Application of Iterative Learning Tuning to a Dragonfly-like Flapping Wing Micro Aerial Vehicle

Chang-ping DU[1,2], Jian-xin XU[2], Yao ZHENG[1]

1. School of Aeronautics and Astronautics, Zhejiang University, Hangzhou, 310027, China

2. Department of Electrical and Computer Engineering, National University of Singapore, 117576, Singapore
E-mail: duchangping@zju.edu.cn

Abstract: The flight control design of a dragonfly-like flapping wing micro aerial vehicle (FWMAV) is studied in this paper. The main contribution of this work is to design an appropriate flight controller by incorporating a linear-quadratic regulator (LQR) method and an iterative learning tuning (ILT). The linear model of dragonfly-like FWMAV is developed at the equilibrium point. Then a linear-quadratic regulator approach is applied to design the flight controller of FWMAV. However the flight controller performance of FWMAV is sensitive to the input weighting matrix of LQR. In order to improve the flight controller, an iterative learning tuning is developed to optimize the input weighting matrix of LQR problem due to unknown constraints and relations between the state and the control input. Numerical simulation results show that the effectiveness and convergence performance of the flight controller of FWMAV are obtained.

Key Words: iterative learning tuning, dragonfly-like flapping wing micro aerial vehicle, linear quadratic regulator, mathematical model.

1 INTRODUCTION

The purpose of this research is to design a flight controller of a dragonfly-like flapping wing micro aerial vehicle. Compared with fixed wing micro aerial vehicle (MAV) and rotary wing MAV, flapping wing MAV has high energy efficiency during flight, flexibility in maneuverability and agility at low speed. Recently, researchers have placed a great emphasis on the development of flapping wing MAV [1-3]. However the flight of flapping wing MAV is more complex than flight with fixed or rotary wing, because the beating motion of flapping wings is the only means that can counter the gravity force and propel themselves against aerodynamic drag. Furthermore, the dragonfly-like FWMAV does not have the conventional control surfaces in tail. This yields a control difficulty due to the loss of the maneuverability in tail.

In [3] the complete dragonfly-like FWMAV model, which is highly nonlinear, was proposed in a companion form. A flight controller specific for dragonfly-like FWMAV was also developed, which is designed to iteratively solve for a desired control signal profile by means of a dual-loop nonlinear dynamic inversion with Newton-Raphson solution. However this flight controller cannot be applied to control the dragonfly-like FWMAV in practice due to its time-consuming iterative computation process.

The concept of iterative learning was first introduced in control to deal with a repeated control task without requiring the perfect knowledge such as the plant model or parameters [4]. Compared with most other learning methods, the convergence property of the iterative learning approach does not require the availability of the information on the gradient, which is usually required by many learning mechanisms as the greatest descending direction is used in the learning updates. Iterative learning control (ILC) is an effective control approach successfully applied to many fields [5-11].

In this work, a linear-quadratic regulator approach is applied to design the flight controller of a dragonfly-like FWMAV. The control performance of FWMAV is sensitive to the input weighting matrix of linear-quadratic regulator problem and there are some unknown constraints and relations between the state and the control input. Thus, an iterative learning tuning is developed to improve the flight controller by optimizing the input weighting matrix of the LQR. The iterative learning is able to achieve the learning convergence even if the plant model is unknown or partially unknown [12].

The rest of the paper is organized as follows. In Section 2, the dynamic model of a dragonfly-like FWMAV is briefly introduced. In Section 3, the proposed design method of the dragonfly-like FWMAV flight controller is developed in detail. Numerical examples are carried out to validate the effectiveness of the controller in Section 4. The paper ends with conclusion, given in Section 5.

2 DYNAMIC MODEL OF FWMAV

In this section, the model of the dragonfly-like FWMAV is given to study its control problem. According to [3] the state-space equation of the dragonfly-like FWMAV is

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}) \ , \tag{1}$$

where $\boldsymbol{x} = [x_g, y_g, z_g, V_x, V_y, V_z, \varphi, \theta, \gamma, w_x, w_y, w_z]'$ is the state vector. The control vector is $\boldsymbol{u} = [\Omega_1, \theta_{p,1}, \Omega_2, \theta_{p,2}, \varphi_{tail}, \theta_{tail}, \varphi_{nose}]'$. x_g, y_g, z_g are the position states of the dragonfly-like FWMAV in the ground coordinate system. V_x, V_y, V_z are the velocity states of body coordinate system on x-axis, y-axis and z-axis respectively. φ, θ, γ are the heading, pitch and roll angles of FWMAV respectively. w_x, w_y, w_z are the

978-1-4673-9714-8/16/$31.00 ©2016 IEEE

angular velocity states of body coordinate system on x-axis, y-axis and z-axis respectively. Ω_1 is the angular velocity of the fore-wing. Ω_2 is the angular velocity of the hind-wing. $\theta_{p,1}$ is the flapping direction of the fore-wing. $\theta_{p,2}$ is the flapping direction of the hind-wing. φ_{tail} is the yaw angle of the tail. θ_{tail} is the pitch angle of the tail. φ_{nose} is the yaw angle of the head. M_x, M_y, M_z are the moments of body coordinate system on x-axis, y-axis and z-axis respectively. I_x、I_y、I_z、I_{xy}、I_{yz}、I_{zx} are the relative moments of inertia of FWMAV. L_1 is the lift force of the fore-wing, D_1 is the drag force of the fore-wing and C_1 is the side force of the fore-wing. L_2 is the lift force of the hind-wing, D_2 is the drag force of the hind-wing and C_2 is the side force of the hind-wing. $\alpha_{w,1}$ is the angle of attack at the local fore-wing. $\beta_{w,1}$ is the sideslip angle of the local fore-wing. $\alpha_{w,2}$ is the angle of attack at the local hind-wing. $\beta_{w,2}$ is the sideslip angle of the local hind-wing.

$$
\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix}
V_x\cos\varphi\cos\theta - V_y\sin\varphi\cos\gamma + V_y\cos\varphi\sin\theta\sin\gamma + V_z\sin\varphi\sin\gamma + V_z\cos\varphi\sin\theta\cos\gamma \\
V_x\sin\varphi\cos\theta + V_y\cos\varphi\cos\gamma + V_y\sin\varphi\sin\theta\sin\gamma - V_z\cos\varphi\sin\gamma + V_z\sin\varphi\sin\theta\cos\gamma \\
-V_x\sin\theta + V_y\cos\theta\sin\gamma + V_z\cos\theta\cos\gamma \\
-w_yV_z + w_zV_y - g\sin\theta \\
-w_zV_x + w_xV_z + g\cos\theta\sin\gamma \\
-w_xV_y + w_yV_x + g\cos\theta\cos\gamma \\
w_y\frac{\sin\gamma}{\cos\theta} + w_z\frac{\cos\gamma}{\cos\theta} \\
w_y\cos\gamma - w_z\sin\gamma \\
w_x + w_y\sin\gamma\cdot\mathrm{tg}\theta + w_z\cos\gamma\cdot\mathrm{tg}\theta \\
\frac{I_z\cdot[-(I_z-I_y)w_yw_z + I_{zx}w_xw_y] + I_{zx}\cdot[-(I_y-I_x)w_xw_y - I_{zx}w_yw_z]}{I_x\cdot I_z - I_{zx}^2} \\
\frac{1}{I_y}[-(I_x-I_z)w_xw_z - I_{zx}(w_x^2 - w_z^2)] \\
\frac{I_{zx}\cdot[-(I_z-I_y)w_yw_z + I_{zx}w_xw_y] + I_x\cdot[-(I_y-I_x)w_xw_y - I_{zx}w_yw_z]}{I_x\cdot I_z - I_{zx}^2}
\end{bmatrix},
$$

$$
\boldsymbol{h}(\boldsymbol{x},\boldsymbol{u}) = \begin{bmatrix}
0 \\ 0 \\ 0 \\
\frac{1}{m}(-D_1\cos\alpha_{b,w1}\cos\beta_{b,w1} + C_1\cos\alpha_{b,w1}\sin\beta_{b,w1} + L_1\sin\alpha_{b,w1} - D_2\cos\alpha_{b,w2}\cos\beta_{b,w2} + C_2\cos\alpha_{b,w2}\sin\beta_{b,w2} + L_2\sin\alpha_{b,w2}) \\
\frac{1}{m}(-D_1\sin\beta_{b,w1} - C_1\cos\beta_{b,w1} - D_2\sin\beta_{b,w2} - C_2\cos\beta_{b,w2}) \\
\frac{1}{m}(-D_1\sin\alpha_{b,w1}\cos\beta_{b,w1} + C_1\sin\alpha_{b,w1}\sin\beta_{b,w1} - L_1\cos\alpha_{b,w1} - D_2\sin\alpha_{b,w2}\cos\beta_{b,w2} + C_2\sin\alpha_{b,w2}\sin\beta_{b,w2} - L_2\cos\alpha_{b,w2}) \\
0 \\ 0 \\ 0 \\
\frac{I_z\cdot M_x + I_{zx}\cdot M_z}{I_x\cdot I_z - I_{zx}^2} \\
\frac{M_y}{I_y} \\
\frac{I_{zx}\cdot M_x + I_x\cdot M_z}{I_x\cdot I_z - I_{zx}^2}
\end{bmatrix},
$$

The aerodynamic forces, such as $L_1, D_1, C_1,\ L_2, D_2$ and C_2, are the function of the states and the control inputs. The moments, such as M_x, M_y and M_z, are also the function of the states and the control inputs. The detailed derivation of equations for the aerodynamic forces, moments, angles of attack, and sideslip angles, are given in [3]. It is noted that this FWMAV is a system of first order nonlinear differential equations, and is non-affine in input vector $\boldsymbol{u}$.

3 FLIGHT CONTROLLER DESIGN

In this section the flight control design of the dragonfly-like FWMAV is developed by incorporating LQR and ILT. As mentioned above, LQR is applied to design the flight controller of the dragonfly-like FWMAV. ILT is to improve the flight controller by optimizing the input weighting matrix of LQR. First, the linear model of the dragonfly-like FWMAV is presented. Second, the objective function of ILT is studied. At last, the iterative learning model and flowchart are provided.

In order to derive the linear model of dragonfly-like FWMAV, the equilibrium state $\boldsymbol{x}_{\mathrm{d}}$ is chosen first. The equilibrium input, namely $\boldsymbol{u}_{\mathrm{d}}$, is the stable solution of formula (1) at the state $\boldsymbol{x}_{\mathrm{d}}$. We will obtain the linear model of the dragonfly-like FWMAV at the equilibrium state and the equilibrium input as follow

$$\Delta\dot{\boldsymbol{x}} = \left[\frac{\partial f(x)}{\partial x}\bigg|_{x_{\mathrm{d}}} + \frac{\partial \boldsymbol{h}(x,u)}{\partial x}\bigg|_{x_{\mathrm{d}},u_{\mathrm{d}}}\right]\cdot\Delta\boldsymbol{x} + \frac{\partial \boldsymbol{h}(x,u)}{\partial u}\bigg|_{x_{\mathrm{d}},u_{\mathrm{d}}}\cdot\Delta\boldsymbol{u}\,,$$

$$\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_{\mathrm{d}},\ \Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_{\mathrm{d}}\,.$$

For simplicity, the above equation is rewritten as below

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{A}\cdot\Delta\boldsymbol{x} + \boldsymbol{B}\cdot\Delta\boldsymbol{u}\,, \tag{2}$$

$$\boldsymbol{A} = \left[\frac{\partial f(x)}{\partial x}\bigg|_{x_{\mathrm{d}}} + \frac{\partial \boldsymbol{h}(x,u)}{\partial x}\bigg|_{x_{\mathrm{d}},u_{\mathrm{d}}}\right],\ \ \boldsymbol{B} = \frac{\partial \boldsymbol{h}(x,u)}{\partial u}\bigg|_{x_{\mathrm{d}},u_{\mathrm{d}}}\,.$$

According to linear-quadratic regulator method of LTI system, the objective function and the input $\Delta\boldsymbol{u}$ are respectively

$$J_{\mathrm{LQR}} = \frac{1}{2}\int_0^{\infty}(\Delta\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{Q}\cdot\Delta\boldsymbol{x} + \Delta\boldsymbol{u}^{\mathrm{T}}\cdot\boldsymbol{R}\cdot\Delta\boldsymbol{u})dt,$$

$$\Delta\boldsymbol{u} = -\boldsymbol{R}^{-1}\cdot\boldsymbol{B}^{\mathrm{T}}\cdot\boldsymbol{P}\cdot\Delta\boldsymbol{x}\,,$$

where $\boldsymbol{Q}$ is a symmetric positive definite 12×12 matrix. The input weighting matrix $\boldsymbol{R}$ is a symmetric positive definite 7×7 matrix. $\boldsymbol{P}$ is the solution of the Riccati equation. The control input of the dragonfly-like FWMAV is obtained as follow

$$\boldsymbol{u} = \boldsymbol{u}_{\mathrm{d}} + \Delta\boldsymbol{u} = \boldsymbol{u}_{\mathrm{d}} - \boldsymbol{R}^{-1}\cdot\boldsymbol{B}^{T}\cdot\boldsymbol{P}\cdot(\boldsymbol{x} - \boldsymbol{x}_{\mathrm{d}})\,. \tag{3}$$

As mentioned above, the control performance of our dragonfly-like FWMAV is sensitive to the input weighting matrix $\boldsymbol{R}$ of LQR. In order to improve the flight controller by optimizing the input weighting matrix of LQR, the objective function of ILT is chosen below

$$J_{\mathrm{IL}}(\boldsymbol{R}) = \exp\{\int_{t_0}^{T} g(\boldsymbol{x}_1)\cdot dt\}\cdot\int_{t_0}^{T}(|\boldsymbol{e}(t,\boldsymbol{R})|\cdot t)dt, \tag{4}$$

$$0\le \boldsymbol{t}_0 < T < \infty\,,\ \boldsymbol{e}(t,\boldsymbol{R}) = \boldsymbol{x} - \boldsymbol{x}_{\mathrm{ref}}\,,$$

$$g(\boldsymbol{x}_1) = \begin{cases} 0 & |\boldsymbol{x}_1| < \boldsymbol{x}_{1,max} \\ \frac{|x_1|}{x_{1,max}} - 1 & \text{others} \end{cases},\ \boldsymbol{x}_1 = [V_x, V_z, \theta, \gamma]',$$

where the error $\boldsymbol{e}(t,\boldsymbol{R})$ is the difference between the reference, $\boldsymbol{x}_{\mathrm{ref}}$ and the state, $\boldsymbol{x}$. The reference is slightly close to the equilibrium state. $\boldsymbol{t}_0$ is the initial time. T is the final time. $\boldsymbol{x}_{1,max}$ is a selected maximum cap of the considered state $\boldsymbol{x}_1$. The introduction of the deadzone function $g(\boldsymbol{x}_1)$ provides additional penalty to the state $\boldsymbol{x}_1$ when it is far away from the equilibrium. According to (4), the faster the convergence rate, the smaller the value of objective function for ILT.

To minimize the objective function (4), the following iterative learning law (5)-(7) is adopted according to [12], which searches the gradient direction with trials, hence is suitable for ILT problem in this work because the mapping between the ILT objective function $J_{\mathrm{IL}}(\boldsymbol{R}_i)$ and the input weighting matrix is unavailable. ILT law is

$$\boldsymbol{R}_{i+1} = \boldsymbol{R}_i - \boldsymbol{\gamma}_i\cdot J_{\mathrm{IL}}(\boldsymbol{R}_i)\,, \tag{5}$$

$$\boldsymbol{\varepsilon}_i = \left|\frac{\boldsymbol{R}_i - \boldsymbol{R}_{i-1}}{J_{\mathrm{IL}}(\boldsymbol{R}_i) - J_{\mathrm{IL}}(\boldsymbol{R}_{i-1})}\right|\,, \tag{6}$$

$$\boldsymbol{\gamma}_i = \pm\lambda\cdot\boldsymbol{\varepsilon}_i,\ \lambda\in(0,1]\,, \tag{7}$$

where $\boldsymbol{R}_{i+1}$, $\boldsymbol{R}_i$, $\boldsymbol{R}_{i-1}$ are the value of the input weighting matrices at the (i+1)-th iteration, the i-th iteration, the (i-1)-th iteration respectively. $\boldsymbol{\varepsilon}_i$ is the inverse of the estimated amplitude of the gradient. $\boldsymbol{\gamma}_i$ is the learning gain. The gradient coefficient λ is in the interval of (0,1].

The detail flowchart is shown in Fig. 1.

4 NUMERICAL SIMULATION EXAMPLE

In order to verify the proposed flight controller design, numerical simulation study is performed in this section. The dynamic model parameters are the same as the dynamic parameters in [3]. The initial time and the final time are given as $\boldsymbol{t}_0 = 0$ s, $\boldsymbol{T} = 250$ s respectively. The maximum of the considered state is $\boldsymbol{x}_{1,max} = [30\mathrm{m/s}, 30\mathrm{m/s}, \pi, \pi]$. According to (2), the linear model of the dragonfly-like FWMAV at the state $\boldsymbol{x}_{\mathrm{d}} = [0\ \mathrm{m},\ 0\ \mathrm{m},\ -5\ \mathrm{m},\ 5.0\ \mathrm{m/s},\ 0.0\ \mathrm{m/s},\ 1.0\ \mathrm{m/s}, 0.0\ \mathrm{rad},\ 0.5\ \mathrm{rad},\ 0.0\ \mathrm{rad},\ 0.0\ \mathrm{rad/s},\ 0.0\ \mathrm{rad/s},\ 0.0\ \mathrm{rad/s}\,]'$ and the input $\boldsymbol{u}_{\mathrm{d}} = [15.0\ \mathrm{rad/s},\ 1.6049\ \mathrm{rad},\ 15.0\ \mathrm{rad/s},\ 1.6120\ \mathrm{rad},\ 0.0\ \mathrm{rad},\ 0.0\ \mathrm{rad},\ 0.0\ \mathrm{rad}\,]'$ is given as

$$\boldsymbol{A} = \begin{bmatrix}
0 & 0 & 0 & 0.8776 & 0 & 0.4794 & 0 & -1.5195 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 4.8673 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & -0.4794 & 0 & 0.8776 & 0 & -4.8673 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.1053 & 0 & 6.0194 & 0 & -1.7937 & 0 & 0 & -0.6253 & 0 \\
0 & 0 & 0 & 0 & -4.9152 & 0 & 0 & 0 & 1.7937 & 0.6253 & 0 & -1.6674 \\
0 & 0 & 0 & -2.8703 & 0 & -6.6309 & 0 & -0.9799 & 0 & 0 & 1.6674 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.1395 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0.5463 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0013 & 0 & 0 & 0 \\
0 & 0 & 0 & -0.0569 & 0 & -0.1292 & 0 & -0.0191 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0959 & 0 & 0 & 0 & -0.0326 & 0 & 0 & 0
\end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.9654 & 303.8839 & 0.9607 & 283.3786 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.07447 & -1957.5535 & -1.0805 & -1957.6232 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1.6025 & 0 & -0.3945 \\
0.0167 & 30.4595 & -0.05882 & -106.5730 & 0 & -0.8754 & 0 \\
0 & 0 & 0 & 0 & 0.8754 & 0 & -0.2155
\end{bmatrix}.$$

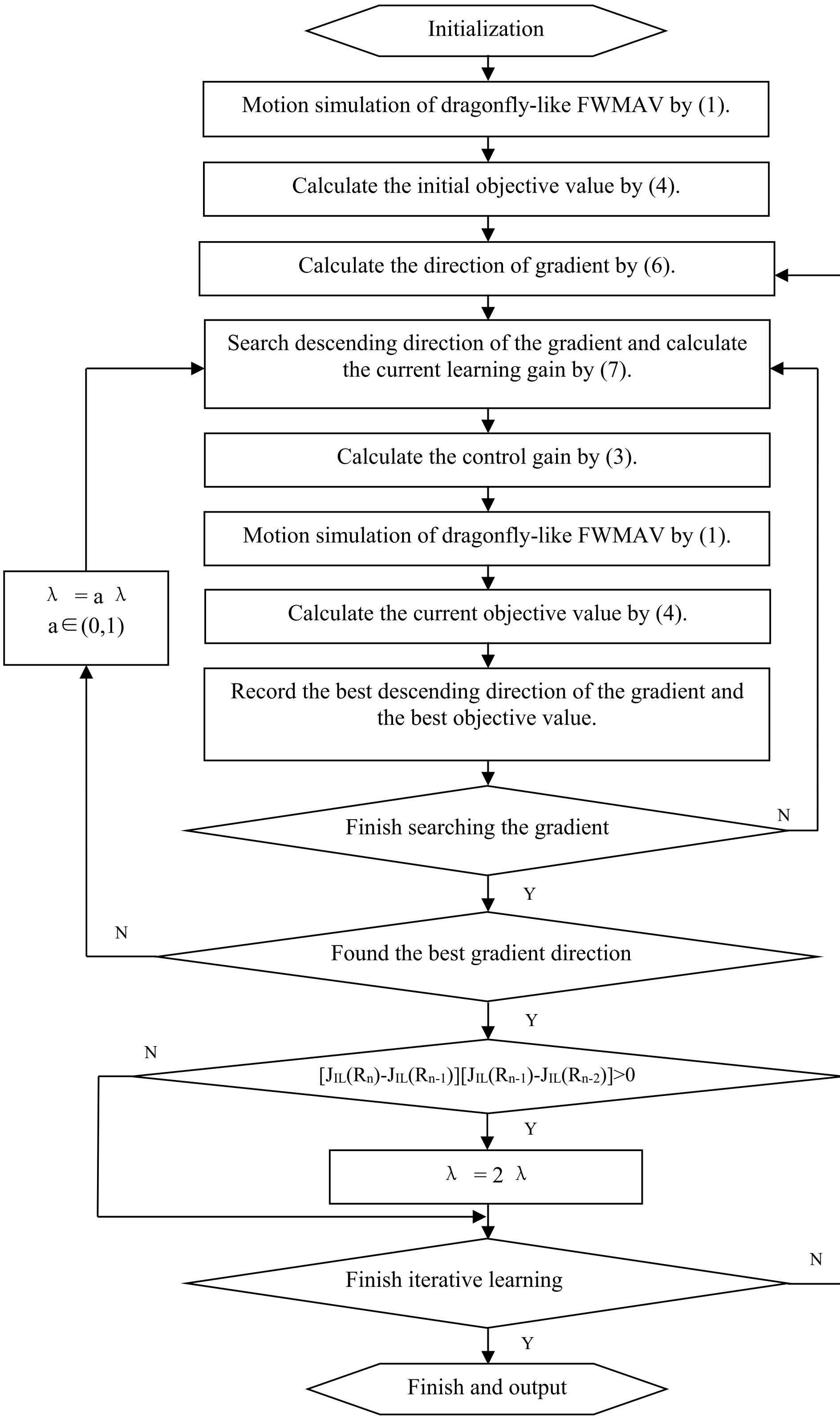

Fig. 1 The flowchart of the proposed flight controller design method for the dragonfly-like FWMAV, a ∈ (0,1) is the constant.

The weighting matrix $\boldsymbol{Q}$ in the objective function of LQR is given as $\boldsymbol{Q} = \mathrm{diag}(0, 0, 0, 2.0, 0.5, 1.5, 2.0, 4.5, 1.2, 0, 0, 0)$. The initial input weighting matrix $\boldsymbol{R}_0$ in the objective function of LQR is given as $\boldsymbol{R}_0 = \mathrm{diag}(10, 1500, 10, 1500, 1500, 1500, 1500)$. The initial direction of the gradient is given as $\boldsymbol{\varepsilon}_0 = \boldsymbol{R}_0$. The initial gradient coefficient is $\lambda = 10^{-6}$, $a = 0.5$. The maximum number of iterations is 70.

The initial state is $\boldsymbol{x}(0)$ = [0 m, 0 m, -5 m, 0.01 m/s, 0.00 m/s, -0.01 m/s, 0.0 rad, 0.01 rad, 0.0 rad, 0.0 rad/s, 0.0 rad/s, 0.0 rad/s]'. The initial control input is $\boldsymbol{u}(0)$ = [11.2951 rad/s, 1.6049 rad, 13.7869 rad/s, 1.6120 rad, 0.0 rad, 0.0 rad, 0.0 rad]' . The reference pitch angle is $\theta_{\mathrm{ref}} = 0.8$ rad, and the reference roll angle is $\gamma_{\mathrm{ref}} = 0.3$ rad. The reference velocities are $\mathrm{V_{x,ref}} = 2.5$ m/s and $\mathrm{V_{z,ref}} = 0.0$ m/s. The simulation results are depicted in Fig.2(a) ~ Fig.2(d) without ILT optimization and Fig.3(a) ~ Fig.3(d) with ILT optimization. Fig.2(a) shows the actual velocity of FWMAV and reference velocity trajectory. Fig.2(b) shows the velocity error between the actual velocity and reference velocity. Fig.2(c) shows the actual angle of FWMAV and reference angular trajectory. Fig.2(d) shows the angular error between the actual attitude and reference attitude. Fig. 3(a) shows the actual velocity of FWMAV and reference velocity trajectory. Fig.3(b) shows the velocity error between the actual velocity and reference velocity. Fig.3(c) shows the actual angle of FWMAV and reference angular trajectory. Fig.3(d) shows the angular error between the actual attitude and reference attitude. Fig.4 shows the objective value of the iterative learning tuning. Fig.5 shows the diagonal elements' value of the input weighting matrix at each iteration of ILT.

The final input weighting matrix $\boldsymbol{R}_{70}$ in the objective function of LQR is $\boldsymbol{R}_{70} = \mathrm{diag}(28.9990,\ 374.6713,\ 25.6242,\ 320.2425, 750.9434,\ 7024.1682,\ 1329.7880)$ at the 70-th iteration. The initial objective function of ILT is $J_{\mathrm{IL}}(\boldsymbol{R}_0) = 3.6000 \times 10^5$. The final objective function of ILT is $J_{\mathrm{IL}}(\boldsymbol{R}_{70}) = 2.1508 \times 10^5$ at the 70-th iteration. There are two reasons the objective function cannot converge to zero. One reason is the velocity and attitude angle fluctuate slightly around their desired values in the convergence stage. As shown in Fig.2(a), Fig.2(c), Fig.3(a) and Fig.3(c), the aerodynamic forces and moments of our dragonfly-like FWMAV fluctuate during the upstroke and the downstroke of a flapping cycle, as well as the changing aerodynamic direction and value of wings. The other reason is the existence of the transition process when the state converges to the desired value.

It can be observed in Fig.3(a) and Fig.3(b) that the velocities V_x, V_z of the dragonfly-like FWMAV with the proposed flight controller converge to the reference velocities within 40 seconds. In comparison, the velocities V_x, V_z of the dragonfly-like FWMAV without the proposed flight controller converge to the reference velocities within 100 seconds as shown in Fig.2(a) and Fig.2(b). Next we observe in Fig. 3(c) and Fig.3(d) that pitch and roll angles of the dragonfly-like FWMAV with the proposed flight controller converge to the reference value within 50 seconds. In comparison, as shown in Fig.2(c) and Fig.2(d), pitch and roll angles of the dragonfly-like FWMAV without the proposed flight controller converge to the reference value within 90 and 110 seconds, respectively. These results illustrate the improvement of the proposed flight controller via ILT in decreasing the convergence time of FWMAV velocities and attitude angles.

It is also noted that the velocities in Fig.3(a) are smoother or less oscillatory than that in Fig.2(a) without ILT. Similarly, we can observe that the roll angle fluctuates with large amplitude in Fig.2(c) and Fig.2(d), but responses much smoother in Fig. 3(c) after ILT. Fig. 4 and Fig.5 show that the convergence of iterative learning tuning is achieved within 30 iterations.

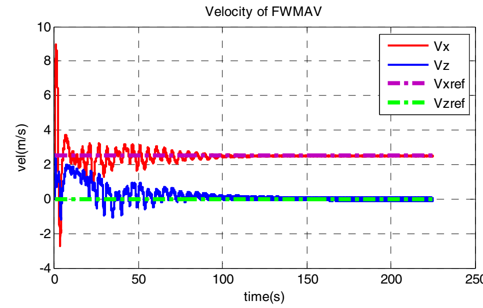

Fig. 2(a) Velocity response curves of FWMAV without ILT optimization, $\mathrm{V_{x,ref}} = 2.5$ m/s , $\mathrm{V_{z,ref}} = 0.0$ m/s

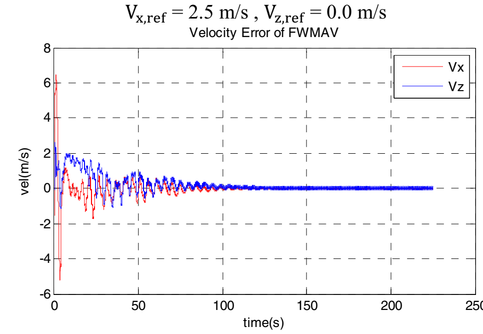

Fig. 2(b) Velocity error curves of FWMAV without ILT optimization

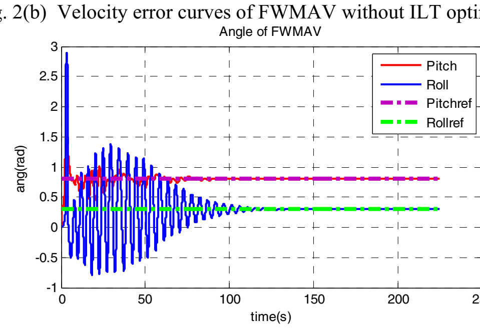

Fig. 2(c) Angular response curves of FWMAV without ILT optimization, $\theta_{\mathrm{ref}} = 0.8$ rad, $\gamma_{\mathrm{ref}} = 0.3$ rad

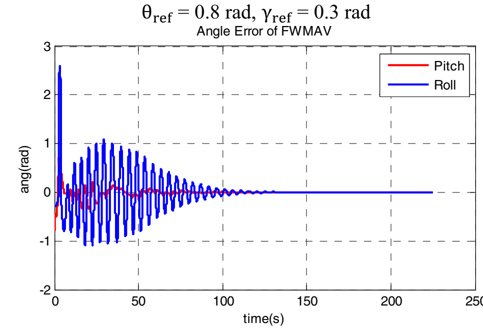

Fig. 2(d) Angular error curves of FWMAV without ILT optimization

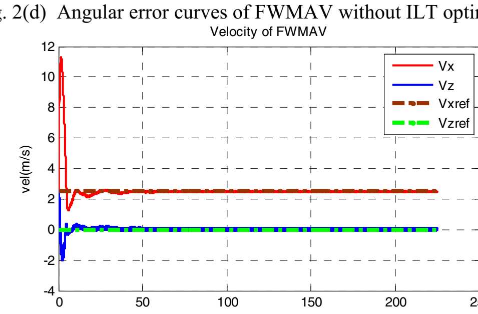

Fig. 3(a) Velocity response curves of FWMAV with ILT optimization, $\mathrm{V_{x,ref}} = 2.5$ m/s , $\mathrm{V_{z,ref}} = 0.0$ m/s

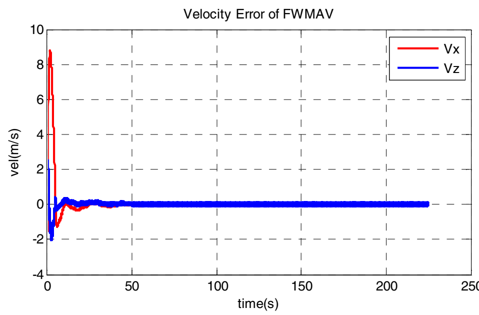

Fig. 3(b) Velocity error curves of FWMAV with ILT optimization

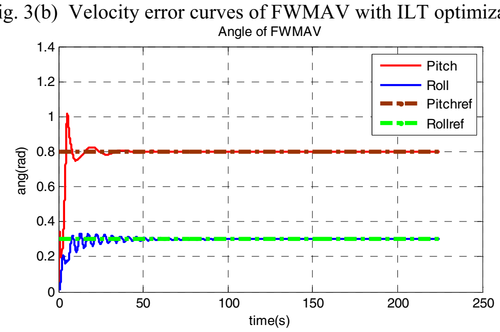

Fig. 3(c) Angular response curves of FWMAV with ILT optimization, $\theta_{ref} = 0.8$ rad, $\gamma_{ref} = 0.3$ rad

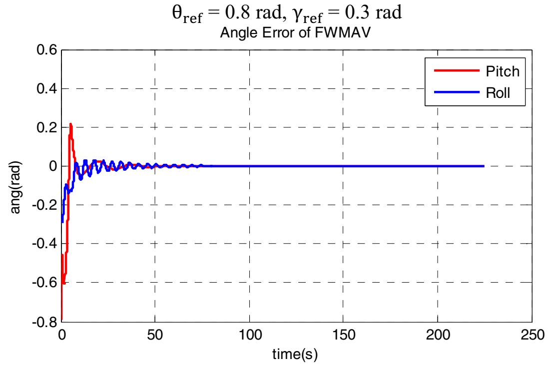

Fig. 3(d) Angular error curves of FWMAV with ILT optimization

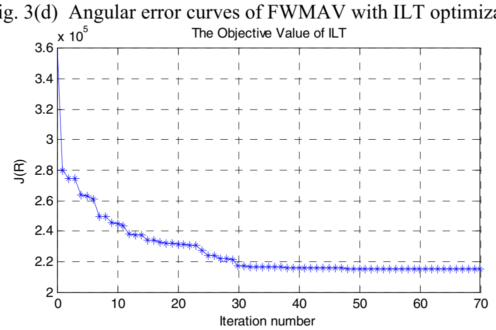

Fig. 4 Objective function curves of the iterative learning tuning

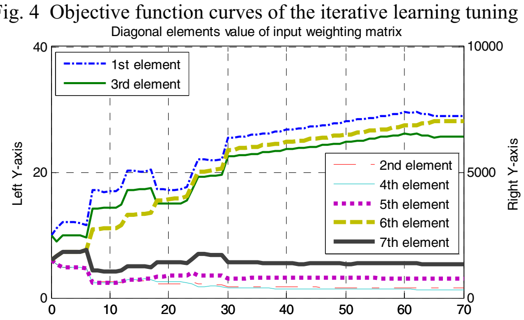

Fig. 5 Diagonal elements' curves of the input weighting matrix. The left y-axis corresponds to the 1st and 3rd diagonal element of input weighting matrix. The right y-axis corresponds to the other diagonal elements of input weigthing matrix.

5 CONCLUSION

This paper focuses on the controller design of the dragonfly-like FWMAV. It mainly employs LQR to design the flight controller. An ILT is further developed to improve the flight controller by optimizing the input weighting matrix, based on the fact that the control performance is sensitive to the input weighting matrix. Numerical simulation results show that the effectiveness and convergence performance of the proposed flight controller for the dragonfly-like FWMAV.

ACKNOWLEDGEMENT

This research is supported by China Scholarship Council.

REFERENCES

[1] W.L. Chan, Q.V. Nguyen, D. Marco, Pitch and yaw control of tailless flapping wing MAVs by implementing wing root angle deflection, IMAV, 34-41, 2014.

[2] J.Y. Sun, C.X. Pan, J. Tong, J. Zhang, Coupled model analysis of the structure and nano-mechanical properties of dragonfly wings, IET Nanobiotechnology, Vol.4, No.1, 10-18, 2010.

[3] C.P. Du, J.X. Xu, Y. Zheng, Modeling and control of a dragonfly-like micro aerial vehicle, Advances in Robotics & Automation, 2015 (accepted)

[4] S. Arimoto, S. Kawamura, F. Miyazaki, Bettering operation of robots by learning, Journal of Robotic Systems, Vol.1, No.2, 123-140, 1984.

[5] Z.S. Hou, J.X. Xu, J.W. Yan, An iterative learning approach for density control of freeway traffic flow via ramp metering, Transportation Research, Part C – Emerging Technology, Vol.16, No.1, 71-97, 2008.

[6] M. Mezghani, G. Roux, M. Cabassud, M.V. Le, B. Dahhou, G. Casamatta, Application of iterative learning control to an exothermic semibatch chemical reactor, IEEE Transactions Control Systems Technology, Vol.10, No.6, 822-834, 2002.

[7] K.L. Moore, Y.Q. Chen, V. Bahl, Monotonically convergent iterative learning control for linear discrete-time systems, Automatica, Vol.41, No.9, 1529-1537, 2006.

[8] A. Tayebi, Adaptive iterative learning control for robot manipulators, Automatica, Vol.40, No.7, 1195-1203, 2004.

[9] J.X. Xu, T. Zhu, Dual-scale direct learning of trajectory tracking for a class of nonlinear uncertain systems, IEEE Transactions on Automatic Control, Vol.44, No.10, 1884–1888, 1999.

[10] J.X. Xu, Y. Tan, A composite energy function based learning control approach for nonlinear systems with time-varying parametric uncertainties, IEEE Transactions Automatic Control, Vol.47, No.11, 1940–1945, 2002.

[11] J.X. Xu, J. Xu, T.H. Lee, Iterative learning control for systems with input deadzone, IEEE Transactions Automatic Control,Vol.50, No.9, 1455–1459, 2005.

[12] J.X. Xu, D.Q. Huang, Optimal tuning of PID parameters using iterative learning approach, In Proceedings of the 22nd IEEE International Symposium on Intelligent Control, 226–231, 2007.

[13] E. Sariyildiz, H.Y. Yu, K. Ohnishi, A practical tuning method for the robust PID controller with velocity feed-back, Machines,No.3, 208-222, 2015.