

Research Article Application of Online Iterative Learning Tracking Control for Quadrotor UAVs

Pong-in Pipatpaibul and P. R. Ouyang

Department of Aerospace Engineering, Ryerson University, Toronto, ON, Canada

Correspondence should be addressed to P. R. Ouyang; pouyang@ryerson.ca

Received 21 April 2013; Accepted 20 May 2013

Academic Editors: Z. Bi, J.-S. Liu, R. Safaric, and Y. Zhou

Copyright © 2013 P.-i. Pipatpaibul and P. R. Ouyang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Quadrotor unmanned aerial vehicles (UAVs) have attracted considerable interest for various applications including search and rescue, environmental monitoring, and surveillance because of their agilities and small sizes. This paper proposes trajectory tracking control of UAVs utilizing online iterative learning control (ILC) methods that are known to be powerful for tasks performed repeatedly. PD online ILC and switching gain PD online ILC are used to perform a variety of manoeuvring such as take-off, smooth translation, and various circular trajectory motions in two and three dimensions. Simulation results prove the ability and effectiveness of the online ILCs to perform successfully certain missions in the presence of disturbances and uncertainties. It also demonstrates that the switching gain PD ILC is much effective than the PD online ILC in terms of fast convergence rates and smaller tracking errors.

1. Introduction

Unmanned aerial vehicles (UAVs) have become very popular among researchers and developers in the last decade, owing to their capabilities of various applications such as meteorological surveillance, disaster monitoring, and military purposes. Depending on their applications, UAVs vary in sizes, shapes, and operating ranges. UAVs are complicated for control considering the nonlinearity of the system, no supervision of pilots, and external disturbances that need sophisticated control system to deal with.

A quadrotor UAV is a special UAV that has four rotors with a symmetric shape to generate thrust. A quadrotor UAV can vertically take-off, hover, swiftly manoeuvre in any direction and carry a large payload comparing to its own weight . In addition, a quadrotor UAV is normally small compared with other types of UAVs, which makes it simple and cheap, and accessible indoors or in urban constrained areas.

In order to achieve autonomous control, many studies and experiments have been performed for UAVs. However, due to their nonlinearity, multi-input and coupling characteristics, traditional control methods such as PID control may perform poorly under uncertainty and wind disturbances for

UAVs, as shown in [1–3]. As an optimization technique with feedback control, Linear Quadratic Regulator (LQR) and its variation of State Dependent Riccati Equation (SDRE) control were proved to perform well for UAVs without disturbances [4, 5]. Comparisons and implementation of Sliding Mode Control (SMC) and Backstepping control were presented in [6-8]. The SMC methods were found to be robust against uncertainties, but they are still relatively complicated and it might cause chattering problems. An H-infinity control [9] was proposed to deal with the problem of stabilization of a rotorcraft with small external disturbances. An adaptivefuzzy control was developed in [10] with robustness but limited to stabilization applications only. Neural Network control methods were implemented in [11, 12] that proved to perform well, but intensive computations are required for training NN that may limit their real applications. A testbed for the development and demonstration of control systems and trajectory planning optimization of UAVs was set up and discussed in MIT [13, 14].

On the other hand, iterative learning control (ILC) [16] is based on simple PD/PID control and adopts the idea of human learning process, and it is mainly used in robotic manipulators where their tasks are performed repeatedly. ILC



FIGURE 1: An example of a quadrotor UAV [15].



FIGURE 2: Coordinate system of UAV.

can improve tracking performances when a specific task is done again and again, and it is robust against uncertainties and disturbances. ILC was originally proposed by Arimoto [16], where only derivative part (D-type) and information from previous iteration(s) were used to determine the control input for current iteration (offline ILC). Some variations of ILC such as P-type [17], PD-type [18, 19], PI-type [20, 21], and PID-type [22, 23] were studied, but most of them cannot assure monotonic convergence for nonlinear systems. Faster convergence rates can be achieved by using current iteration errors (online ILC) in [24-27] or a combination of both online and offline ILCs in [28–31]. An even faster monotonic convergence rate and less tracking error can be achieved by using switching gain learning control developed in [32, 33]. Performance comparison of various types of ILCs was conducted in [34].

Many control methods have been developed to deal with the control problems of UAVs. The majority of those controllers focused heavily on the stabilization problem which is the first step toward successfully autonomous flights. Some of them also handled position maintaining or velocity holding in order to fulfill certain manoeuvrings and thus full autonomous control. Some researches focused on the trajectory tracking control of UAVs through developing different control methods [35-37]. Based on gain-scheduling control theory, the nonlinear problem of UAV dynamics was represented as piece-wise error dynamics over a predefined set of operating regions in [35], and the design and performance evaluation of a trajectory tracking controller were discussed. LQR control was applied as a trajectory follower to minimize errors between the real trajectory and the reference trajectory in [36]. A switching control method was introduced for trajectory tracking of fixed-wing UAVs in [37].

To the best of our knowledge, there is rare application of ILC on the control of UAVs. The contribution of this work to the UAV field consists in the exploration of different online ILCs on trajectory tracking control of quadrotor UAVs for the first time, focusing on performance improvements under conditions of uncertainty and disturbances. The capability of online ILCs to reject random disturbances and noise is examined and demonstrated. The research can be viewed as a new application of online ILCs. In this research, two

competitive types of ILCs, namely, PD online ILC and switching gain PD (SPD online ILC), are employed in order to control a quadrotor UAV to follow various desired trajectories in the presence of disturbances and uncertainties. Nonlinear dynamic model of the quadrotor UAV is established where major aerodynamic and gyroscopic effects are included. Simulations are conducted to verify the capability of performing various missions repeatedly with some constraints.

2. Quadrotor UAV: Dynamic Model and Parameters

As named, a quadrotor UAV consists of four rotors producing thrust upward against its own weight and payloads. Usually, payloads are placed at the center of the UAV, and rotors propelled by DC motors with or without gearbox. Figure 1 shows an example of a quadrotor UAV and Figure 2 illustrates the coordinate system and positions of rotors used in this paper. The rotors are divided into two pairs; one pair rotates in the opposite direction of the other pair. Increasing the thrust on one side and decreasing the thrust on the other side of the same pair will result in rotation in the pitch or roll direction and the quadrotor will tend to translate toward the direction that it inclines to. Simply increasing thrust equally in one pair and decreasing thrust in the other pair will result in the rotation of the yaw angle while maintaining the position and altitude.

According to the coordinate system shown in Figure 2, a nonlinear translational and rotational dynamic model for the quadrotor UAV can be expressed as [12]

$$\tau = \mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \left(\mathbf{J}\boldsymbol{\omega} + \mathbf{J}_r \boldsymbol{\Omega}_r \mathbf{e}_3 \right) + \mathbf{d}_\tau,$$

$$m\dot{\mathbf{v}} = \mathbf{R}\mathbf{T} - ma\mathbf{e}_2 - \mathbf{d}_\tau,$$
(1)

where $\boldsymbol{\tau}$ is a torque vector of the three axes; **J** is the inertia tensor of a quadrotor; $\boldsymbol{\omega}$ is an angular velocity vector; \mathbf{J}_r is the moment of inertia of rotor; $\boldsymbol{\Omega}_r = \boldsymbol{\Omega}_1 - \boldsymbol{\Omega}_2 + \boldsymbol{\Omega}_3 - \boldsymbol{\Omega}_4$ is the total rotor speed; $\mathbf{e}_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ is a unit vector in the inertial frame; **R** is the rotation matrix; **T** is a force vector; \mathbf{d}_{τ} and \mathbf{d}_T are the disturbance vectors for rotations and translations, respectively.

TABLE 1: Parameters of a quadrotor UAV.

| Parameters | Description | Value |
|--------------------|---------------------------------------|---|
| 1 | Quadrotor arm length | 0.232 m |
| Ь | Rotor thrust coefficient | $3.13\times 10^{-5}~N\cdot s^2$ |
| d | Rotor drag coefficient | $7.5\times10^{-7}~m\cdot s^2$ |
| т | Total quadrotor mass | 0.52 kg |
| I_x | Moment of inertia about X axis | $6.228\times 10^{-3}\text{kg}\cdot\text{m}^2$ |
| I_{v} | Moment of inertia about <i>Y</i> axis | $6.225\times10^{-3}\text{kg}\cdot\text{m}^2$ |
| I_z | Moment of inertia about Z axis | $1.121\times 10^{-2}\ \text{kg}\cdot\text{m}^2$ |
| J _r | Moment of inertia of the rotors | $6 \times 10^{-5} \text{kg} \cdot \text{m}^2$ |
| $\Omega_{\rm max}$ | Maximum rotor speed | 279 rad/s |

The equations of motion, along with control inputs, can be described in each axis based on (1) as [12]

$$\ddot{X} = -\sin\theta\cos\phi\left(\frac{b}{m}\right)u_1 - \frac{d_x}{m},\tag{2}$$

$$\ddot{Y} = \sin\phi\left(\frac{b}{m}\right)u_1 - \frac{d_y}{m},\tag{3}$$

$$\ddot{Z} = -g + \cos\theta \cos\phi\left(\frac{b}{m}\right)u_1 - \frac{d_z}{m},\tag{4}$$

$$\ddot{\phi} = \dot{\theta}\dot{\psi}\frac{I_y - I_z}{I_x} - \dot{\theta}\Omega_r\frac{J_r}{I_x} + \frac{lb}{I_x}u_2 - \frac{d_{\emptyset}}{I_x},\tag{5}$$

$$\ddot{\theta} = \dot{\phi}\dot{\psi}\frac{I_z - I_x}{I_y} + \dot{\phi}\Omega_r\frac{J_r}{I_y} + \frac{lb}{I_y}u_3 - \frac{d_\theta}{I_y},\tag{6}$$

$$\ddot{\psi} = \dot{\theta}\dot{\phi}\frac{I_x - I_y}{I_z} + \frac{d}{I_z}u_4 - \frac{d_\psi}{I_z},\tag{7}$$

where l is the span of the quadrotor; b is the thrust coefficient; d is the drag coefficient, which relates to torque in the yaw angle.

The control inputs related to each rotor speed are defined as follows:

$$\mathbf{u} = \begin{cases} u_1 \\ u_2 \\ u_3 \\ u_4 \end{cases} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{cases} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{cases}.$$
 (8)

Note that (2)–(4) are derived using the rotation matrix of R_{ZXY} instead of R_{ZYX} as used by most robotic applications. Design parameters and some limitations of a quadrotor UAV are referred from a real application in [15] and listed in Table 1.

To design a control system for the quadrotor UAV, it is more convenient to establish a state-space model. First, we define a state variable vector as follows:

$$\mathbf{x} = \begin{bmatrix} X & Y & Z & \phi & \theta & \psi & \dot{X} & \dot{Y} & \dot{Z} & \dot{\phi} & \dot{\theta} \end{bmatrix}^{T}.$$
 (9)

Then the state space system for the UAV dynamic model described in (2)-(7) can be expressed as

$$\dot{\mathbf{x}} = \begin{pmatrix} x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \\ -\sin x_5 \cos x_4 \left(\frac{b}{m}\right) u_1 - \frac{d_x}{m} \\ \sin x_4 \left(\frac{b}{m}\right) u_1 - \frac{d_y}{m} \\ -g + \cos x_5 \cos x_4 \left(\frac{b}{m}\right) u_1 - \frac{d_z}{m} \\ a_1 x_{11} x_{12} - a_2 \Omega_r x_{11} - a_{11} u_2 - \frac{d_{\theta}}{I_x} \\ a_3 x_{10} x_{12} + a_4 \Omega_r x_{10} - a_{22} u_3 - \frac{d_{\theta}}{I_y} \\ a_5 x_{10} x_{11} + a_{33} u_4 - \frac{d_{\psi}}{I_z} \end{pmatrix},$$
(10)

where

$$a_{1} = \frac{I_{y} - I_{z}}{I_{x}}, \qquad a_{2} = \frac{J_{r}}{I_{x}}, \qquad a_{3} = \frac{I_{z} - I_{x}}{I_{y}}, \qquad a_{4} = \frac{J_{r}}{I_{y}},$$
$$a_{5} = \frac{I_{x} - I_{y}}{I_{z}}, \qquad a_{11} = \frac{l}{I_{x}}, \qquad a_{22} = \frac{l}{I_{y}}, \qquad a_{33} = \frac{d}{I_{z}}.$$
(11)

In this paper, the disturbances given in (10) have wind disturbance components along the X and Y directions that are modeled by a correlated Gauss-Markov process [9]. In all other directions, some random noises are added to simulate the disturbances and the uncertainties of the dynamics.

3. Different ILCS: Basics and Convergence Conditions

ILC can improve tracking performance of a nonlinear system through iterative operations when a specific task is performed repeatedly. Consider a nonlinear time varying system with the following form:

$$\dot{x}_{k}(t) = f(x_{k}(t), t) + B(t)U_{k}(t) + d_{k},$$

$$y_{k}(t) = C(t)x_{k}(t),$$
(12)

where subscription k denotes the iteration number.

A general form of the control update law for ILCs involving online and offline learning technique can be written



FIGURE 3: Control scheme for PD online ILC and SPD ILC.



FIGURE 4: Effect of K_d on tracking errors for PD online ILCs.

as

$$U_{k+1} = U_k + K_{pon} e_{k+1} (t) + K_{ion} \int_0^t e_{k+1} (t) dt + K_{don} \dot{e}_{k+1} (t) + K_{poff} e_k (t) + K_{ioff} \int_0^t e_k (t) dt + K_{doff} \dot{e}_k (t) ,$$
(13)

where $e(t) = y_d(t) - y(t)$ and $\dot{e}(t) = \dot{y}_d(t) - \dot{y}(t)$. K_p , K_i , and K_d are PID control gains, and subscriptions on and off represent online (feedback, using errors from the current iteration) and offline (feed-forward, using errors from the previous iteration), respectively. Based on the selection of the control



FIGURE 5: Effect of changing the whole set of control gains by a factor of 0.5 and 1.5.

gains, ILCs can be mainly classified as three types, namely, offline, online, and online-offline. In the following sections, the convergence conditions and the convergence rate are discussed.

3.1. Offline ILCs. Traditional ILCs belong to offline learning control, where only the errors from previous iterations have been included in the control law:

$$U_{k+1} = U_k + K_{p\,\text{off}} e_k(t) + K_{i\,\text{off}} \int_0^t e_k(t) \, dt + K_{d\,\text{off}} \dot{e}_k(t) \, .$$
(14)



FIGURE 6: Tracking performance comparison between PD online ILC and SPD ILC for take-off.

In the case that only a proportional part is utilized (P-ILCs), the control law can be expressed as

$$U_{k+1} = U_k + K_{p \text{ off}} e_k(t) \,. \tag{15}$$

For a P-type ILC, the convergence condition for the controlled system (12) is [17]

$$\rho_{p \text{ off}} = \left\| I - K_{p \text{ off}} CB \right\| < 1.$$
(16)

Whenever a derivative part is involved in an ILC, including D-ILC, PD-ILC, and PID-ILC, the convergence condition is [16, 18, 19]

$$\rho_{d\,\text{off}} = \|I - K_{d\,\text{off}} CB\| < 1. \tag{17}$$

The two aforementioned convergence conditions imply that offline learning gains are upper bounded, which means that only a certain range of control gains can be chosen. Therefore, offline ILCs have a relatively slow convergence rate.

3.2. Online ILCs. On the other hand, online ILCs only rely on current iteration errors:

$$U_{k+1} = U_k + K_{pon} e_{k+1}(t) + K_{ion} \int_0^t e_{k+1}(t) dt + K_{don} \dot{e}_{k+1}(t)$$
(18)

and their convergence conditions are [27-29]

$$\rho_{p \text{ on}} = \left\| \left(I + K_{p \text{ on}} CB \right)^{-1} \right\| < 1 \quad (P-\text{type only}),$$

$$\rho_{d \text{ on}} = \left\| \left(I + K_{d \text{ on}} CB \right)^{-1} \right\| < 1 \quad (D-\text{type involved}).$$
(19)

It is noticed that the learning gains are unbounded for online ILCs from (19), which makes it very flexible to choose control gains. Furthermore, it is demonstrated that the larger the online control gains, the faster the system convergence rate.

In addition to ordinary online ILCs, one might utilize switching gain learning control to increase the convergence rate by increasing online learning gains for each iteration. The convergence conditions then become [32]

$$\rho_{p \text{ on}} = \left\| \left(I + K_{p \text{ on}} \left(0 \right) CB \right)^{-1} \right\| < 1 \quad (\text{P-type only}),$$

$$\rho_{d \text{ on}} = \left\| \left(I + K_{d \text{ on}} \left(0 \right) CB \right)^{-1} \right\| < 1 \quad (\text{D-type involved}).$$
(20)

3.3. Online-Offline ILCs. When online and offline ILCs are combined and used simultaneously, it simply turns into online-offline ILCs. Convergence conditions are then a combination of both types and can be expressed as [31]

$$\begin{array}{l}
\rho_{p\,\text{off}} < 1 \\
\rho_{p\,\text{on}} < 1
\end{array} \quad (P-type \text{ only}) \quad \text{or} \\
\rho_{d\,\text{off}} < 1 \\
\rho_{d\,\text{on}} < 1
\end{array} \quad (D-type \text{ involved}).$$
(21)

4. Controller Design for Quadrotor UAV

In this paper, the integral part of PID control in (13) will be neglected as ILC itself has the feature of integration, by adding control actions from previous iterations to the current one. Furthermore, offline and online-offline ILCs are not



FIGURE 7: Tracking performance of PD online ILC in iteration 1, 3, 6, and 9 for take-off.

considered as many researchers have demonstrated a superior performance of online ILCs over offline ILCs. Figure 3 shows a scheme of a control diagram for online ILCs. Only ordinary PD online ILCs and SPD online ILCs will be tested and compared in this research. According to (8), there are four control inputs for the UAV system. Based on the state variables defined in (9), the control variables are defined as x_3 to x_6 , corresponding to Z axis and three rotation angles; their corresponding derivatives are x_9 to x_{12} . The control inputs of the UAV are calculated using the



FIGURE 8: Tracking performance improvement comparison for take-off.

(22)



FIGURE 9: Rotor speed at the 9th iteration controlled by PD online ILC for take-off.

following ILC update law:

$$u_{i,k+1} = u_{i,k} + (K_{p \text{ on}})_{i+2} (x_{i+2,d} - x_{i+2})_{k+1} + (K_{d \text{ on}})_{i+2} (x_{i+8,d} - x_{i+8})_{k+1}$$
for $i = 1, 2, 3, 4$,

where $(K_{pon})_j$ and $(K_{don})_j$ are the corresponding online learning proportional and derivative gains.

If the *Z* direction needs to be controlled, from (22), we can see that $x_{3,d}$ and $x_{9,d}$ are the only desired states determined by the desired trajectories. In order to find properly desired states as a function of time for the rest of state variables, two dummy control inputs for the *X* and *Y* direction motions are defined as

$$u_{x,k+1} = u_{x,k} + (K_{p \text{ on}})_1 (x_{1,d} - x_1)_{k+1} + (K_{d \text{ on}})_1 (x_{7,d} - x_7)_{k+1},$$

$$u_{y,k+1} = u_{y,k} + (K_{p \text{ on}})_2 (x_{2,d} - x_2)_{k+1} + (K_{d \text{ on}})_2 (x_{8,d} - x_8)_{k+1},$$
(23)

where $x_{1,d}$, $x_{2,d}$, $x_{7,d}$, and $x_{8,d}$ are the desired trajectories in the X and Y directions and their derivatives, respectively; $(K_{pon})_j$ and $(K_{don})_j$ are the online learning gains. Based on the defined rotation matrix **R**, the desired angles can be calculated from

$$\phi_{d} = x_{4,d} = a \sin(u_{y}),$$

$$\dot{\phi}_{d} = x_{10,d} = \frac{1}{\sqrt{1 - u_{y}^{2}}} \dot{u}_{y},$$

$$\theta_{d} = x_{5,d} = -a \sin\left(\frac{u_{x}}{\cos(x_{4})}\right),$$

$$\dot{\theta}_{d} = x_{11,d} = -\frac{\dot{u}_{x} + \tan(x_{4}) x_{10} u_{x}}{\cos(x_{4}) \sqrt{1 - (u_{x}/\cos(x_{4}))^{2}}}.$$
(24)



(b) Tracking errors for X T translation using of D online TEC

FIGURE 10: Tracking errors from iteration to iteration for X-Y translation (S = 10 m, T = 5 sec.).

Since a quadrotor UAV is very agile and is able to maneuver in any direction, the yaw angle in this paper is defined as

$$\psi_d = x_{6,d} = \dot{\psi}_d = x_{12,d} = 0.$$
 (25)

According to (19), the convergence conditions for online ILCs are only related to matrices **B** (input matrix) and **C** (output matrix) and are not related to function f(x,t) in (12). This feature provides the robustness of the control law to deal with the mismatch in modeling the system. Also, the choices of control gains in (22)-(23) are flexible because they are unbounded as long as (19) is satisfied. It should be noticed that, in the developed control law (22)-(23), the angular positions and angular speeds should be measured by sensors. A fibre optic gyroscope sensor can be used for this purpose because of its high accuracy, fast response, and low weight.

Note that in the SPD ILC, the online learning gains are defined as functions of iteration number and are chosen as

$$K_{p \text{ on }}(k) = s(k) * K_{p \text{ on }}(0)$$

$$K_{d \text{ on }}(k) = s(k) * K_{d \text{ on }}(0)$$

$$s(k) = k, \text{ for } k \ge 1.$$
(26)

The switching action here occurs in iteration domain rather than in time domain. For traditional switching control, the switching action may cause problems in transient process of the switched system, while this phenomenon does not occur in iteration domain.

5. Simulation Results

In order to obtain a smooth trajectory, that is, continuous in position, velocity, and acceleration, a 5th order polynomial



FIGURE 11: Trajectory tracking performance of PD online ILC in iteration 1, 3, 6, and 9 for X-Y translation (S = 10 m, T = 5 sec.).

function of time is used for each desired trajectory in our simulation studies. With the boundary conditions of f(0) = 0, f(T) = 1, f'(0) = f'(T) = f''(0) = f''(T) = 0, the following polynomial function [38] is used to generate the

desired trajectory:

$$f(t) = 10\left(\frac{t}{T}\right)^3 - 15\left(\frac{t}{T}\right)^4 + 6\left(\frac{t}{T}\right)^5, \quad t \in [0, 1].$$
(27)



FIGURE 12: Trajectory tracking performance of SPD ILC in iteration 1, 3, and 6 for X-Y translation (S = 10 m, T = 5 sec.).

In the simulation studies, two different types of disturbances are introduced to deal with different environmental effects on different direction motions of the quadrotor UAV.

For the *X* and *Y* direction motions, external disturbances caused by wind gusts are described by stochastic functions where a Gauss-Markov process is chosen to simulate the wind

effect to the UAV system

$$d_{x}\left(t_{j}\right) = d_{x}\left(t_{j-1}\right) + A * \operatorname{rand}\left(\right),$$

$$d_{y}\left(t_{j}\right) = d_{y}\left(t_{j-1}\right) + A * \operatorname{rand}\left(\right).$$
(28)



FIGURE 13: Trajectory tracking performance comparison between PD online ILC and SPD ILC for X-Y translation (S = 10 m, T = 5 sec.).



FIGURE 14: Trajectory tracking performance comparison between PD online ILC and SPD ILC for X-Y translation (S = 100 m, T = 15 sec.).

Throughout all the simulations, disturbances in the Z direction and all three angles are presented in (29). Note that the disturbances with random noise are very large compared to the components of the dynamic model

$$D = 10 + 20 \sin (2\pi t) + \text{rand} (),$$

$$d_z = 0.1D,$$

$$d_{\phi} = D,$$

$$d_{\theta} = D,$$

$$d_{\psi} = 0.001D.$$
 (29)

To demonstrate tracking performance and robustness of online ILCs, the following simulation will cover main missions of a quadrotor UAV, that is, take-off, hovering, and moving in various trajectories.



(b) Tracking errors for X-Y translation using SPD online ILC

FIGURE 15: Tracking errors from iteration to iteration for X-Y translation (S = 100 m, T = 15 sec.).

5.1. Take-Off Control. In a take-off mission, the quadrotor UAV is commanded to smoothly take-off and ultimately hover at a desired height $h_d = 10$ m within a time duration of T = 5 sec. The desired trajectory is defined as

$$Z_d(t) = h_d \left[10 \left(\frac{t}{T}\right)^3 - 15 \left(\frac{t}{T}\right)^4 + 6 \left(\frac{t}{T}\right)^5 \right],$$

$$X_d = Y_d = 0, \quad t \in [0, T]$$
(30)

and a set of online learning gains is chosen as

$$K_{p \text{ on}} = K_{\text{on}} \begin{bmatrix} 0.1 & 0.1 & 50 & 5 & 5 & 500 \end{bmatrix},$$

 $K_{d \text{ on}} = 1.5 K_{p \text{ on}},$
(31)

where $K_{on} = 0.5, 1, 1.5$ for different control gains. For most simulations conducted in this research, we set $K_{on} = 1$

unless otherwise stated. The choice of control gains above was based on trial and error, with regard to stability and trajectory tracking performance.

The constant K_{on} is used to demonstrate the effect of increasing-decreasing the whole gain set by a factor. A variation of learning gains and its trajectory tracking performance for take-off are listed in Table 2. It should be mentioned that, in order to get fair comparison results, the disturbances and uncertainties for each simulation in the same iteration are assumed to be the same.

As shown in Table 2, reducing $(K_{p \text{ on}})_1$ and $(K_{p \text{ on}})_2$ by a factor of 10 significantly reduces the convergence rate and yields much larger errors in the first three iterations. Likewise, decreasing $(K_{p \text{ on}})_4$ and $(K_{p \text{ on}})_5$ by a factor of 10 yields similar results. This is due to a slower reaction in attitude control, making the quadrotor UAV away from the desired position



FIGURE 16: Trajectory tracking performance comparison between PD online ILC and SPD ILC for X-Y-Z translation.

| Control gain | k | $\ e_x\ $ (m) | $\ e_{y}\ $ (m) | $\ e_z\ (\mathbf{m})$ | <i> φ (deg)</i> | $\ \theta\ $ (deg) | $\ \psi\ $ (deg) |
|--------------------------|---|--------------------|--------------------|-----------------------|--------------------------------|--------------------|-------------------|
| | 1 | 5.423 <i>e</i> – 1 | 5.423 <i>e</i> – 1 | 2.228e - 2 | 1.244 <i>e</i> 0 | 1.244 <i>e</i> 0 | 6.8584 <i>e</i> 0 |
| [0.1 0.1 50 5 5 500] | 3 | 5.857e - 2 | 5.856e - 2 | 1.157 <i>e</i> – 5 | 4.632e - 1 | 4.636e - 1 | 4.7586e0 |
| $K_{\rm on} = 1$ | 5 | 1.532e - 2 | 1.529e - 2 | 1.151e - 5 | 3.287e - 1 | 3.286e - 1 | 3.0638e0 |
| | 9 | 5.261 <i>e</i> – 3 | 5.258e – 3 | 1.141e - 5 | 2.415e - 1 | 2.415e - 1 | 1.3656e0 |
| | 1 | 7.472 <i>e</i> 0 | 7.471 <i>e</i> 0 | 2.229 <i>e</i> – 2 | 2.331 <i>e</i> 0 | 2.330e0 | 6.8584 <i>e</i> 0 |
| [0.01 0.01 50 5 5 500] | 3 | 2.627 <i>e</i> 0 | 2.627 <i>e</i> 0 | 2.099 <i>e</i> – 5 | 2.101 <i>e</i> 0 | 2.104 <i>e</i> 0 | 4.7586e0 |
| $K_{\rm on} = 1$ | 5 | 1.204e0 | 1.204e0 | 1.693 <i>e</i> – 5 | 1.298e0 | 1.298e0 | 3.0638e0 |
| | 9 | 5.748e - 1 | 5.750e - 1 | 1.244e - 5 | 1.727 <i>e</i> 0 | 1.727e0 | 1.3656e0 |
| | 1 | 5.429e – 1 | 5.429 <i>e</i> – 1 | 2.214 <i>e</i> – 1 | 1.243e0 | 1.243e0 | 6.8584 <i>e</i> 0 |
| [0.1 0.15 5 5 500] | 3 | 5.951 <i>e</i> – 2 | 5.950 <i>e</i> – 2 | 2.414e - 4 | 4.638e - 1 | 4.642e - 1 | 4.7586e0 |
| $K_{\rm on} = 1$ | 5 | 1.569 <i>e</i> – 2 | 1.566 <i>e</i> – 2 | 3.032e - 5 | 3.307e - 1 | 3.307e - 1 | 3.0638e0 |
| | 9 | 5.281 <i>e</i> – 3 | 5.278 <i>e</i> – 3 | 1.784e - 5 | 2.442e - 1 | 2.442e - 1 | 1.3656e0 |
| | 1 | 5.132e0 | 5.152e0 | 2.231 <i>e</i> – 2 | 1.296e1 | 1.288e1 | 6.8584 <i>e</i> 0 |
| [0.1 0.1 50 0.5 0.5 500] | 3 | 5.954e - 1 | 5.955e - 1 | 1.224e - 4 | 5.599e0 | 5.687 <i>e</i> 0 | 4.7586e0 |
| $K_{\rm on} = 1$ | 5 | 1.487e - 1 | 1.453 <i>e</i> – 1 | 2.560 <i>e</i> – 5 | 3.962e0 | 3.879e0 | 3.0638e0 |
| | 9 | 5.173 <i>e</i> – 2 | 5.130 <i>e</i> – 2 | 1.298 <i>e</i> – 5 | 2.955e0 | 2.891 <i>e</i> 0 | 1.3656e0 |
| | 1 | 5.421 <i>e</i> – 1 | 5.422 <i>e</i> – 1 | 2.213e – 2 | 1.247e0 | 1.247e0 | 7.9413e0 |
| [0.1 0.1 50 5 5 50] | 3 | 5.940e - 2 | 5.938e – 2 | 1.085e - 5 | 4.627e - 1 | 4.632e - 1 | 7.6792 <i>e</i> 0 |
| $K_{\rm on} = 1$ | 5 | 1.577 <i>e</i> – 2 | 1.568 <i>e</i> – 2 | 1.062e - 5 | 3.272e - 1 | 3.273 <i>e</i> – 1 | 7.4223e0 |
| | 9 | 5.360 <i>e</i> – 3 | 5.315e – 3 | 1.062e - 5 | 2.421e - 1 | 2.424e - 1 | 6.9237 <i>e</i> 0 |
| | | | | | | | |

TABLE 2: Effect of each control gain on tracking errors.

in the presence of disturbances. Lastly, reducing $(K_{pon})_6$ by a factor of 10 shows a considerable decrease in yaw angle stability and a slower convergence rate. The effect of changes in K_{don} on the performance is illustrated in Figure 4. For the X and Y axes tracking, when $K_{don} = K_{pon}$ the system yields larger errors and tends to produce steady errors. For $K_{d \text{ on}} = 1.5K_{p \text{ on}}$ and $K_{d \text{ on}} = 3K_{p \text{ on}}$, the results are much better in terms of maximum errors and final steady errors. These results strongly show the importance of the derivative gains. For the Z direction, the results are merely identical.



FIGURE 17: Trajectory tracking performance comparison between PD online ILS and SPD ILC for X-Y-Z translation.

TABLE 3: Maximum tracking errors in the Z direction for take-off under LQR control.

| Low $Q(10^6 I)$ | Middle $Q(10^7 I)$ | Large $Q(10^8 I)$ |
|-----------------|--------------------|-------------------|
| 0.0802 | 0.0327 | 0.0105 |

Figure 5 shows how changing the whole set of gains by a factor affects trajectory tracking performances. With $K_{on} = 0.5$, for the X and Y directions, it is obvious that the control gains are not high enough to ensure monotonic convergence and yield large errors. On the other hand, when $K_{on} = 1$ and

 $K_{\text{on}} = 1.5$, the system is monotonically convergent and even faster for the latter case. Again, the effect is hardly detected in the *Z* direction. This is because control in the *Z* direction depends solely on u_1 , while the *X* and *Y* motions also depend on the attitude of the quadrotor UAV.

To compare the performances of ILCs with traditional control such as LQR control, take-off simulations were fulfilled. Table 3 lists the maximum tracking errors under different LQR performance indices where an approach for the design of LQR controller discussed in [4] is used. Comparing the maximum errors in Z direction shown in Table 3 with those listed in Table 2, it is clearly shown that the ILC



FIGURE 18: Trajectory tracking performance comparisons for X-Y plane circular motion.

performed much better than the LQR. Such a result would be expected as the ILCs is a kind of nonlinear learning control while the LQR is a linearized control for a nonlinear system.

After finishing the initial comparison study on the performance improvements by changing online control gains, some take-off simulation studies with wind disturbances and uncertainties are conducted. Figure 6 shows the maximum tracking errors from iteration to iteration controlled by both PD online ILCs and SPD ILCs. In the presence of disturbances, there are small errors for the *X* and *Y* direction motions controlled by the PD online ILC because of the existence of wind disturbance. But SPD ILCs outperform PD online ILCs with their incredibly fast convergence rates, and achieve very good performances in all three translation motions after 3 iterations. It demonstrates the capability of dealing with disturbances in SPD ILC.

Trajectory tracking performance in translations and orientations controlled by PD online ILCs is shown in Figure 7 for different iterations. It can be seen that the quadrotor UAV can smoothly take-off, with some errors in the X and Ydirections that were caused by the wind disturbances. It is also noticed that the PD online ILCs and SPD ILCs can reject disturbances in orientations. Figure 8 shows the tracking performance improvements in Z direction from iteration to iteration for both online ILCs. It clearly shows the significant differences between the PD online ILC and the SPD ILC after the first iteration. Rotor speed in the 9th iteration is depicted in Figure 9 for the PD online ILC, which shows that all four rotors operate in the specified ranges.

5.2. Smooth Translation Tracking. To demonstrate a good trajectory tracking performance for an *X*-*Y* plane translation

motion, desired trajectories are chosen as follows:

$$X_{d}(t) = Y_{d}(t) = S\left[10\left(\frac{t}{T}\right)^{3} - 15\left(\frac{t}{T}\right)^{4} + 6\left(\frac{t}{T}\right)^{5}\right],$$

$$Z_{d}(t) = 0, \quad t \in [0, T].$$
(32)

Two sets of simulations are performed here. The first simulation is set for a short range linear translation motion with S = 10 m and T = 5 sec., and the second one is for a long range translation motion at a higher speed with S = 100 m and T = 15 sec.

For the first set motion, it shows that the PD online ILCs tend to take more iterations, as many as 9, to converge to approximate zero errors in X and Y directions while SPD ILCs take 6 iterations. Trajectory tracking errors from iteration to iteration are shown in Figure 10. It clearly shows that the SPD ILC can ensure less tracking errors and faster convergence rates.

Figures 11 and 12 show the tracking performance improvements from iteration to iteration for both PD online ILC and SPD ILC. The effectiveness of the online ILCs in terms of tracking error reductions and resistance to disturbances is clearly shown.

Note that shown in Figure 13, there are relatively large errors in the *X* and *Y* directions for the first iteration and then they converge to stable error boundaries for the PD ILC, but the SPD ILCs obtained very good tracking performance with very small ($<10^{-2}$ m) tracking errors in all three directions even under the wind gust disturbance conditions.

For the second set test with large motion ranges and fast speeds, the maximum tracking errors from iteration to iteration are depicted in Figure 14 for both PD online ILC



(b) Tracking errors for X-Y circular translation using SPD online ILC

FIGURE 19: Tracking performance comparison for X-Y circular motion.

and SPD ILC. It is shown that the PD online ILCs obtain acceptable tracking performance (the tracking errors are around 1 m for both X and Y directions), while the SPD ILCs still perform much better than the PD online ILCs.

Figure 15 shows the tracking errors in the X and Y directions from iteration to iteration for both PD online ILC and SPD online ILC. Considering the long range motions and relatively higher speed than the previous short range motion case, it is clearly shown that, after a few iterations, the quadrotor can obtain smooth and good motions that follow the desired trajectories, especially for the SPD ILC.

Furthermore, to examine the tracking performance for a three dimensional translation motion based on the ILCs, a simulation of linear motion from point A (0, 0, 0) to point B (100, 100, 10) in 15 seconds is conducted. Figure 16 shows the maximum tracking errors from iteration to iteration controlled by these two different online ILCs. It is clearly shown that the SPD ILCs performed much better than the PD online ILCs in terms of fast convergent rate, small tracking errors, and good capability to compensate the disturbances

and uncertainties. It also shows that the tracking performance in the Z direction was controlled very well for both online ILCs. Figure 17 shows the tracking errors from iteration to iteration for both PD online ILC and SPD online ILC for 3D translation motions. It demonstrates the effectiveness of the online ILCs and the better tacking performance obtained by the SPD online ILC.

5.3. Circular Trajectories Tracking. To prove that the quadrotor UAV can freely maneuver a complex trajectory, a circular trajectory tracking test is performed by using the online ILCs. In this mission, the quadrotor revolves the X-Y plane with a radius R = 20 m for a time duration T = 20 sec., smoothly starting from and ending at the velocity of 0. The desired trajectories are

$$\begin{aligned} X_d(t) &= R\cos\left(2\pi\left[10\left(\frac{t}{T}\right)^3 - 15\left(\frac{t}{T}\right)^4 + 6\left(\frac{t}{T}\right)^5\right] + \pi\right) \\ &+ R, \end{aligned}$$



FIGURE 20: Tracking performance comparisons for 3D circular motion.

$$Y_{d}(t) = R \sin\left(2\pi \left[10\left(\frac{t}{T}\right)^{3} - 15\left(\frac{t}{T}\right)^{4} + 6\left(\frac{t}{T}\right)^{5}\right] + \pi\right),$$

$$Z_{d}(t) = 25, \qquad X_{d} \in [0, 2R], \qquad Y_{d} \in [-R, R]$$

$$t \in [0, T].$$
(33)

Figure 18 shows the performance improvements from iteration to iteration for both PD online ILCs and SPD ILCs. Once again, it demonstrated the effectiveness of the SPD ILCs. Figure 19 shows the tracking errors controlled by PD online ILCs and SPD ILCs for the *X*-*Y* plane circular trajectory, respectively. It can be observed that PD online ILCs can obtain acceptable tracking performances through iterative learning processes, and SPD ILCs converge to approximate zero errors at the 3rd iteration while PD online ILCs still have small bounded tracking errors. Therefore, it can be seen that SPD ILCs perform much better in yaw angle than the PD online ILCs.

By adding the Z direction motion (20 m) accordingly with the X-Y plane circular motion discussed above, a 3D circular tracking case was formed and used to examine the effectiveness of the online ILCs. Figure 20 shows the maximum tracking errors in three axes for the PD online ILCs and the SPD ILCs, while Figure 21 shows the tracking performance improvements from iteration to iteration for both online ILCs. It demonstrates again that the SPD ILCs can obtain much better tracking performances than the PD online ILCs under the existence of disturbances and uncertainties.

6. Conclusion and Discussion

Iterative learning control is simple to build, easy to implement, and robust to uncertainties and disturbances. In order to perform some fundamental missions for a quadrotor UAV, some trajectory tracking control for a UAV was conducted based on online ILCs in this paper. Different trajectory tracking tasks using PD online ILCs and SPD ILCs were simulated and compared. It was proven that tracking performances can be improved by choosing large control gains for the online ILCs. In addition, the choice of control gains is very flexible. Despite nonlinearity of the system and the presence of disturbances, both types of online ILCs demonstrated good tracking capabilities of smooth take-off, straight-line, and circular motions in two and three dimensions and provided good performances after a few iterations. Simulation results demonstrated that SPD ILCs performed much better than PD online ILCs in terms of convergence rate and disturbance rejection.

The online ILCs have been successfully applied to trajectory tracking control of a quadrotor UAV through simulations for different missions. To deal with the real application of online ILCs, it should be mentioned that a PD feedback control is applied to control the quadrotor UAV in the first iteration, as there is no previous experimental learning process. The PD feedback control was proved to provide the stability of the UAV system, even with some big tracking errors. The control input information obtained from the first iteration will be stored and used as a feedforward control for the next iterations. In case the state of the system cannot be measured directly, the online ILCs update law could include



FIGURE 21: Tracking performance improvement for 3D circular motion.

a Kalman filter to estimate the states. With all the aspects shown above, implementation to a real application seems to be practical, and it is a future work.

Acknowledgment

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through a Discovery Grant to the second author.

References

- [1] Y. Wu, Development and implementation of a control system for a quadrotor UAV [Ph.D. thesis], University of Applied Science Ravensburg-Weingarten, 2009.
- [2] G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, "Quadrotor helicopter trajectory tracking control," in AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, Hawaii, USA, August 2008.

- [3] K. Miller, "Path tracking control for quadrotor helicopters," 2008.
- [4] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke, "A prototype of an autonomous controller for a quadrotor UAV," in *European Control Conference*, 2007.
- [5] H. Voos, "Nonlinear state-dependent riccati equation control of a quadrotor UAV," in *Proceedings of IEEE Conference on Control Applications (CCA '06)*, pp. 2547–2552, Munich, Germany, October 2006.
- [6] P. Adigbli, C. Grand, J.-B. Mouret, and S. Doncieux, "Nonlinear attitude and position control of a micro quadrotor using sliding mode and backstepping techniques," in *European Micro Air Vehicle Conference and Flight Competition*, Toulouse, France, 2007.
- [7] S. Bouabdallah and R. Siegwart, "Backstepping and slidingmode techniques applied to an indoor micro Quadrotor," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2247–2252, April 2005.
- [8] G. Hoffmann, D. G. Rajnarayan, S. L. Waslander, D. Dostal, J. S. Jang, and C. J. Tomlin, "The Stanford testbed of autonomous rotorcraft for multi agent control (STARMAC)," in *Proceedings of the 23rd Digital Avionics Systems Conference*, vol. 2, pp. 121–130, October 2004.
- [9] J. Gadewadikar, F. L. Lewis, K. Subbarao, K. Peng, and B. M. Chen, "H-infinity static output-feedback control for rotorcraft," *Journal of Intelligent and Robotic Systems: Theory & Applications*, vol. 54, no. 4, pp. 629–646, 2009.
- [10] C. Coza and C. J. B. Macnab, "A new robust adaptive-fuzzy control method applied to quadrotor helicopter stabilization," in *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS '06)*, pp. 475–479, June 2006.
- [11] T. Dierks and S. Jagannathan, "Output feedback control of a quadrotor UAV using neural networks," *IEEE Transactions on Neural Networks*, vol. 21, no. 1, pp. 50–66, 2010.
- [12] C. Nicol, C. J. B. Macnab, and A. Ramirez-Serrano, "Robust neural network control of a quadrotor helicopter," in *Proceed*ings of IEEE Canadian Conference on Electrical and Computer Engineering (CCECE '08), pp. 1233–1237, May 2008.
- [13] J. How, E. King, and Y. Kuwata, "Flight demonstrations of cooperative control for UAV teams," in *Collection of Technical Papers—AIAA 3rd "Unmanned-Unlimited" Technical Conference, Workshop, and Exhibit*, pp. 505–513, September 2004.
- [14] J. P. How, B. Bethke, A. Frank, D. Dale, and J. Vian, "Real-time indoor autonnomous vehicle test environment," *IEEE Control Systems Magazine*, vol. 28, no. 2, pp. 51–64, 2008.
- [15] S. Bouabdallah and R. Siegwart, "Towards intelligent miniature flying robots," *Springer Tracts in Advanced Robotics*, vol. 25, pp. 429–440, 2006.
- [16] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, pp. 123– 140, 1984.
- [17] K. L. Moore, "An observation about monotonic convergence in discrete-time, P-type iterative learning control," in *Proceedings of IEEE International Symposium on Intelligent Control* (*ISIC '01*), pp. 45–49, Mexico City, Mexico, September 2001.
- [18] C. Chen and J. Hwang, "PD-type iterative learning control for the trajectory tracking of a pneumatic X-Y table with disturbances," *JSME International Journal, Series C*, vol. 49, no. 2, pp. 520–526, 2006.

- [19] K. Park, "An average operator-based PD-type iterative learning control for variable initial state error," *IEEE Transactions on Automatic Control*, vol. 50, no. 6, pp. 865–869, 2005.
- [20] Y. Chen and K. L. Moore, "An optimal design of PD-type iterative learning control with monotonic convergence," in *Proceedings of IEEE International Symposium on Intelligent Control*, pp. 55–60, Vancouver, Canada, October 2002.
- [21] Y. Chen and K. L. Moore, "PI-type iterative learning control revisited," in *Proceedings of the American Control Conference*, pp. 2138–2143, Anchorage, Alaska, USA, May 2002.
- [22] K. Park, Z. Bien, and D. Hwang, "A study on the robustness of a PID-type iterative learning controller againts initial state error," *International Journal of Systems Science*, vol. 30, no. 1, pp. 49–59, 1999.
- [23] A. Madady, "PID type iterative learning control with optimal gains," *International Journal of Control, Automation and Systems*, vol. 6, no. 2, pp. 194–203, 2008.
- [24] C. Chien and J. Liu, "P-type iterative learning controller for robust output tracking of nonlinear time-varying systems," in *Proceedings of the American Control Conference*, pp. 2595–2599, July 1994.
- [25] X. G. Yan, I. M. Chen, and J. Lam, "D-type learning control for nonlinear time-varying systems with unknown initial states and inputs," *Transactions of the Institute of Measurement and Control*, vol. 23, no. 2, pp. 69–82, 2001.
- [26] S. Yu, S. Xiong, and J. Bai, "Design of iterative learning controller combined with feedback control for electrohydraulic servo system," in *Proceedings of the 4th International Conference* on Natural Computation (ICNC '08), vol. 2, pp. 622–626, October 2008.
- [27] P. R. Ouyang, W. J. Zhang, and M. M. Gupta, "PD-type online learning control for systems with state uncertainties and measurement disturbances," *Control and Intelligent Systems*, vol. 35, no. 4, pp. 351–358, 2007.
- [28] "Iterative learning control for uncertain nonlinear discretetime systems using current iteration tracking error," in *Iterative LearnIng Control*, vol. 248, chapter 4, Springer, Berlin, Germany, 1999.
- [29] J. Shou, D. Pi, and W. Wang, "Sufficient conditions for the convergence of open-closed-loop PID-type iterative learning control for nonlinear time-varying systems," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 2557–2562, October 2003.
- [30] S. Yu, J.-H. Wu, and X.-W. Yan, "A PD-type open-closedloop iterative learning control and its convergence for discrete systems," in *Proceedings of the 1st International Conference on Machine Learning and Cybernetics*, pp. 659–662, Beijing, China, November 2002.
- [31] P. R. Ouyang, "PD-PD type learning control for uncertain nonlinear systems," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (DETC '09)*, pp. 699–707, San Diego, Calif, USA, September 2009.
- [32] P. R. Ouyang, B. A. Petz, and F. F. Xi, "Iterative learning control with switching gain feedback for nonlinear systems," *Journal of Computational and Nonlinear Dynamics*, vol. 6, no. 1, Article ID 011020, 7 pages, 2011.
- [33] P. R. Ouyang, W. J. Zhang, and M. M. Gupta, "An adaptive switching learning control method for trajectory tracking of robot manipulators," *Mechatronics*, vol. 16, no. 1, pp. 51–61, 2006.

- [34] P. R. Ouyang and P. Pong-in, "Iterative learning control: a comparison study," in *Proceedings of ASME International Mechanical Engineering Congress & Exposition*, Vancouver, Canada, 2010.
- [35] B. Guerreiro, C. Silvestre, R. Cunha, and D. Antunes, "Trajectory tracking H2 controller for autonomous helicopters: an application to industrial chimney inspection," in *Proceedings of the 17th IFAC Symposium on Automatic Control in Aerospace* (ACA '07), pp. 431–436, June 2007.
- [36] R. Beard, D. Kingston, M. Quigley et al., "Autonomous vehicle technologies for small fixed-wing UAVs," *Journal of Aerospace Computing, Information and Communication*, vol. 2, no. 1, pp. 92–108, 2005.
- [37] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke, "A prototype of an autonomous controller for a quadrotor UAV," in *European Control Conference*, Kos, Greece, July 2007.
- [38] E. Dombre and W. Khalil, "Trajectory generation," in *Modeling*, *Performance Analysis and Control of Robot Manipulators*, p. 191, ISTE Ltd., 2007.

