Quadrotor System Design for a 3 DOF platform based on Iterative Learning Control

Husam A. Foudeh[1] Patrick Luk[2] and James F. Whidborne[3]

Abstract— **Research into autonomous control and behaviour of mobile vehicles has become more and more widespread. Unmanned aerial vehicles (UAVs) have seen an upsurge of interest and of the many UAVs available, the quadrotor has shown significant potential in monitoring and surveillance tasks. This paper examines the performance of iterative learning control (ILC) in gradient-based control that enhances a quadrotor's controllability and stability during attitude control. It describes the development of the learning algorithms which exploit the repeated nature of the fault-finding task. Iterative learning control algorithms are derived and implemented on a quadrotor in a test bench. The proposed ILC algorithms on the quadrotor model are evaluated for system stability, convergence speed, and trajectory tracking error. Finally, the performance of the proposed algorithms is compared against a baseline performance of the PID control schemes.**

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) are being increasingly applied to civilian and commercial operations, rather than predominantly in military applications as in the past. The Unmanned Vehicles can fly and perform a multitude of tasks without pilot involvement, and may be designed to operate on the ground or under water. Figure. 1 shows the most common commercially available quadrotors as well as the Hummingbird quadrotor which is used in this research. Whilst primary fields of commercial and research interests include surveillance, monitoring and object tracking systems, autonomous UAVs have been used to move goods in industry and deliver products and services directly to the customer's home. These vehicles have also shown significant potential in applications including disaster monitoring (e.g. recent natural disasters of Indonesia, and the earthquake in Nepal), 3D mapping, and aerial photography. Recently, an American presidential candidate proposed using UAVs for monitoring USA borders as a countermeasure against illegal immigration [1].

The global UAV payload market was valued at $43.7 billion at the end of 2012, and is estimated to increase to $68.6 billion by 2022. The number of UAVs increased dramatically in civilian use such as the registered number of UAVs in use in the U.S. exceeded 200000 in the first 20 days of January 2016, just days after the USA Federal Aviation

[1]Husam A. Foudeh with the Electric Power and Drives Group, Cranfield University, Bedford, MK43 0AL, UK Husam-Abdel-Rasoul-Yousef.Foudeh@cranfield.ac.uk

[2]Patrick Luk with the Electric Power and Drives Group, Cranfield University, Bedford, MK43 0AL, UK p.c.k.luk@cranfield.ac.uk

[3]James F. Whidborne with the Centre for Aeronautics, Cranfield University, Bedford, MK43 0AL, UK j.f.whidborne@cranfield.ac.uk

Fig. 1. (a) Ascending Hummingbird, (b) Phantom 3 Standard DJI, (c) Firefly, (d) Indago

Administration (FAA) started requiring owners to officially register [2].

The quadrotor performance depends critically on control strategies, which in turn depend on the underlying dynamic model. This area is non-trivial since quadrotors are always under-actuated and unstable with significant non-linearity and strong dynamic coupling. Moreover, quadrotor stability can easily be compromised by environmental disturbances [3]. In general, the quadcopter consists of four actuators connected with identical arms as shown in Figure. 2. It is important to note that there are two approaches to modelling the dynamics of quadrotors, Newton-Euler and Euler-Lagrange formulation. Newton-Euler depends on the spatial Cartesian coordinates to describe the system equations of motion and it is obtained by projecting the external forces that affect the quadrotor onto these coordinates. The Euler-Lagrange approach is less dependent on the coordinate system and makes use of the conservation of energy to derive the equations of motion instead [4].

The most important key point in using the quadrotors is how to control a quadrotor. Researchers tend to use different control approaches, some of them is based on the classical control such as PID [5], or Optimal Control such as Linear Quadratic Regulation (LQR) [6], whilst others use sliding-mode control [7], back-stepping control [8], and learning control which includes iterative learning control (ILC) [9].

Iterative Learning Control (ILC) is an advanced control method that aims to reduce the error between a desired reference trajectory signal and the system output by updating the control input signal in a repetitive manner each iteration. In a small number of cases Iterative learning control (ILC) has been applied to quadrotors. ILC can be used for systems in which the finite-duration task is repeated. Each trial must have the same initial conditions and ILC updates the input signal with the aim of ensuring that the system

output converge to a reference signal as the number of trials increase. In [10] iterative learning control was applied to a quadrotor to obtain increased performance through learning. The emphasis was on combining classical control methods with ILC. To control the quadcopter, three different methods were applied involving iterative learning control as follows: off-line ILC, on-line ILC, and a combination of both on-line and off-line ILC. Pipatpaibul et al. designed an on-line ILC update for quadrotor trajectory tracking control, employing an inner PD controller to stabilize the system [11]. The system showed large tracking error but ILC could reduce it in subsequent iterations. Another study was conducted by Zhaowei et al. who implemented ILC with an adaptive component to enhance the controller performance and robustness. This was applied to a XAIRCRAFT quadrotor and experimental results showed good tracking performance in the presence of disturbances [12].
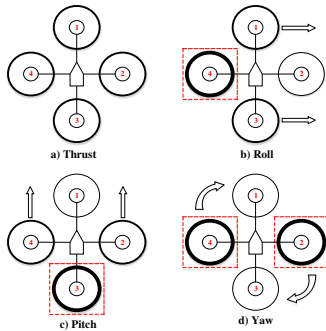

Fig. 2. Quadcopter movements with a) Thrust, b) Roll, c) Pitch, and d) Yaw.

Very few ILC algorithms have been implemented in this area, and no analysis or practical results exist to investigate robustness to modelling uncertainty and exogenous disturbances. No further applications of ILC have been applied to quadrotors, and the above methods are limited in terms of the accuracy they have attained. They also require a significant level of computation, as well as initial identification procedures and tuning. There is clear scope to evaluate a wider range of ILC methods on quadrotors. A prime example is the gradient-based method which seeks to minimize a cost function involving the tracking error. Our focus in this paper is the design of iterative learning controller based on the gradient method in which the Roll and Pitch will be more stable.

II. SYSTEM DYNAMICS

A. Dynamic for Hummingbird Quadrotor

The model complexity depends highly on underlying assumptions of the geometry and mass distribution. In term of quadrotors, most studies in the field of UAV used model based on the first order approximation which has been successfully utilized in various quadrotor control designs so far. The most common assumptions are as follows:

- The structure is rigid.
- The structure is symmetrical.
- The CoG (center of gravity) and the body fixed frame origin coincide.

Euler angle is the most commonly used approach to describe the orientation of a rigid body. Therefore, they will be adopted in representation of angles. Also, the reference system frames of the quadcopter is shown in Fig.3. The position of the quadcopter is expressed in the inertial frame F^i as $(x, y, z)^T$ axes with ξ. The attitude, is defined with three Euler angles η, First frame F^i is rotated around its z by ψ, to produce frame F^{v2}. Then F^{v2} is rotated about its y axis by θ to produce F^{v1}. Lastly F^{v1} is rotated about its x axis by ϕ to produce F^v. The frame F^v has the same orientation as body frame F^b as shown in Figure. 3. As defined in the equation (1) include the linear and angular position vectors
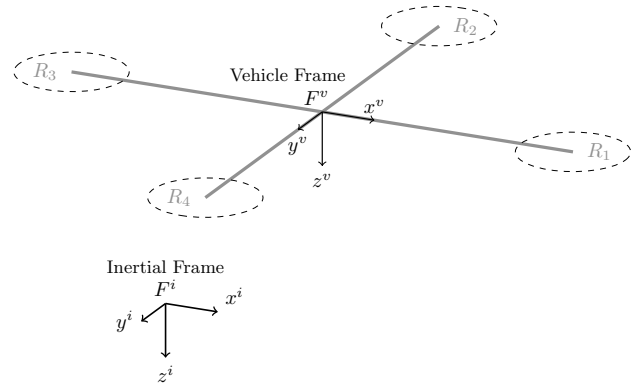

Fig. 3. The configuration of quadrotor UAV with respect to its frames

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (1)$$

The origin of the body frame is in the center of mass of the quadrotor. The velocity $(u, v, w)^T$ and the angular velocity $(p, q, r)^T$ of the quadrotor are defined with respect to the body frame F^b.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix}, \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2)$$

The transformation from a point p^v in F^v to a point p^{v1} in F^{v1} is given by $p^{v1} = R_v^{v1}(\psi)p^v$, where

$$R_v^{v1}(\psi) = \begin{bmatrix} C_\psi & -S_\psi & 0 \\ S_\psi & C_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$R_{v1}^{v2}(\theta) = \begin{bmatrix} C_\theta & 0 & S_\theta \\ 0 & 1 & 0 \\ -S_\theta & 0 & C_\theta \end{bmatrix} \quad (4)$$

$$R_{v2}^{b}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi & C_\phi \end{bmatrix} \quad (5)$$

The transformation from the vehicle frame to the body frame is given by

$$R_v^b(\phi;\theta;\psi) = R_{v2}^b(\phi)R_{v1}^{v2}(\theta)R_v^{v1}(\psi) =$$

$$\begin{bmatrix} C_\psi C_\theta & S_\psi C_\theta & -S_\theta \\ C_\psi S_\theta S_\phi - S_\psi C_\phi & S_\psi S_\theta S_\phi + C_\psi C_\phi & C_\theta S_\phi \\ C_\psi S_\theta C_\phi + S_\psi S_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi & C_\theta C_\phi \end{bmatrix} \quad (6)$$

The positions $(x,y,z)^T$ are inertial frame quantities, where velocities $(u,v,w)^T$ are body frame quantities. Therefore the relationship between position and velocities is given by

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ z \end{bmatrix} = (R_b^v)^{-1}\begin{bmatrix} u \\ v \\ w \end{bmatrix} = (R_v^b)^T\begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (7)$$

in which $S_x = \sin(x)$ and $C_x = \cos(x)$. Therefore, R_v^b

$$\begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \quad (8)$$

It is important to find the transformation between body angular velocities $(p,q,r)^T$ and rate of change of Euler angles $(\dot\phi, \dot\theta, \dot\psi)^T$.

$$\begin{bmatrix} \dot\phi \\ \dot\theta \\ \dot\psi \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi/C_\theta & C_\phi/C_\theta \end{bmatrix}\begin{bmatrix} p \\ q \\ r \end{bmatrix}, \quad (9)$$

It is possible to describe the quadrotor dynamics by considering the the formulation of Newton-Euler is most common due to its efficiency which can be rearranged as follows

$$m\frac{dv}{dt_i} = F \quad (10)$$

where F is the total applied to the CoG, and $\frac{d}{dt_i}$ is the time derivative in the inertial frame. The translational equation of motion can be derived from Newton's law as follows

$$m\left(\frac{dv}{dt_i} + \omega_{b/i} \times v\right) = F \quad (11)$$

where $\omega_{b/i}$ is the angular velocity of the airframe with respect to the inertial frame. Since the control force is computed and applied in the body coordinate system, and since ω is measured in body coordinates, we will express the equation (11) in body coordinates, where $v^b = (u,v,w)^T$, and $\omega_{b/i}^b = (p,q,r)^T$. Therefore, in the body frame, equation (11) can be defined as

$$\begin{bmatrix} \dot u \\ \dot v \\ \dot w \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + \frac{1}{m}\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \quad (12)$$

The rotational equation of motion can also be derived from Newton's law as follow

$$I\frac{d\omega}{dt} + \omega \times I\omega = \mathbf{T} \quad (13)$$

where $\mathbf{T}$ is the total moments applied to the quadrotor. The control inputs related to each rotor speed Ω_i are defined as follows:

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & lb & 0 & -lb \\ lb & 0 & -lb & 0 \\ d & d & d & d \end{bmatrix}\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \quad (14)$$

where b is the thrust coefficient, d is the drag coefficient and l is the arm length.

The full dynamic model for 3 DOF as following:

$$\dot p = \frac{1}{I_{xx}}[u_2 + qr(I_{yy} - I_{zz}) + qJ_P\Omega_r] \quad (15)$$

$$\dot q = \frac{1}{I_{yy}}[u_3 + pr(I_{zz} - I_{xx}) + pJ_P\Omega_r] \quad (16)$$

$$\dot r = \frac{1}{I_{zz}}[u_4 + qp(I_{xx} - I_{yy})] \quad (17)$$

B. Design the Test Frame

In case of Hummingbird quadrotor only six physical parameters $(m,l,I_{xx},I_{yy},I_{zz} \text{ and } J_P)$ are required to realise the 3 DOF system model as shown in the Table I. A test frame is design to hold the use Hummingbird quadrotor in place to allow for analysis of the quadrotor performance whilst tuning the control parameters was designed and built as shown in Figure. 4.

TABLE I
PHYSICAL PARAMETERS OF THE QUADROTOR

Parameter	Value
I_x_x	$10.7 \times 10^{-3} kg\,m^2$
I_y_y	$10.7 \times 10^{-3} kg\,m^2$
I_z_z	$18.4 \times 10^{-3} kg\,m^2$
Rotor I_z_z (J_P)	$47 \times 10^{-6} kg\,m^2$
Quadrotor Mass	0.547 kg
Arm Length	0.168 m

The test bed is constructed from 21.5 (mm) PVC pipe and bearings to allow for one DOF of rotation. PVC pipe was chosen as it was easy to obtain and allows for reconfiguration of the test bed if required. The quadrotor is locked in place, minimising the risk of damage during controller testing. The quadrotor sits in-between the two uprights, connected by dowels to the bearings.
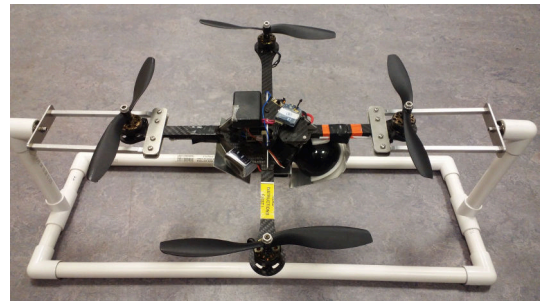

Fig. 4. A test frame design to control Hummingbird quadrotor

III. ILC DESIGN

The purpose of this section is to introduce the gradient-based algorithm. The system is assumed to be operating in a repetitive mode where at the end of each repetition, the state is reset to a specified repetition independent initial condition for the next operation during which a new control signal can be used. A reference signal $r(t)$ is assumed to be specified and the ultimate control objective is to find an input function $u^*(t)$ so that the resultant output function $y(t)$ tracks this reference signal $r(t)$ exactly on $[1;N]$. The process model is written in the form:

$$\begin{aligned} x_k(t+1) &= Ax_k(t) + Bu_k(t), \qquad 0 \leqslant t \leqslant N \\ y_k(t) &= Cx_k(t) \end{aligned} \tag{18}$$

where $x_k(t) \in \mathbb{R}^n$, $u_k(t) \in \mathbb{R}^m$, $y_k(t) \in \mathbb{R}^p$ and $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ are the system matrices. Moreover, x_k, u_k and y_k are the states, inputs and outputs vectors respectively as trial k.

A. Gradient-based ILC (G-ILC)

Comparing to a simple ILC controllers, gradient-based ILC relies on the system model to achieve faster error convergence and uses the properties of gradient descent to construct the ILC control action update. This is done by minimizing a cost function of the form

$$\min_{\mathbf{u}} J(\mathbf{u}_k) = \frac{1}{2}\|\mathbf{e}_k\|^2 = \frac{1}{2}\|\mathbf{y}_d - G\mathbf{u}_k\|^2 \tag{19}$$

where

$$G = \begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ g_{N-1} & g_{N-2} & \cdots & g_0 \end{bmatrix} \tag{20}$$

$$g_k = CA^{k+n_r-1}B \qquad k = 0,1,2....,N-1$$

and the tracking error $\mathbf{e}_k$ from the kth trial which is the error between the actual outputs $\mathbf{y}_k$ of the system and their desired reference signal $\mathbf{y}_d$ is then

$$\mathbf{e}_k = \mathbf{y}_d - \mathbf{y}_k \tag{21}$$

Using gradient descent to solve equation (19) optimization problem yields

$$\mathbf{u}_{k+1} = \mathbf{u}_k - \beta \bigtriangledown J(\mathbf{u}_k) \tag{22}$$

$$= \mathbf{u}_k - \beta \bigtriangledown \frac{1}{2}\|\mathbf{y}_d - G\mathbf{u}_k\|^2 \tag{23}$$

$$= \mathbf{u}_k - \beta G^T(\mathbf{y}_d - G(\mathbf{u}_k)) \tag{24}$$

$$= \mathbf{u}_k + \beta G^T \mathbf{e}_k \tag{25}$$

where β is the learning gain.
From equation (25), the error evolution for the gradient ILC can be derived as

$$\mathbf{e}_{k+1} = \mathbf{y}_d - G\mathbf{u}_{k+1} \tag{26}$$

$$= \mathbf{y}_d - G(\mathbf{u}_k + \beta G^T \mathbf{e}_k) \tag{27}$$

$$= (I - \beta GG^T)\mathbf{e}_k \tag{28}$$

by choosing the learning gain β from the range $0 < \beta < 2/\bar{\sigma}(G)$ where $\bar{\sigma}(G)$ is the largest singular of the matrix G, it can be easily shown that $\|I - \beta GG^T\| < 1$. Therefore, the error converges monotonically to zero as the trials k goes to infinity [13].

Instead of heuristically selecting the learning gain β from the previous range, the author in [14] proposed that the learning gain is to be chosen every iteration so that the error convergence rate is optimized. For this purpose, equations (24) and (28) could be rewritten as

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \beta_k G^T \mathbf{e}_k \tag{29}$$

$$\mathbf{e}_{k+1} = (I - \beta_k GG^T)\mathbf{e}_k \tag{30}$$

where the iteration varying learning gain β_k is obtained by minimizing:

$$J(\beta_k) = \|\mathbf{e}_{k+1}\|^2 + w\beta_k^2 \tag{31}$$

where w is a small positive weighting number. Substituting equation (30) in equation (31) we can write

$$J(\beta_k) = ((I - \beta_k GG^T)e_k)^T((I - \beta_k GG^T)e_k) + w\beta_k^2 \tag{32}$$

$$= e_k^T e_k - 2\beta_k e_k^T GG^T e_k + \beta_k^2 e_k^T GG^T GG^T e_k + w\beta_k^2 \tag{33}$$

After differentiating equation (33) with respect to β_k and set it to zero, the optimal learning gain is given by

$$\beta_k = \frac{e_k^T GG^T e_k}{e_k^T GG^T GG^T e_k + w} \tag{34}$$

$$= \frac{(G^T e_k)^T G^T e_k}{(GG^T e_k)^T GG^T e_k + w} \tag{35}$$

$$= \frac{\|G^T e_k\|^2}{\|GG^T e_k\|^2 + w} \tag{36}$$

The necessary and sufficient conditions to ensure error convergence are

$$\|\mathbf{e}_{k+1}\| < \|e_k\| \quad \forall\, k \geq 0 \quad \text{and} \quad \lim_{k \to \infty} e_k = 0 \tag{37}$$

From equation (30) we can write

$$\|\mathbf{e}_{k+1}\|^2 - \|e_k\|^2 = e_k^T(I - \beta_k GG^T)^T(I - \beta_k GG^T)e_k - e_k^T e_k \tag{38}$$

$$= e_k^T((I - \beta_k GG^T)^2 - I)e_k \tag{39}$$

$$= e_k^T(-2\beta_k GG^T + \beta_k^2 GG^T GG^T)e_k \tag{40}$$

$$= \beta_k^2(-2\frac{e_k^T GG^T e_k}{\beta_k} + e_k^T GG^T GG^T e_k) \tag{41}$$

$$= \beta_k^2(-2\frac{\|G^T e_k\|^2}{\beta_k} + \|GG^T e_k\|^2) \tag{42}$$

From equation (36) we have

$$\frac{\|G^T e_k\|^2}{\beta_k} = \|GG^T e_k\|^2 + w \tag{43}$$

Substituting equation (42) in equation (43)

$$\|e_{k+1}\|^2 - \|e_k\|^2 = \beta_k^2(-2(\|GG^T e_k\|^2 + w) + \|GG^T e_k\|^2$$
$$= -\beta_k^2(2w + \|GG^T e_k\|^2) \leqslant 0$$

(44)

From equation (44) it can be deduced that $\|e_{k+1}\| = \|e_k\|$ iff $\beta_k = 0$ and from equation (36), $\beta_k = 0$ iff $e_k = 0$ since GG^T is a positive definite matrix. Therefore the conditions in equation (37) are satisfied and the convergence is monotonic.

IV. RESULTS AND DISCUSSION

In this section, simulations and experiments are carried out in order to assess the performance of G-ILC applied to the linearized quadrotor dynamics. The simulations here achieved using MATLAB with fixed-step size solver to produce a suitable estimation of a real-time commercial flight controller. The assessment criteria in comparing the results will rely on the second norm of the error between the reference signal and the actual output and for fair comparison the duration of all simulated algorithms for both trajectories will be 8 seconds with a step size of 0.02 seconds which is achievable by most commercial microcontrollers. Initially, the baseline evaluation for quadrotor control will be taken from a PID controller and the performance compared with G-ILC.

A. Experimental Result for 3 DOF

The input demand using here is $\theta_{ref} = \sin\theta$ as shown in Figure. 5 (Trajectory I). The following figure shows the variation in pitch over a time of about 30s.
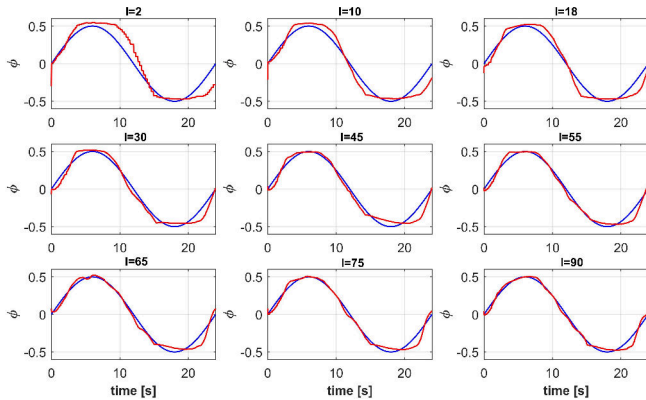
Fig. 5. The PI controller on trajectory I for during variation of ϕ.

Table II shows the norm error reduction, where the error value was at the highest level when using integral gain $K_i = 0.02$ and the tuning parameters chosen with the $K_p=0.95$ with value at 0.1117 but this reduce the error when the integral gain $K_i = 0.9$ is increased and with the $K_p=0.98$ at value 0.0455. Unfortunately, any time reduction such as reduction from t=30s to t=15s or t=10s will impact the results in an undesirable way, where there is no effect of the increase or decrease by the gain.

The step response (Trajectory II) of the controller Hummingbird quadrotor shown below includes the demand $\theta_{ref} = $

K_i	$\|\hat{\theta} - \theta\|$	K_d	$\|\hat{\theta} - \theta\|$
0.02	0.1117	0.1	0.0678
0.01	0.0885	0.2	0.0624
0.03	0.0732	0.3	0.0549
0.055	0.0543	0.5	0.0496
0.065	0.0539	0.7	0.0522
0.09	0.0455	0.9	0.0543

$H(t - t_0)$. Figure. 5 demonstrates that there is no delay due to the simulations only without a practical experiment, while the Figure. 6 proves that this slightly delay is in the case of practical application within the laboratory environment.
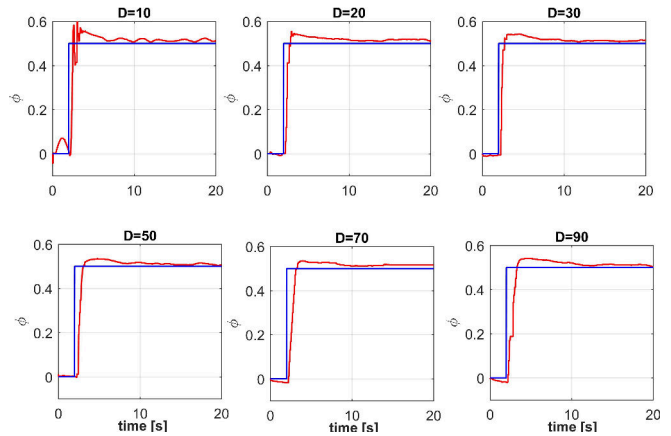
Fig. 6. The PD controller on trajectory II for during variation of θ.

The gradient ILC update equation (28) has been applied to track the reference and an optimal gain β is choose to be from 0.01 to 1. After applying a wide range of values β the best performance has been found to correspond to 0.1. The experimental results has shown a significant decrease of the error during first five iterations as shown in Figure. 7, just slightly fluctuation happen after the 6th iteration but didn't impact the performance.
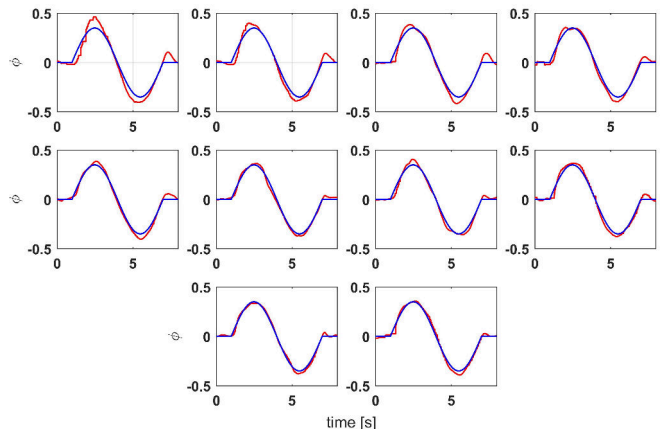
Fig. 7. Experimental Gradient-based ILC with different iteration.

Again it is important to prove that the system of G-ILC has monotonic convergence as shown in Figure. 8, this can be proven by the 2-norm condition of the error, the decrease happen from 1.315 at first iteration to the value of 0.548 at the 6th iteration.
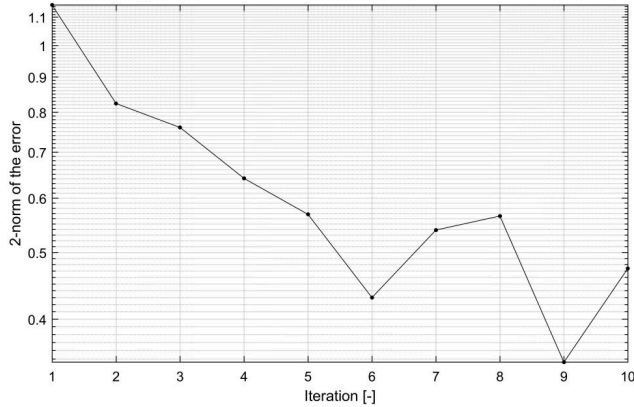

Fig. 8. Monotonic convergence result for gradient based ILC.

B. Comparison

The results of the proposed learning scheme are applied to Hummingbird quadrotor. In order to evaluate the quadcopter performance, the PID control is considered to compare. Therefore the tracking and step response will quantitatively evaluate the Gradient ILC performance as shown in Table III.

It is also important to prove that the system of Gradient-based ILC has monotonic convergence. The simulation results show a significant decrease of the error during different iteration. This can be proved by the condition of 2nd norm of the errors as shown in Table III with 0.3092 at the iteration 16th.

For experiment, the gradient-based ILC with the optimal gain formulation had a good overall performance and the error norm converged below the baseline value after the 2nd iteration for the attitude angles. Furthermore, as seen in Table IV, gradient-base ILC exhibited fluctuations in the error norms as the iteration increased, although it was minimal for the attitude angles. These fluctuations however did not affect the general trend as the error norm still converged after 10 iterations.

TABLE III

SIMULATION COMPARISON FOR ϕ AND θ WITH DIFFERENT CONTROL APPROACHES.

Control Approaches	Trial No	Norm of Error $\|\hat{\theta} - \theta\|$	Norm of Error $\|\hat{\phi} - \phi\|$
PID	-	2.82	2.82
Gradient ILC	1	1.92	1.93
	3	1.38	1.38
	6	0.332	0.331
	16	0.309	0.303

In this comparison, the baseline evaluation of quadrotor control has been taken from a PID controller scheme as shown in Table III and IV at value 2.82 for ϕ and θ and

3.366 for ψ. On the other hand, the PID had the poorest performance for attitude. This outcome was expected for several reasons, such as the unguaranteed monotonic convergence and increases sensitivity to noise due to derivative action. The baseline result was generated for a unit step and sinusoidal reference signal trajectories, These outcomes were compared with the proposed ILC algorithms. The ILC algorithms performed better than the baseline controller (PID) for both trajectory and the G-ILC method had a dominating performance overall, and managed to substantially reduce the tracking error within 16 iterations as shown in Table III.

TABLE IV

EXPERIMENTAL COMPARISON FOR PID AND ILC ALGORITHMS ERROR 2-NORM.

ILC methods	Iteration(1)		Iteration(3)		Iteration(7)		Iteration(10)	
	$\|\mathbf{e}_\theta\|$	$\|\mathbf{e}_\psi\|$	$\|\mathbf{e}_\theta\|$	$\|\mathbf{e}_\psi\|$	$\|\mathbf{e}_\theta\|$	$\|\mathbf{e}_\psi\|$	$\|\mathbf{e}_\theta\|$	$\|\mathbf{e}_\psi\|$
PID	3.392	3.366	3.392	3.366	3.392	3.366	3.392	3.366
G-ILC	1.277	1.277	0.920	0.920	0.612	0.612	0.534	0.534

V. CONCLUSIONS

The proposed ILC controllers (G-ILC) have been formulated and implemented, and the results are compared against a standard PID controller, where the G-ILC has shown superior performance in terms tracking performance. Furthermore, the G-ILC controllers proposed here have demonstrated significant improvement over the existing PID controllers regarding error reduction and monotonic convergence.

The results in this paper serve to demonstrate that standard PID controllers alone cannot adequately track the reference in case the length time decreases to more than 30s. So, in order to produce better performance, the G-ILC control has been taken with a combined PID controller rather than implementing each controller individually. The combined controller can be tuned to perform very well and from this the reliability of the attitude controller for 3 DOF derived can be inferred. However, this combined control may yet produce give the desired performance and robustness for advanced applications. It is therefore important to develop more advanced and novel ILC controllers to achieve better performance in future.

In addition, the G-ILC managed to achieve monotonic error convergence and reduced the error norm below the baseline values for attitude angles, which justifies the fact that ILC algorithm have good performance and has smoother transitions in a way of tracking this type of reference. These confirm the theoretical predictions given in this paper. Only bench tests on the controllers under specific conditions have been reported here. It shows that there is an urgent need to look at the additional work that is needed to expand ILC to specify tracking, for example, along a straight line for a power line monitoring task, and to explore viable ways to achieve similar applications.

REFERENCES

[1] N. Hossein Motlagh, T. Taleb and O. Arouk, "Low-Altitude Unmanned Aerial Vehicles-Based Internet of Things Services: Comprehensive Survey and Future Perspectives," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 899-922, Dec. 2016.

[2] Hanscom AF, Bedford MA. Unmanned aircraft system (UAS) service demand 2015–2035, literature review & projections of future usage. Res. Innov. Technol. Admin., US Dept. Transp., Washington, DC, USA, Tech. Rep. DOT-VNTSC-DoD-13-01, Sep 2013.

[3] S. Bouabdallah, P. Murrieri and R. Siegwart, "Design and control of an indoor micro quadrotor," *IEEE International Conference on Robotics and Automation*, New Orleans, LA, USA, pp. 4393-4398 Vol.5, 2004.

[4] J. de Jesus Rubio, J. Humberto Perez Cruz, Z. Zamudio and A. J. Salinas, "Comparison of two quadrotor dynamic models," *IEEE Latin America Transactions*, vol. 12, no. 4, pp. 531-537, June 2014.

[5] J. Ren, D. Liu, K. Li, J. Liu, Y. Feng and X. Lin, "Cascade PID controller for quadrotor," *IEEE International Conference on Information and Automation (ICIA)*, Ningbo, pp. 120-124, 2016.

[6] S. Bouabdallah, A. Noth and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, Sendai, pp. 2451-2456 vol.3, 2004.

[7] M. K. Shaik and J. F. Whidborne, "Robust sliding mode control of a quadrotor," *UKACC 11th International Conference on Control (CONTROL)*, Belfast, pp. 1-6, 2016.

[8] Lining Tan, Libin Lu and Guodong Jin, "Attitude stabilization control of a quadrotor helicopter using integral Backstepping," *International Conference on Automatic Control and Artificial Intelligence (ACAI 2012)*, Xiamen, pp. 573-577, 2012.

[9] M. Hehn and R. D'Andrea, "An iterative learning scheme for high performance, periodic quadrocopter trajectories," *European Control Conference (ECC)*, Zurich, pp. 1799-1804, 2013.

[10] T. V. Dinh, C. T. Freeman, P. Lewin and Y. Tan, "Assessment of gradient-based point-to-point ILC for MIMO systems with varying interaction," *IEEE International Symposium on Intelligent Control*, Dubrovnik, pp. 1220-1225, 2012.

[11] Pipatpaibul, P.I. and Ouyang, P.R., 2013. Application of online iterative learning tracking control for quadrotor UAVs. ISRN robotics, 2013.

[12] M. Zhaowei, H. Tianjiang, S. Lincheng, K. Weiwei, Z. Boxin and Y. Kaidi, "An iterative learning controller for quadrotor UAV path following at a constant altitude," *Chinese Control Conference (CCC)*, Hangzhou, pp. 4406-4411, 2015.

[13] D. H. Owens, J. Hatonen and S. Daley, "Robust Gradient-based Iterative Learning Control," *IEEE International Conference on Networking, Sensing and Control*, London, pp. 163-168, 2007.

[14] J. J. Hatonen, K. Feng and D. H. Owens, "New connections between positivity and Parameter-Optimal Iterative Learning Control," *IEEE International Symposium on Intelligent Control*, Houston, TX, USA, pp. 69-74, 2003.