has favorable control performance for MIMO time-varying nonlinear systems, even the system is with strong coupling. The essential idea is that, for a class of nonlinear discrete-time systems, using a virtual equivalent dynamic linearization data model based on a novel concept named Pseudo Jacobi Matrix (PJM) at every current operation point replaces the general discrete-time nonlinear system, where the unknown time-varying parameter of the data model is online estimated using the I/O data controlled from the plant. Then, the model-free adaptive control strategy is designed base on the established local data model. Recently, MFAC has been successfully applied in many practical applications, e.g., injection moulding control process [10], artificial heart control [11], wind turbine [12] and power system [13], etc..

In this work, a novel model free adaptive control method combined with the intelligent PID method (i-PID-MFAC) is proposed for the attitude adjustment of a quadrotor aircraft. First, since the quadrotor system may be open-loop unstable, the i-PID controllers are introduced serving as the stabilizing controllers. After that, the closed-loop quadrotor system controlled via i-PID controllers can be considered as a new MIMO system and is then controlled by using compact form MFAC algorithm, by which the quadrotor system is further decoupled via PJM, avoiding conventional complex decoupling process existing in model-based control methods. Experimental results verify

This work was supported by the International Cooperation Program of National Science Foundation of China (61120106009), the National Natural Science Foundation of China under Grant (61433002).



(a) (b)

Fig. 1 Quad-rotor aircraft control system. (a). Equipment. (b). Schematic diagram.

the effectiveness of proposed control scheme.

The rest of this paper is structured as follows. In Section 2, the description of the quadrotor platform is presented. In Section 3, the specific controller design process of i-PID-MFAC for the quadrotor system is described. Experimental results verifying the effectiveness of proposed control scheme are presented in Section 4. The conclusion is drawn in Section 5.

2 PLATFORM DESCRIPTION

2.1 Introduction of Quadrotor

The equipment and the schematic diagram of the quadrotor aircraft control system are shown in Fig. 1. The aircraft moves around a fixed central point, such that it can show the attitude variation via the changes of three angles, including the pitch, roll, and yaw angles.

As shown in Fig 1 (a), when the sum of the rotating speed of the left propeller and the right propeller is larger than that of the front propeller, the quadrotor body will do the pitch motion. Consequently, the pitch angle is generated between the body of the quadrotor and the horizontal plane. Moreover, when the rotating speed of the left propeller is larger (smaller) than that of the right propeller, the body will do the flip-flop movement. At this time, the roll angle is generated between the body and the horizontal plane. In addition, when the rotating speed of the back propeller is nonzero, the quadrotor will do the yaw motion and the corresponding angle is denoted as the yaw angle.

As shown in Fig 1 (b), the information of three angles (outputs) can be collected via corresponding incremental encoders in Analog to Digital (A\D) channel. Then, the digital values of three angles are transferred to the control module. According to the received information, the control module can calculate the voltage of propellers as V-Front, V-Right, V-Left, and V-Back that are utilized to drive corresponding propellers. Consequently, the attitude angles including the pitch, roll and yaw angles can be controlled.

2.2 Problem Formulation

As a result, the dynamics of the quadrotor can be considered as a nonlinear MIMO discrete system as

$$\begin{aligned}\boldsymbol{y}(k+1)=\boldsymbol{f}(&\boldsymbol{u}(k),\boldsymbol{u}(k-1),...,\boldsymbol{u}(k-n_u),\\ &\boldsymbol{y}(k),\boldsymbol{y}(k-1),...,\boldsymbol{y}(k-n_y)),\end{aligned} \tag{1}$$

where $\boldsymbol{u}(k)=[u_1(k),u_2(k),u_3(k),u_4(k)]^T$ denotes the control input vector, and $u_1(k), u_2(k), u_3(k)$, and $u_4(k)$ denote the rotating speed of the rotor in front, left, right, and back propellers, respectively. $\boldsymbol{f}(k)\in R^3$ is an unknown nonlinear function and k denotes the sampling time. $\boldsymbol{y}(k)=[y_1(k),y_2(k),y_3(k)]^T$ denotes the control output vector, where $y_1(k), y_2(k), y_3(k)$ denote the pitch angle, roll angle and the yaw angle, respectively.

The quadrotor aircraft is a typically complex multivariable system, and the outputs are coupled with each other. In addition, the system has strong nonlinearity, which means that the conventional linear model is difficult to describe the behaviors of the system for a wide operating range. Consequently, designing a controller for this nonlinear MIMO system without the accurate system model in a data-driven manner is of significance and necessary.

3 CONTROLLER DESIGN OF I-PID-MFAC FOR QUADROTOR SYSTEM

The quadrotor dynamics (1) is a system with two integrity components and is open-loop unstable. For this reason, intelligent PID (i-PID) controllers are introduced [7, 14] to stabilize the system first.

For a general nonlinear system, the essential idea of i-PID can be concluded into two parts. First, a proposed 'ultra-local model' is established to replace the general nonlinear plant. Second, the controller is designed based on this model and the controller parameters are adjusted at each time instant.

The i-PID control method has been already successfully applied to many practical examples. Compared with classical PID method, i-PID controllers can ensure good performances with disparate plants without having to tune the PID parameters again and again. Moreover, i-PID controllers can guarantee a suitable adaptation when the plant is time-varying.

3.1 Ultra-Local Model

The system (1) is first replaced by three ultra-local models special local models for three outputs, respectively, described as

$$\begin{cases} y_1^{n_1} = F_1 + \alpha_1 \cdot u_1^1, \\ y_2^{n_2} = F_2 + \alpha_2 \cdot u_2^1, \\ y_3^{n_3} = F_3 + \alpha_3 \cdot u_3^1, \end{cases} \tag{2}$$

where $n_i, i=1,2,3$ are the derivation orders, which are set as 1 in most of practical cases [15]. α_i, $i=1,2,3$, are non-physical constant parameters, which are chosen by the practitioner such that $\alpha_i \cdot u_i^1$ and $y_i^{n_i}$ are with the same magnitude. Notably, their values are obtained by trial and error. Moreover, $F_i, i=1,2,3$ are continuously updated, containing partly known parts of the plant as well as various of possible disturbances that are unnecessary to distinguish [16]. Denote $\boldsymbol{u}^1$ as $[u_1^1, u_2^1, u_3^1]^T$. Let $n_i = 1, i=1,2,3$, then the discrete version of (2) are described as

$$\begin{cases} y_1(k+1) = y_1(k) + F_1(k) + \alpha_1 \cdot u_1^1(k), \\ y_2(k+1) = y_2(k) + F_2(k) + \alpha_2 \cdot u_2^1(k), \\ y_3(k+1) = y_3(k) + F_3(k) + \alpha_3 \cdot u_3^1(k), \end{cases} \tag{3}$$

Remark 3.1: $u_i^1, i=1,2,3$ in $\boldsymbol{u}^1(k)$ are different from $u_1(k), u_2(k), u_3(k),$ and $u_4(k)$ in $\boldsymbol{u}(k)$. After $\boldsymbol{u}^1(k)$ is obtained, it must be decoupled as four inputs in $\boldsymbol{u}(k)$, which can be conducted as follows.

$$\begin{cases} u_1(k) = \dfrac{1}{2} u_1^1(k) \\ u_2(k) = -\dfrac{1}{2}(u_1^1(k) - u_2^1(k)) \\ u_3(k) = -\dfrac{1}{2}(u_1^1(k) + u_2^1(k)) \\ u_4(k) = u_2^1(k) \end{cases} . \tag{4}$$

3.2 Intelligent Controllers

The i-PID controllers for (3) are defined as

$$u_i^1(k) = -\frac{F_i(k) - \Delta y_i^*(k+1)}{\|\alpha_i\|} + K_p^i e_i(k) + K_I^i \sum_{t=1}^{k} e_i(t),\ i=1,2,3 \quad , \tag{5}$$

where y_i^* is the desired output of each y_i. $e_i(k) = y_i^*(k) - y_i(k)$ is the tracking error. K_P^i, K_I^i are gains.

According to the algebraic parameter identification method, $F_i(k)$ can be approximated via the operational calculus [17, 18]. Here, we only give the brief structure of the estimation of $F_i(k)$, and more details can be seen in reference [7].

First, (2) is rewritten in the operational domain as

$$sY_i = \frac{F_i}{s} + U_i + y_i(0), i=1,2,3, \tag{6}$$

where F_i is a constant, and U_i, Y_i are the input and output defined at the operational domain. Then the initial condition $y_i(0)$ is cancelled by multiplying both sides on the left by d/ds as

$$Y_i + s\frac{dY_i}{ds} = -\frac{F_i}{s^2} + \alpha\frac{dU_i}{ds}, i=1,2,3 , \tag{7}$$

After multiplying both sides by s^{-2}, the real time estimation of $\hat{F}_i(k)$ in time domain is inferred as

$$\hat{F}_i(t) = -\frac{6}{\tau^3}\int_{t-\tau}^{t}[(\tau-2\sigma)y_i(\sigma) + \alpha_i\sigma(\tau-\sigma)u_i^*(\sigma)]\cdot d\sigma, \quad , i=1,2,3 \tag{8}$$

where $\tau > 0$ is a small constant. In order to implement (8) in computer conveniently, the discrete time version of (8) is given as

$$\hat{F}_i(k) = -\frac{6}{\tau^3}\sum_{\sigma=k-\tau}^{k}((\tau-2\sigma)y_i(\sigma) + \alpha_i\sigma(\tau-\sigma)u_i^*(\sigma)) \quad , i=1,2,3 . \tag{9}$$

Using (5) and (9), $u_i^1(k), i=1,2,3$ can be obtained.

3.3 Model Free Adaptive Control for Quadrotor

After the application of i-PID controllers, the closed loop system of the quadrotor becomes stable. However, the nonlinearity of the system and coupling between the system variables are not canceled completely. Since MFAC has well adaptive property, it is then adopted as the main loop controller.

The whole control scheme is listed in Fig. 2. After stabilized via i-PID controllers, the closed-loop system can be regarded as a new plant, denoted as $\boldsymbol{g}(\cdot)$. The MFAC controller is then utilized to control the plant $\boldsymbol{g}(\cdot)$, whose input and output at time instant k are denoted as $\boldsymbol{u}^*(k)$ and $\boldsymbol{y}(k)$, respectively.

The plant $\boldsymbol{g}(\cdot)$ is described as

$$\boldsymbol{y}(k+1) = \boldsymbol{g}(\boldsymbol{u}^*(k), \boldsymbol{u}^*(k-1), ..., \boldsymbol{u}^*(k-n_u), \boldsymbol{y}(k), \boldsymbol{y}(k-1), ..., \boldsymbol{y}(k-n_y)) \quad , \tag{10}$$

where $\boldsymbol{g}(k) \in R^3$ is an unknown nonlinear vector-valued function. In practice, two assumptions are reasonable as follows.

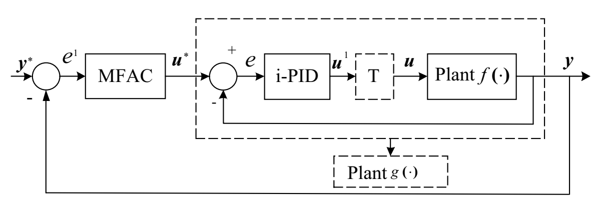


Fig. 2 Control flow chart of i-PID MFAC. T denotes the transform module, which is utilized to transform $\boldsymbol{u}^1$ to $\boldsymbol{u}$.

Assumption 3.1: The partial derivatives of $\boldsymbol{g}(\cdot)$ with respect to control inputs $\boldsymbol{u}^*(k)$ are continuous.

Assumption 3.2: The system is generalized *Lipschitz*, i.e. $\|\Delta\boldsymbol{y}(k+1)\| \le b\cdot\|\Delta\boldsymbol{u}^*(k)\|$ for each k and $\|\Delta\boldsymbol{u}^*(k)\| \ne 0$, where $\Delta\boldsymbol{y}(k+1)=\boldsymbol{y}(k+1)-\boldsymbol{y}(k)$, and b is a positive constant. $\Delta\boldsymbol{u}^*(k)=\boldsymbol{u}^*(k)-\boldsymbol{u}^*(k-1)$.

Remark 3.2: These two assumptions imposed on the controlled system are reasonable and acceptable from a practical viewpoint. Assumption 3.1 is a typical condition of control system design for general nonlinear systems. Assumption 3.2 limits the rates of changes of the system outputs driven by the changes of the control inputs. From the 'energy' point of view, the rates of the output energy change inside a system cannot go to infinity if the changes of the control input energy are in a finite altitude.

Theorem 3.1: For the system (10) satisfying Assumptions 1 and 2 with $\|\Delta\boldsymbol{u}^*(k)\| \ne 0$ for any k, there must exist $\boldsymbol{\varphi}(k)$, named Pseudo Jacobi matrix (PJM), such that (10) can be transformed into following equivalent compact form dynamic linearization model (DLM) as

$$\Delta\boldsymbol{y}(k+1)=\boldsymbol{\varphi}(k)\cdot\Delta\boldsymbol{u}^*(k) \quad (11)$$

where and $\boldsymbol{\varphi}(k)$ is a matrix with components φ_{ij}, i=1,2,…,p, j=1,2,…,p, and $\|\boldsymbol{\varphi}(k)\| \le b$.

Proof: See reference [19].

Remark 3.3: According to Theorem 3.1, for each fixed k, there exists time-varying PJM $\boldsymbol{\varphi}(k)$, such that the nonlinear system can be transformed into CFDL data model. The existence of PPD matrix is guaranteed by rigorous mathematical analysis, thus it is an accurate and equivalent linearization description [19].

3.4 Differences between i-PID and MFAC

MFAC has some differences compared with i-PID method proposed in section 3.1. First, both methods have corresponding local models to describe the nonlinear system. However, the difference lies in that the nonlinearity and the coupling of the MIMO system are contained in F_1-F_3 of the ultra local models in i-PID, whereas the same content is included in $\boldsymbol{\varphi}(k)$ in the MFAC.

Moreover, compared with F_1-F_3, $\boldsymbol{\varphi}(k)$ is related to the system inputs and outputs till time instant k, which is a differential signal in some sense and bounded for any k. Thus, $\boldsymbol{\varphi}(k)$ can be regarded as a slowly time varying parameter, and the relationship with respect to the control input $\boldsymbol{u}^*$ can be ignored if $\|\Delta\boldsymbol{u}^*(k)\| \ne 0$. Besides, $\boldsymbol{\varphi}(k)$ not only can describe nonlinearity and the inner connections of the system more detailedly, but also can eliminate the decoupling procedure existing in model-based control methods.

Moreover, the i-PID controller is original proposed for single-input ingle-output (SISO) systems. For MIMO systems, although i-PID can be implemented, it is lack of rigorous theoretical guidance for application. Compared with i-PID, MFAC not only has systematic design scheme for both SISO and MIMO systems, e.g. CFDL-MFAC, PFDL-MFAC, FFDL-MFAC [9]), but also has rigorous proof for the stability of closed-loop systems controlled via MFAC controller, e.g. CFDL-MFAC, PFDL-MFAC.

Further, when the described plant is a SISO system and the order n_i in the ultra-local model is set as 1, the ultra-local model and the DLM can be simplified as

$$\Delta y(k+1)=F(k)+\alpha\cdot u(k), \quad (12)$$

$$\Delta y(k+1)=-\phi(k)\cdot u(k-1)+\phi(k)\cdot u(k), \quad (13)$$

where $\phi(k)$ is called pseudo-partial-derivative (PPD). In this situation, it can be observed that the effect of $F(k)$ and α in (12) is similar to $\phi(k)$ in (13).

3.5 Model Free Adaptive Controller Design

Consider the following cost function of the control input

$$J(\boldsymbol{u}^*(k))=\|\boldsymbol{y}^*(k+1)-\boldsymbol{y}(k+1)\|^2+\lambda\|\boldsymbol{u}^*(k)-\boldsymbol{u}^*(k-1)\|^2, \quad (14)$$

where the weighting factor $\lambda>0$ is introduced to restrain the control input changes. Let $\partial J(\boldsymbol{u}^*(k))/\boldsymbol{u}^*(k)=0$, it has

$$\begin{aligned}\boldsymbol{u}^*(k)&=\boldsymbol{u}^*(k-1)\\&+(\lambda I+\boldsymbol{\varphi}^T(k)\boldsymbol{\varphi}(k))^{-1}\boldsymbol{\varphi}^T(k)(\boldsymbol{y}^*(k+1)-\boldsymbol{y}_m(k))\end{aligned} \quad (15)$$

Notably, to avoid calculating the matrix inverse existing in (15), it is simplified as

$$\boldsymbol{u}^*(k)=\boldsymbol{u}^*(k-1)+\frac{\rho\boldsymbol{\varphi}^T(k)(\boldsymbol{y}^*(k+1)-\boldsymbol{y}(k))}{\lambda+\|\boldsymbol{\varphi}(k)\|^2}, \quad (16)$$

where the step vector $\rho\in(0,1]$ is introduced to make the controller algorithm become more general. Moreover, $\boldsymbol{\varphi}(k)$ is unknown in (16), and it can be estimated by considering the following function

$$\begin{aligned}J(\boldsymbol{\varphi}(k))&=\|\Delta y(k)-\boldsymbol{\varphi}(k)\Delta\boldsymbol{u}^*(k-1)\|\\&+\mu\|\boldsymbol{\varphi}(k)-\hat{\boldsymbol{\varphi}}(k-1)\|\end{aligned}, \quad (17)$$

where the weighting factor $\mu>0$ is utilized to restrain the change of the PJM estimation. By minimizing (17), it has

$$\begin{aligned}\hat{\boldsymbol{\varphi}}(k)&=\hat{\boldsymbol{\varphi}}(k-1)\\&+(\Delta\boldsymbol{y}(k-1)-\hat{\boldsymbol{\varphi}}(k-1)\Delta\boldsymbol{u}^*(k-1)\Delta(\boldsymbol{u}^*)^T(k-1))\\&\cdot(\mu I+\Delta\boldsymbol{u}^*(k-1)\cdot\Delta(\boldsymbol{u}^*)^T(k-1))^{-1}\end{aligned}, \quad (18)$$

where $\hat{\boldsymbol{\varphi}}(k)$ is the estimation of $\boldsymbol{\varphi}(k)$. Since (18) also contains matrix inversion computation, the following simplified PJM estimation algorithm is utilized as

$$\begin{aligned}\hat{\boldsymbol{\varphi}}(k)&=\hat{\boldsymbol{\varphi}}(k-1)\\&+\frac{\eta(\Delta\boldsymbol{y}(k-1)-\hat{\boldsymbol{\varphi}}(k-1)\Delta\boldsymbol{u}^*(k-1)\Delta\boldsymbol{u}^*(k-1))}{\mu+\|\Delta\boldsymbol{u}^*(k-1)\|^2}\end{aligned}. \quad (19)$$

where $\eta\in(0,2]$ is a weighting factor, Moreover, the reset mechanism is needed as follows.

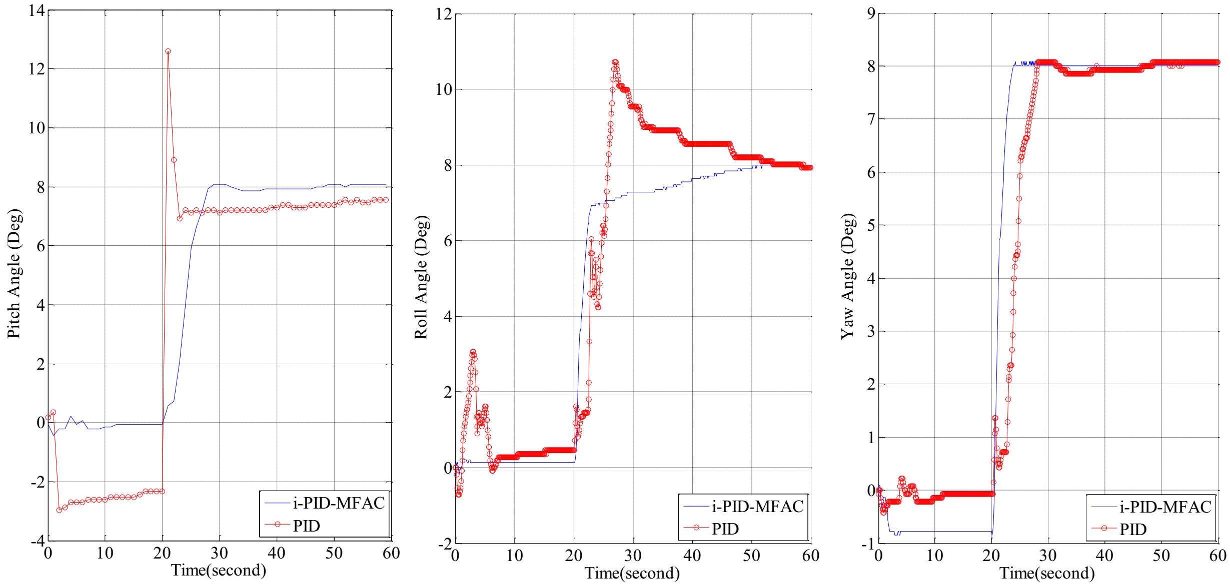


Fig. 3 Real-time response of attitude angles for step signal using two data-driven methods. (a). Pitch angle. (b). Roll angle. (c). Yaw angle

$$\begin{aligned}&\hat{\varphi}_{ii}(k)=\hat{\varphi}_{ii}(1),\ if\ \hat{\varphi}_{ii}(k)<b_2\ or\ \hat{\varphi}_{ii}(k)>\alpha b_2\ or\\ &sign(\hat{\varphi}_{ii}(k))\neq\hat{\varphi}_{ii}(1),\ i=1,2,3\end{aligned}\ , \qquad (20)$$

$$\begin{aligned}&\hat{\varphi}_{ij}(k)=\hat{\varphi}_{ij}(1),\ if\ \hat{\varphi}_{ij}(k)>b_1\ or\ \ sign(\hat{\varphi}_{ij}(k))\neq\hat{\varphi}_{ij}(1),\\ &i,\ j=1,2,3,\ i\neq j\end{aligned}\ . \qquad (21)$$

Remark 3.4: The effect of the reset mechanism (20)-(21) is to endow the parameter estimation algorithm (15)-(19) with a strong ability to track the time-varying parameter. Notably, this reset mechanism is not activated frequently in most practical cases. Only when the control process becomes abnormal, e.g. a sudden big disturbance or behavior for the system, this mechanism will be triggered as a last resort.

The overall control scheme has been mentioned in Fig. 2. In the control process, the inner loop adopts i-PID controllers, which can adjust the plant until it become stable. Then MFAC is adopted as the main-loop of the control system, whose outputs are served as the input of the inner loop.

4 EXPERIMENTS

In order to verify the effectiveness of proposed method, an experiment is implemented on a practical quadrotor in this section. The introduction of the platform has been given in section 2.1.

For i-PID-MFAC method, three i-PI controllers utilized for three outputs in the minor loop. For the comparison purpose, conventional PID method is applied here, and the parameters settings of these two methods are shown in Table 1.

Before the start of the quadrotor aircraft, a reasonable original position must be set firstly. Here, the original pitch angle, roll angle, and yaw angle are set as 0, respectively. The simulation time is 60s. At 20th second, the desire signals are set as 8 degs for three outputs, respectively. The response curves for three outputs obtained using conventional PID and i-PID-MFAC are shown in Fig. 3.

Table 1. Parameters of Two Data-Driven Methods

PID			
	K_p	K_I	K_d
y_1	0.4	0.01	0.7
y_2	0.08	0.01	0.3
y_3	0.5	0.001	0.7
i-PID-MFAC			
$\hat{\boldsymbol{\varphi}}(1)=diag(1,1,1)$	$\hat{\boldsymbol{\varphi}}(2)=\hat{\boldsymbol{\varphi}}(1)$		
$K_p^1=0.55,\ K_I^1=0.03$	$\varepsilon=10^{-4}$		
$K_p^2=0.1,\ K_I^2=0.37$	$\mu=200$		
$K_p^3=0.78,\ K_I^3=0.5$	$\eta=1$		
$\rho=1,\ \lambda=3$	$\alpha_1\sim\alpha_3=1.5$		

From Fig. 3, it can be observed that the attitude angles curves fluctuate slightly at first 10 seconds. The reason is that the system cannot immediately enter the equilibrium state when the system gets started, and there must exist a short-term adjusting process. Meanwhile, it can be seen from three sub-figures that the fluctuation by using PID method is larger than that of proposed method.

After 20s, i-PID-MFAC can quickly response the step signals for three angles with small steady-state errors. In contrast, although conventional PID can also complete the task finally, it has longer oscillation and larger steady-state errors for all outputs than those of i-PID-MFAC. Consequently, the control performance of i-PID-MFAC is more satisfactory than that of conventional PID.

5 CONCLUSION

In this work, a novel data-driven control scheme is proposed to adjust the attitude of the MIMO quadrotor flight simulator. The proposed scheme combines a novel MIMO model free adaptive control (MFAC) method with an novel intelligent PID (i-PID) control algorithm. The design process is model free, which avoids the complex modelling process of the quadrotor simulator and naturally decouples the MIMO simulator system via a novel concept termed Pseudo Jacobi matrix (PJM) only using the I/O data of the system. Experimental results verify the effectiveness of the proposed method.

REFERENCES

[1] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004, pp. 2451-2456.

[2] Z. Zuo, "Trajectory tracking control design with command-filtered compensation for a quadrotor," IET Control Theory & Applications, vol. 4, no. 11, pp. 2343-2355, Nov. 2010.

[3] A. Das, F. Lewis, and K. Subbarao, "Backstepping approach for controlling a quadrotor using lagrange form dynamics," J. Intell. Rob. Syst. , vol. 56, no. 1-2, pp. 127-151, Sep. 2009.

[4] K. Alexis, G. Nikolakopoulos, and A. Tzes, "Model predictive quadrotor control: attitude, altitude and position experimental studies," IET Control Theory & Applications, vol. 6, no. 12, pp. 1812-1827, Aug. 2012.

[5] K. Ang, G. Chong, and Y. Li, "PID control system analysis, design, and technology," IEEE Trans. Contr. Syst. Technol., vol. 13, no. 4, pp. 559-576, Jul. 2005.

[6] M. Ö. Efe, "Neural network assisted computationally simple PI D control of a quadrotor UAV," IEEE Transactions on Industrial Informatics, vol. 7, no. 2, pp. 354-361, May. 2011.

[7] M. Fliess, and C. Join, "Model-free control," Int. J. Control, vol. 86, no. 12, pp. 2228-2252, Jul. 2013.

[8] Z. Hou, and W. Huang, "The model-free learning adaptive control of a class of SISO nonlinear systems," in Proc. IEEE Amer. Control Conf., 1997, pp. 343-344.

[9] Z. Hou, and S. Jin, Model Free Adaptive Control: Theory and Applications. CRC Press, 2013.

[10] K. Tan, T. Lee, S. Huang, and F. Leu, "Adaptive-predictive control of a class of SISO nonlinear systems," Dyn. Control, vol. 11, no. 2, pp. 151-174, Apr. 2001.

[11] Y. Chang, B. Gao, and K. Gu, "A model-free adaptive control to a blood pump based on heart rate," ASAIO Journal, vol. 57, no. 4, pp. 262-267, Feb. 2011.

[12] X. Lu, W. Li, and Y. Lin, "Load Control of Wind Turbine Based on Model-free Adaptive Controller," Transactions of the Chinese Society for Agricultural Machinery, vol. 42, no. 2, pp. 109-115, Feb. 2011.

[13] C. Lu, Y. Zhao, K. Men, L. Tu, et al., "Wide-area power system stabiliser based on model-free adaptive control," IET Control Theory & Applications, vol. 9, no. 13, pp. 1996-2007, Aug. 2015.

[14] M. Fliess, and C. Join, "Model-free control and intelligent PID controllers: towards a possible trivialization of nonlinear control?," arXiv preprint arXiv:0904.0322, 2009.

[15] C. Join, F. Chaxel, and M. Fliess, "" Intelligent" controllers on cheap and small programmable devices," arXiv preprint arXiv:1307.4841, 2013.

[16] M. Fliess, and C. Join, "Intelligent PID controllers," in Proc. 16th Mediterranean Conference on Control and Automation, , 2008, pp. 326-331.

[17] J. Mikusinski, Operational calculus, vol. 1. Elsevier, 2014.

[18] M. Fliess, and H. Sira-Ramirez, "An algebraic framework for linear identification," ESAIM controle optimisation et calcul des variations, vol. 9, pp. 151,2004.

[19] Z. Hou, and S. Jin, "Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems," IEEE Trans. Neural. Networ., vol. 22, no. 12, pp. 2173-2188, Dec. 2011.