Cascaded control for balancing an Inverted Pendulum on a Flying Quadrotor

Chao Zhang[ab*], Huosheng Hu[b], DongBing Gu[b] and Jing Wang[a]

[a]*Engineering Research Institute of USTB, University of Science and Technology Beijing, Beijing, China;*
[b]*Department of Computer Science and Electronic Engineering, University of Essex, Colchester, UK*

Abstract: This paper focuses on the flying inverted pendulum problem, i.e balancing a pendulum on a flying vehicle. After analyzing the system dynamic, a three loop cascade control strategy based on active disturbance rejection control (ADRC) is proposed. Not only the pendulum balancing, but also the trajectory tracking of the vehicle can be realized successfully even when strong disturbances exist. The simulation results from a 3D mechanical systems simulation platform including a comparison to a LQR controller are given to verify the control performance and robustness of the proposed strategy. Finally, the practical flying experiments are achieved in our lab.

Keywords: Inverted pendulum, Micro aerial vehicles; ADRC; Cascade control.

1. Introduction

The inverted pendulum is a classical nonlinear control problem and is a common benchmark to evaluate advanced control techniques. Recently, it has extended to many different scenarios, such as an inverted pendulum on a cart, multi-degrees pendulum and Furuta pendulum[1]. The dynamics of pendulum systems is related to two-wheels robots, rocket guidance, etc[2]. It has been investigated for several decades. Many researchers regard it as a benchmark for advanced control strategies, e.g. PID control [3], neural networks[4] and controlled Lagrangians [5].

With the recent advancement of MEMS (micro electro mechanical sensors) and energy storage devices, Micro Aerial Vehicles (MAVs) have demonstrated a great potential, and been widely used in many civil and military applications, e.g. wildfire monitoring, aerial filming, and pollution assessment due to their easy construction and ability to take-off and land vertically (VTOL) [6]. Meanwhile, these small flying machines, especially quadrotors have populated in the research institutes worldwide because of their simple steering principle and low cost [7], such as the Autonomous Systems Lab at Swiss Federal Institute of Technology [8] and the General Robotics, Automation, Sensing and Perception (GRASP) Laboratory at University of Pennsylvania [9].

Recently, the flying inverted pendulum has attracted much attention worldwide. It is a nonlinear, under-actuated, coupled system with 8 Degrees of freedom (DOFs), i.e. a 6-DOF quadrotor and a 2-DOF inverted pendulum. ETH Flying Lab has first investigated this problem [10]. After the nominal pendulum equilibrium is realized, linear state feedback controllers are designed to stabilize the whole system. Reinforcement learning was adopted to improve system performance by decomposing the task into two subtasks[11], i.e. initial balancing and balanced hover. In addition, a global stabilizing controller for inverted pendulum derived from controlled Lagrangians is used to command the desired angels with another parallel quadrotor position controller together [12].

However, linear state feedback design is based on a linearization model of the equilibrium. This

---

*Corresponding author. Email: czhangd@essex.ac.uk

is only valid in a small dynamics range and vulnerable to external disturbances. Meanwhile, the parameters of two independent controllers for pendulum and quadrotor are difficult to adjust and need many trials. Although reinforcement learning could be deployed in the controller design by dividing the task into two subtasks, the designed controllers actually can not control the pendulum and quadrotor positions together. The final quadrotor position may be dozens of meters away from the starting point [11]. Other nonlinear control theory such as controlled lagrangian method for a single inverted spherical pendulum shows very explicit in mathematical proof, however, the controller form would be too complicated to be deployed practically. The research of classic inverted pendulum problem may provide some ideas, but unable to be deployed directly.

The flying pendulum problem has an unstable equilibrium, i.e. the upwards vertical position of the pendulum. Its internal and external disturbances are ubiquitous and difficult to be described in mathematical methods. Other factors such as communication delay and aerodynamics may degrade the control quality in practical systems [13]. Recently, many researchers analyzed practical complex systems in terms of anti-disturbance view and proposed two kinds of disturbance estimation techniques [14]. One is the disturbance observer technique, which is originally proposed by Ohnishi et al. [15], and has been widely applied to different systems, e.g., missile systems [16], humanoid joint [17], precise motion control systems [18]. and so on. Another one is Active Disturbance Rejection Control (ADRC), which lumps the internal uncertain dynamic and the external disturbances as a system state and on-line estimate them by an extended state observer (ESO), then compensates for them in control input signals. An ESO can estimate both the system states and the lumped disturbances. Meanwhile, ADRC design does not require an accurate mathematical model and is relatively simple to be realized in practical systems. It has been widely deployed in diverse examples of systems with promising results [19].

In [20], a linear high order observer-based linear controller for the Furuta pendulum is designed on the basis of a flat tangent linearization model. A two loop linear ADRC controller is designed to solve the classic inverted pendulum on a cart problem [21]. From the results of their works, ADRC shows great potential to solve underactuated nonlinear systems control problems. Therefore, inspired by all the previous work in this subject, a three-loop cascade ADRC architecture for the flying pendulum system is proposed to control the quadrotor track its reference trajectory while the pendulum can always maintain its position.

The inner loop is a high bandwidth attitude controller on the quadrotor, which tracks desired attitude angles using feedback from gyroscopes. The quadrotor has a fast tracking response to the desired commands since it has very low rotational inertia and powerful brushless motors. Then, on the basis of the fast inner loop, we design an ADRC controller for the pendulum position using tangent angles as pseudo control inputs.

To design the outer loop controller for quadrotor horizontal position, time scale control concept [22] is adopted since the position control of the inverted pendulum has to be guaranteed first and thus the position states of quadrotor has to be slower. An ADRC controller using the pendulum position as control inputs is used here and with small parameters to meet the time-scale separation principle. Another independent controller using total thrust as control input is designed to control the altitude of the quadrotor. From simulation, we find system performance is sensitive to the gain coefficient of control input, thus we introduce an adaptive gain parameter after analyzing the system. Through this multi-level design, each sub-loop is a negative feedback controller, which is different from the common LQR and PID controllers, i.e the vehicle loop is a positive feedback controller. This change can utilize the natural connection between all sub-loops and can improve security in practical experiments. The robustness of the system is validated by adding disturbances during path tracking. In addition, many communications are involved in this system such as TCP/IP connection from Vicon system to PC and wireless Zigbee connection from PC to the quadrotor. Time delay is an important issue here and may affect the system stability. Inspired by Lupashin's work [8], we proposed a compensation algorithm based on a tracking differentiator to generate the estimations of the feedbacks and velocities. It is simple and effective in practical system.

The remainder of this paper is organized as follows. Section 2 presents the dynamics models for the quadrotor-pendulum system used in this research. In Section 3, a cascade ADRC controller is designed for quadrotor trajectory tracking and balancing inverted pendulum. Simulation on Matlab SimMechanics platform [23] and practical flight tests are carried out in Section 4 to show the feasibility and effectiveness of the proposed control strategy. Finally, a brief conclusion and future work are given in Section 5.

2.   System Dynamics

A Hummingbird quadrotor made by Ascending Technology is used in this research [24]. Fig.1 shows two right-handed coordinate systems used in describing the MAV pose. The world frame, $\mathcal{W}$, is defined by axes X_W, Y_W and Z_W with Z_W axis pointing upward. The body frame, $\mathcal{B}$, coincides with the center of mass and is defined by the axes X_B, Y_B and Z_B. X_B is always aligned with the preferred forward direction and Z_B perpendicular to the plane of four rotors. In the following analysis, we assume that the body-fixed frame $\mathcal{B}$ is attached to the center of mass of the quadrotor and the pendulum is rigidly attached to the center as well.

2.1.   *Quadrotor-Pendulum Model*

As shown in Fig.2, the position vector of quadrotor is denoted by $\mathbf{p} = [x\ y\ z]^T$ in the world frame and the pendulum position is described by r and s, which represent the translational position of the pendulum mass center relative to the supporting point, i.e., r along the X_W-axis and s along the Y_W-axis. The relative height ζ can be calculated by $\sqrt{L^2 - r^2 - s^2}$ where L is the length from the bottom to the mass center of the pendulum and its position can be given by $\mathbf{p}_p = [x+r\ y+s\ z+\zeta]^T$. Since the pendulum has no moment of inertia about its z-axis, the rotation kinetic energy can be given by $\frac{1}{6}m_p L^2 \Omega_{p,xy}^T \Omega_{p,xy}$.

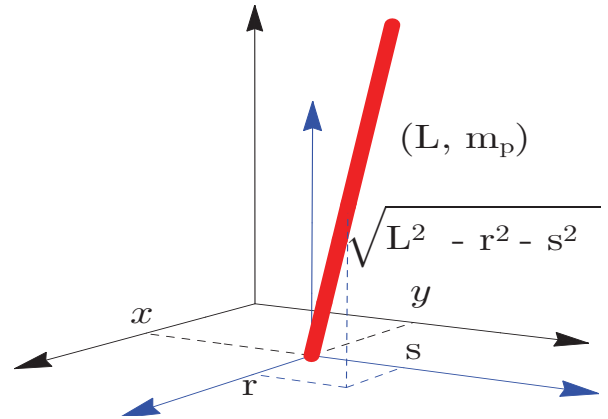Figure 1. Fixed body and world coordinate systems.

Figure 2. Inverted pendulum on top of a quadrotor.

From the rigid body kinematics, we can derive $\Omega_{p,xy} = \mathbf{p}_r \times \dot{\mathbf{p}}_r / L^2$ where $\mathbf{p}_r = [r\ s\ \zeta]^T$. The total kinetic energy T_p and the potential energy V_p of the pendulum are

$$T_p = \frac{1}{2}m_p((\dot{x}+\dot{r})^2 + (\dot{y}+\dot{s})^2 + (\dot{z} - \frac{r\dot{r}+s\dot{s}}{\zeta})^2) + \frac{1}{6}m_p L^2 \Omega_{p,xy}^T \Omega_{p,xy} \qquad (1)$$

$$V_p = m_p g(z + \zeta) \tag{2}$$

To derive the pendulum dynamics, applying Lagrangian mechanics as below

$$\frac{d}{dt}\left(\frac{\partial L_p}{\partial(\dot{r}, \dot{s})}\right) - \frac{\partial L_p}{\partial(r, s)} = 0 \tag{3}$$

where $L_p = T_p - V_p$. Then we obtain

$$\ddot{r} = -\frac{3}{4}\left(\frac{\zeta^2}{L^2 - s^2}\right)\ddot{x} + \frac{3r\zeta(g + \ddot{z})}{4(L^2 - s^2)} + \frac{r^3(\dot{s}^2 + s\ddot{s}) - 2r^2 s\dot{r}\dot{s} + r(-L^2 s\ddot{s} + s^3\ddot{s} + s^2\dot{r}^2 - L^2\dot{r}^2 - L^2\dot{s}^2)}{(L^2 - s^2)\zeta^2} \tag{4}$$

$$\ddot{s} = -\frac{3}{4}\left(\frac{\zeta^2}{L^2 - r^2}\right)\ddot{y} + \frac{3s\zeta(g + \ddot{z})}{4(L^2 - r^2)} + \frac{s^3(\dot{r}^2 + r\ddot{r}) - 2s^2 r\dot{s}\dot{r} + s(-L^2 r\ddot{r} + r^3\ddot{r} + r^2\dot{s}^2 - L^2\dot{s}^2 - L^2\dot{r}^2)}{(L^2 - r^2)\zeta^2} \tag{5}$$

From the dynamic equations, $[\ddot{x}\ \ddot{y}\ \ddot{z}]^T$ can be regarded as control input to the system. However, for Z-axis, we have to adopt an independent controller to keep the height of the quadrotor at a desired value. And considering the gain coefficient of $\ddot{z}$ is much smaller than that of $\ddot{x}$ and $\ddot{y}$, it can be regarded as a disturbance in the pendulum system when the height is changing. So the system dynamics of the pendulum can be written in the following form

$$[\ddot{r}\ \ddot{s}]^T = B[\ddot{x}\ \ddot{y}]^T + \mathbf{f_z}(r, s)\ddot{z} + \mathbf{f_n}(r, s, \dot{r}, \dot{s}) \tag{6}$$

where $\mathbf{f_z}$ and $\mathbf{f_n}$ are the second term and third term of right hand in (4) and (5) respectively, and

$$B = \begin{bmatrix} -\frac{3}{4}\left(\frac{\zeta^2}{L^2 - s^2}\right) & 0 \\ -\frac{3}{4}\left(\frac{\zeta^2}{L^2 - r^2}\right) & 0 \end{bmatrix} \tag{7}$$

2.2. *Quadrotor Dynamics*

It is reasonable to assume that the dynamics of the quadrotor is not affected by the movements of the pendulum since the mass and the inertia of the pendulum are one magnitude less than that of the quadrotor. The quadrotor is described by six degrees of freedom: the translation position in the world frame and the vehicle attitude parametered by XYZ-Euler angles (roll-ϕ,pitch-θ,yaw-ψ). Using The Newton's second law, translational equations of the MAV motion can be derived as below[25].

$$m\ddot{\mathbf{p}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R\begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} \tag{8}$$

where m, g denote the total mass and the gravitational constant respectively, R is the transformation matrix between $\mathcal{W}$ and $\mathcal{B}$ and T is the total thrust produced by propellers.

The attitude is not directly controllable, but it is related to the angular velocity of the quadrotor in the body frame and the angular acceleration is determined by three torques generated by four propellers

$$\dot{R} = R\hat{\Omega} \tag{9}$$

$$J\dot{\Omega} = -\Omega \times J\Omega + \tau. \tag{10}$$

where J denote the inertia matrix w.r.t the frame $\mathcal{B}$, $\mathbf{p}$ is the position vector w.r.t the frame $\mathcal{W}$. The control input $\tau = [\tau_1\ \tau_2\ \tau_3]^T$ represent the torques produced by propellers. $\Omega = [w_x, w_y, w_z]^{\mathrm{T}}$ denotes the angular velocity vector w.r.t the frame $\mathcal{B}$. The notation $\hat{\Omega}$ denotes the skew-symmetric matrix of Ω.

Based on these dynamic equations, we have done a high-bandwidth controller to track the desired euler angle values to generate torques commands. In order to transfer these commands to motor speeds, a brief motor model is introduced here. As seen in Fig.1, motor 1 is the motor on the $+X_B$ arm and the other three motors are allocated to $+Y_B$, $-X_B$, $-Y_B$ arms, respectively. For a typical multi-rotor flying vehicle, each rotor produces a thrust force F_i in its Z_B-axis and a torque M_i around its Z_B-axis. A basic relationship between them and trotation speed n_i is $F_i = k_f n_i^2, M_i = k_m n_i^2$. Then, we can write the general relationship in matrix form for a quad-rotor used in this paper.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ T \end{bmatrix} = \begin{bmatrix} 0 & lk_f & 0 & -lk_f \\ -lk_f & 0 & lk_f & 0 \\ k_m & -k_m & k_m & -k_m \\ k_f & k_f & k_f & k_f \end{bmatrix} \begin{bmatrix} n_1^2 \\ n_2^2 \\ n_3^2 \\ n_4^2 \end{bmatrix} \tag{11}$$

where l denotes the distance from rotor to the center of quadrotor, and the unit of motor speed is revolutions per minute (rpm). The parameters k_f and k_m can be regarded as constants and be determined from static thrust tests. Hence, we can achieve the required motor speeds by inverse operation of (11). The attitude control loop and the calculation of the motor speed run on the Asctec Autopilot board equipped on the quadrotor at 1KHz.

3.   Controller Design

The quadrotor-pendulum control system is integrated by three loops, i.e. the onboard attitude loop based on gyroscope feedbacks, the pendulum balancing loop and the quadrotor position loop. Since the system is highly underactuated and the vertical position of the pendulum is a unstable equilibrium, a cascaded control strategy is adopted, i.e. the mid loop is the pendulum control and the quadrotor position loop is the outer loop. Desired attitude angles and thrust would be their control inputs which are tracked by onboard inner loop.

3.1.   *Inverted Pendulum Loop*

The target of the mid loop is to control the pendulum position (r, s) falling within a very small region. From (8), we can derive

$$\begin{aligned} \ddot{x} &= (\sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi)T/m \\ \ddot{y} &= (-\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi)T/m \\ \ddot{z} &= (\cos\theta\cos\phi)T/m - g \end{aligned} \tag{12}$$

Substitute (12) into (6) and take the attitude angles as control inputs

$$\ddot{r} = -\frac{3}{4}(1 - \frac{r^2}{L^2 - s^2})g\tan\theta + f_r \tag{13}$$

5

$$\ddot{s} = \frac{3}{4\cos\theta}(1 - \frac{s^2}{L^2 - r^2})g\tan\phi + f_s \tag{14}$$

where we assume the yaw angle can always keep nearly 0 and f_r, f_s represent the dynamics produced by $\ddot{z}, s, r$, parameter uncertainties and external disturbances.

An ADRC controller is adopted here in order to handle complex disturbance compensation. Take the r-tanθ loop as example to show the controller details. The linear ESO (extended state observer) with the following discrete form is employed to estimate system states $[r \ \dot{r} \ f_r]^T$ including the extended state f_r.

$$\begin{cases} e_r = z_{r1} - \hat{r} \\ b_r = -\frac{3}{4}(1 - \frac{r^2}{L^2 - s^2})g \\ z_{r1} = z_{r1} + h(z_{r2} - \beta_{r1}e_r) \\ z_{r2} = z_{r2} + h(z_{r3} - \beta_{r2}e_r + b_r\tan\theta) \\ z_{r3} = z_{r3} + h(-\beta_{r3}e_r) \end{cases} \tag{15}$$

where $\hat{r}$ denotes the measurement value of r, e_r denotes the tracking error of the observer, $z_{ri}(i = 1, 2, 3)$ are the outputs of the ESO and denote the estimates of states r, $\dot{r}$ and total disturbances f_r respectively, h is the sampling time and $\beta_{ri}(i = 1, 2, 3)$ are observer gains to be tuned. Generally, the larger the observer gain is, the faster response and better performance will be. However, a large gain will increase noise sensitivity. According to [26], we can find a proper parameter, in compromise between the suppression performance and the noise tolerance in practical system tuning.

Through the ESO, observations on position, velocity and unknown lumped disturbances are achieved. Hence, the state error feedback control law is designed as follows

$$u_r = (k_{rp}(r_d - z_{r1}) - k_{rd}(z_{r2}) - z_{r3})/b_r \tag{16}$$

where $u_r = \tan\theta$, k_{rp}, k_{rd} are the controller gains and r_d is the reference command from outer loop.

The controller and observer of the $s - \tan\phi$ loop can be derived in the same way.

$$\begin{cases} e_s = z_{s1} - \hat{s} \\ b_s = \frac{3}{4\cos\theta}(1 - \frac{s^2}{L^2 - r^2})g \\ z_{s1} = z_{s1} + h(z_{s2} - \beta_{s1}e_s) \\ z_{s2} = z_{s2} + h(z_{s3} - \beta_{s2}e_s + b_s\tan\phi) \\ z_{s3} = z_{s3} + h(-\beta_{s3}e_s) \end{cases} \tag{17}$$

$$u_s = (k_{sp}(s_d - z_{s1}) - k_{sd}(z_{s2}) - z_{s3})/b_s \tag{18}$$

All the variables have the same definitions with that in (15) and (16). The desired angle commands of the onboard attitude loop can be achieved by antitangent calculation.

$$\theta_d = \arctan u_r, \ \ \phi_d = \arctan u_s \tag{19}$$

3.2.   Quadrotor Position Loop

In order to make the pendulum not fall down when the quadrotor is tracking a reference trajectory, a fast response speed of the pendulum loop has to be guaranteed and the quadrotor position loop has to be relatively slow. We will assume the variables r, s are at steady state due to this time

scale separation. The MAV reference trajectories are generated with sufficient smoothness and bounded time derivatives $p_d = [x_d(t),\ y_d(t),\ z_d(t)]^T$. The altitude of the quadrotor is controlled by an independent ADRC controller which can make $\ddot{z}$ always small all the time. Therefore, the outer loop controller is actually $x - r$ and $y - s$ loops. From (13)-(16) and denote external disturbances and unmodelled dynamics as ξ, we can derive

$$\begin{cases} \ddot{x} = b_x r + f_x \\ \ddot{y} = b_y s + f_y \end{cases} \tag{20}$$

where b_x, b_y are the estimated control input gains and f_x, f_y stand for the whole nonlinear dynamics and disturbances

$$\begin{cases} b_x = -\frac{g}{\zeta} \\ b_y = -\frac{g}{\zeta} \\ f_x = \frac{-4(r^3(\dot{s}^2+s\ddot{s})-2r^2 s\dot{r}\dot{s}+r(-L^2 s\ddot{s}+s^3\ddot{s}+s^2\dot{r}^2-L^2\dot{r}^2-L^2\dot{s}^2))}{3\zeta^4} + \xi_x \\ f_y = \frac{-4(s^3(\dot{r}^2+s\ddot{r})-2s^2 r\dot{s}\dot{r}+s(-L^2 r\ddot{r}+r^3\ddot{r}+r^2\dot{s}^2-L^2\dot{s}^2-L^2\dot{r}^2))}{3\zeta^4} + \xi_y. \end{cases} \tag{21}$$

The design of the outer ADRC controller is similar to the pendulum loop, hence the details are omitted and the results are showed as follows.

$$\begin{cases} e_x = z_{x1} - \hat{x} \\ z_{x1} = z_{x1} + h(z_{x2} - \beta_{x1} e_x) \\ z_{x2} = z_{x2} + h(z_{x3} - \beta_{x2} e_x + b_x r_d) \\ z_{x3} = z_{x3} + h(-\beta_{x3} e_x) \\ r_d = (k_{xp}(x_d - z_{x1}) + k_{xd}(\dot{x}_d - z_{x2}) - z_{x3})/b_x \end{cases} \tag{22}$$

$$\begin{cases} e_y = z_{y1} - \hat{y} \\ z_{y1} = z_{y1} + h(z_{y2} - \beta_{y1} e_y) \\ z_{y2} = z_{y2} + h(z_{y3} - \beta_{y2} e_y + b_y s_d) \\ z_{y3} = z_{y3} + h(-\beta_{y3} e_y) \\ s_d = (k_{yp}(y_d - z_{y1}) + k_{yd}(\dot{y}_d - z_{y2}) - z_{y3})/b_y \end{cases} \tag{23}$$

Note the controller parameters in the outer loop have to be much smaller than that in the inner r, s loop to implement the two-time scale control.

3.3. *Time Delay Compensation*

In our practical system, a commercial motion capture system VICON is deployed to measure the position of the quadrotor and the pendulum. A commercial computer running the position and pendulum control loops will acquire these data through TCP/IP protocol and send commands to the onboard controller via wireless Zigbee modules. The system latency will be introduced mainly by the communications. If the pendulum has made a big displacement during the time delay, the vehicle has to generate a very large acceleration to save the pendulum. However, the maximal acceleration has a limit, i.e around $13.1 m/s^2$ on Hummingbird, and our VICON area has a limit range as well. Therefore, a short delay may result in the pendulum falling down from the quadrotor since the quadrotor can not generate the needed acceleration or will fly out of the area which is not covered by VICON.

In this paper, a TD (tracking differentiator) is introduced to compensate for the time delay on the basis of ESO. The TD input is the raw measurement $\mathbf{y}$ and the outputs are filtered measurements $\tilde{\mathbf{y}}$ and estimated differentiations $\tilde{\dot{\mathbf{y}}}$. The specific tracking differentiator system is shown as below

$$\begin{cases} \text{fh} = \text{fhan}(\tilde{\mathbf{y}} - \mathbf{y}, \tilde{\dot{\mathbf{y}}}, r_0, h_0) \\ \tilde{\mathbf{y}} = \tilde{\mathbf{y}} + h\tilde{\dot{\mathbf{y}}} \\ \tilde{\dot{\mathbf{y}}} = \tilde{\dot{\mathbf{y}}} + h\text{fh}. \end{cases} \tag{24}$$

Where $\text{fhan}(x_1, x_2, r_0, h_0)$ is a special nonlinear function with the following form which can track the signal differential fast.

$$\begin{cases} d = r_0 h_0, d_0 = h_0 d, y = x_1 + h_0 x_2 \\ a_0 = \sqrt{d^2 + 8r_0|y|} \\ a = \begin{cases} x_2 + \frac{a_0 - d}{2}\text{sign}(y), & |y| > d_0 \\ x_2 + \frac{y}{h_0}, & |y| \le d_0 \end{cases} \\ \text{fhan} = -\begin{cases} r_0\text{sign}(a), & |a| > d \\ r_0\frac{a}{d}, & |a| \le d \end{cases} \end{cases} \tag{25}$$

where r_0 and h_0 are positive parameters to be designed to determine the tracking speed and the filtering performance, respectively. According to the TD output signals, we can adopt a conventional algorithm to predict the latency-compensated measurements $\hat{\mathbf{y}}$.

$$\hat{\mathbf{y}} = \tilde{\mathbf{y}} + \hat{\tau}\tilde{\dot{\mathbf{y}}} \tag{26}$$

where $\hat{\tau}$ is the estimation of the practical latency. $\hat{\mathbf{y}}$ is the input of ESO instead of directly measurement $\mathbf{y}$ and the latency-free control is generated with compensated signals. The whole time delay is 30ms~50ms from practical test results. We choose $\hat{\tau} = 40ms$ in our controller design.

4. Experimental Tests

4.1. *Simulation Results*

To make simulations as accurate as possible, a multibody simulation environment for 3D mechanical systems called SimMechianics is used instead of numerical equations calculations. The multibody system can be modelled using blocks representing bodies, joints, constraints, and force elements, and then SimMechanics formulates and solves the equations of motion for the complete mechanical system. Models from CAD systems, including mass, inertia, joint, constraint, and 3D geometry, can be imported into SimMechanics and an assemble file with practical data from quadrotor and pendulum drawn by Solidworks is used in this paper.

Additionally, an automatically generated 3D animation can visualize the system dynamics as shown in Fig.1. The inner onboard attitude loop consists of three individual PD controller which showed fast response and good robustness with well tuned parameters. These controller details are omitted here since there are many papers and works existed [24,27]. The altitude controller is an independent ADRC controller similar to the mid quadrotor $x - y$ position loop to command the thrust. The inner loop and the other higher level loops, i.e quadrotor position and pendulum position, are running at 1k Hz and 50Hz respectively, which are the same settings as our practical system.

Firstly, the simulation of set point tracking while balancing the pendulum is conducted. The quadrotor takes off and hovers in the beginning ($0 - 5s$), then a move command is send to the
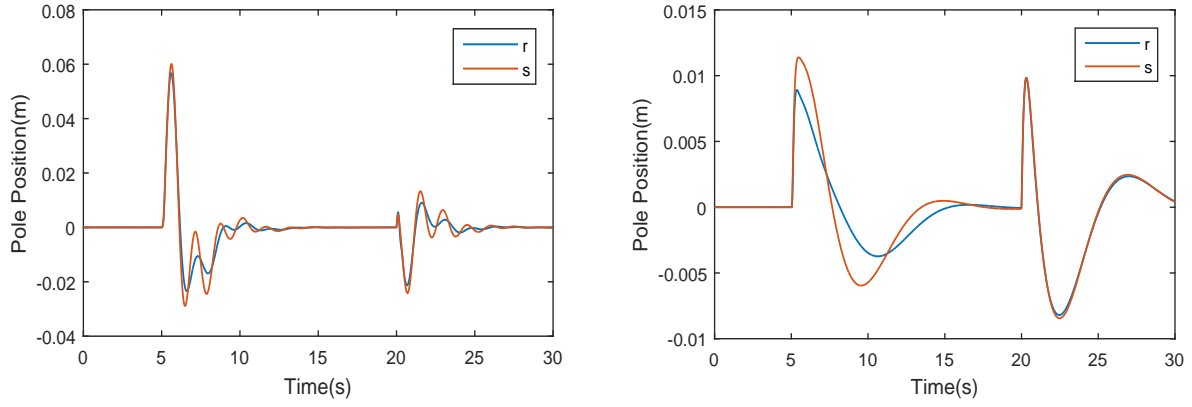
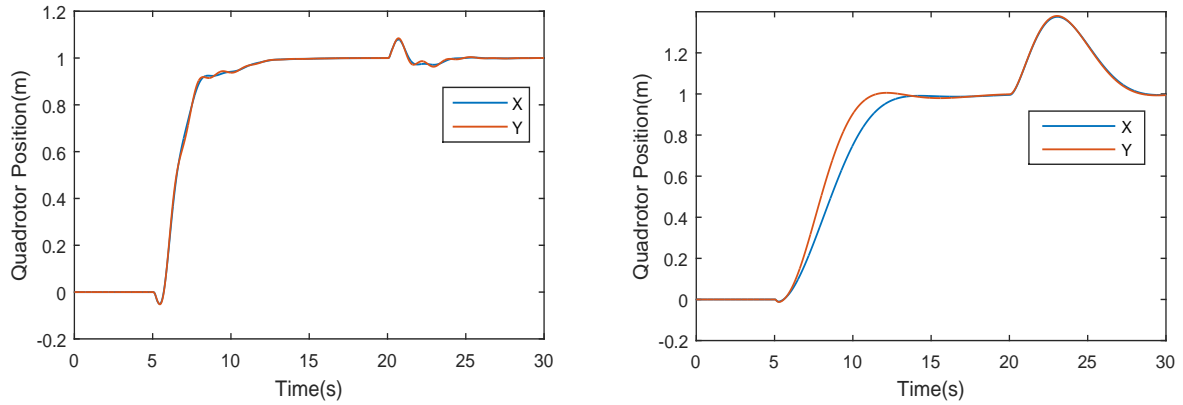Figure 3. Pendulum relative position output.

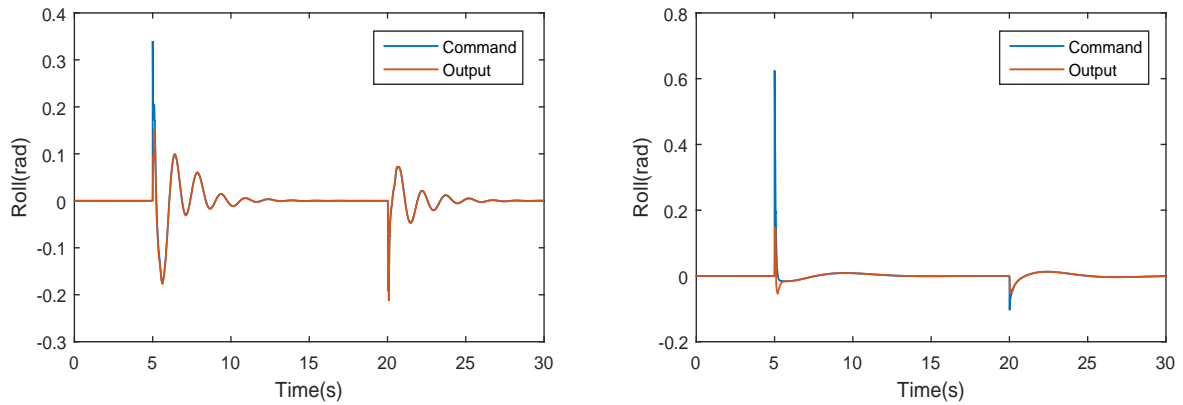

Figure 4. Quadrotor position output.

Figure 5. Roll angle command and output.

system at $t = 5s$, an $5N$ force disturbance with $0.01s$ duration is given to the pendulum at $t = 20s$ to validate its robustness finally. To show the advantage of the proposed method, the results of a LQR controller are depicted here to compare the system performance.

LQR (linear quadratic regulator) is an efficient methods to solve kinds of inverted pendulum problem. It has been used on classical pendulum on a cart case [28] and as well as the flying pendulum [10]. To design the LQR controller for the pendulum on a quadrotor system, a dynamic

9
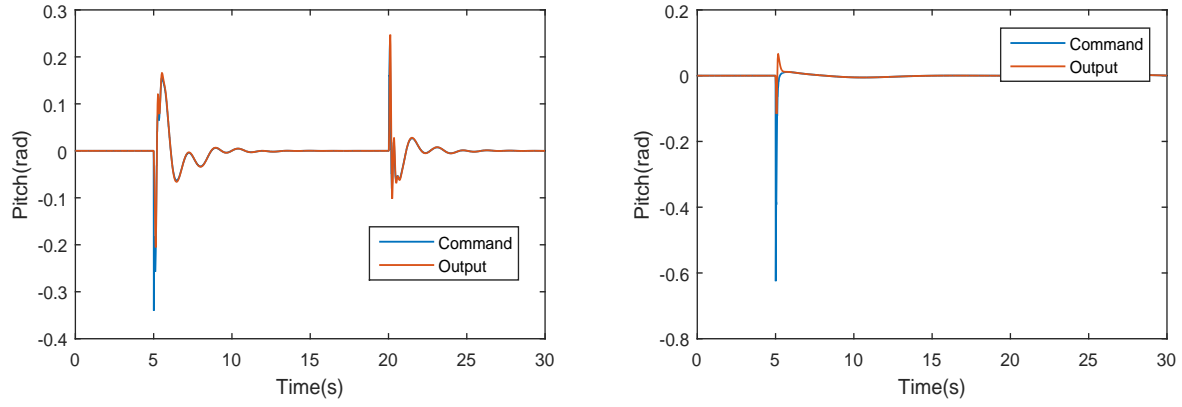
Figure 6. Pitch angle command and output.

linearization has to be done first. Using system state $[r \ \dot{r} \ s \ \dot{s} \ x \ \dot{x} \ y \ \dot{y}]^T$ and dynamic equations can be written in the following form by linearizing near the equilibrium point.

$$
\begin{cases}
\ddot{r} = \frac{3g}{4L}r - \frac{3}{4}g\tan\theta \\
\ddot{s} = \frac{3g}{4L}s + \frac{3}{4}g\tan\phi \\
\ddot{x} = g\tan\theta \\
\ddot{y} = -g\tan\phi
\end{cases}
\tag{27}
$$

Regard $[\tan\theta \ -\tan\phi]$ as control input, then the LQR controller can be achieved with a proper selection of matrix Q and R. After many trials, $Q = diag([10; 0; 10; 0; 1; 0; 1; 0])$ and $R = diag([1; 1])$ is selected and feedback matrix K can be derived. It can show excellent performance on the linear model, however the performance always decrease much on the nonlinear system and we have to tune the feedback gains many times for a good performance.

The simulation outputs are shown in Fig.3 to Fig.6, the left panels of figures are from the proposed controller and the right panels are from the LQR controller. On the other hand, LQR actually consists of two parallel controllers and the gains for the vehicle loop are positive sometimes,i.e. [1 1.9588] in this case. It means the vehicle position loop is a positive feedback control which is the same when using PID controller [3]. The positive loop will make the parameters tuning more complicated and may lose control of the vehicle when using in practical experiments. LQR needs system state information to build the feedback loop, however the velocities of the vehicle and the pendulum can not be measured directly. Therefore, some extra sensors or observer design is needed as well. But in the proposed control strategy, it is perfectly solved by the ESO design.

Another big difference is the weight of the pendulum state has to be much greater than that of the quadrotor position state in Q matrix design. It is very hard to find a balance between all system states and have a good performance on quadrotor position control especially in trajectory tracking applications. From Fig.3 and Fig.4, we can see although the pendulum is moving in a smaller range under the LQR controller, the quadrotor position tracking performance is much worse which takes nearly 10s to arrive at the set point. This problem can also be solved in the proposed controller since the on-line estimation of the disturbance can cover the work of the Integral part in a PID controller and ESO of two loops will make the control input balance between them. Thus both stable hovering and path tracking can be realized.

On the other hand, the proposed cascaded controller behaves more robust than the LQR, which has smaller reaction time to the disturbance and recovers faster, i.e, 10% vs 40% in displacement and 5s vs 10s in recover time. From the desired angles of two controllers generated in Fig.5 and
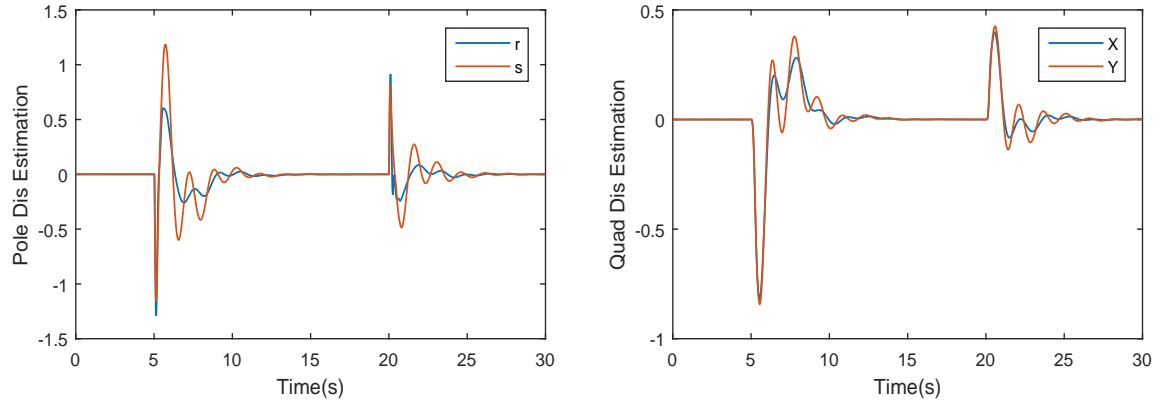
10

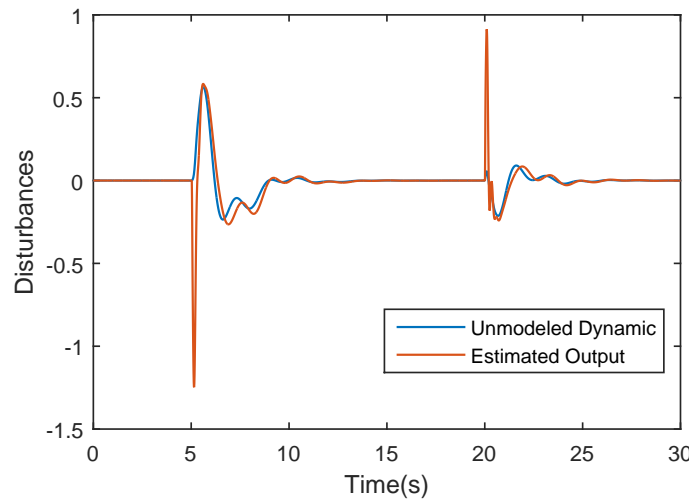Figure 7. Disturbance estimation results of system.

Figure 8. Comparison of calculated unmodelled dynamic and estimated disturbance output.

Fig.6, we can also find the system behavior is more aggressive under the proposed method.

The disturbance estimation outputs are shown in Fig.7, we can see the observer responses rapidly when the external force exerts. Take the pendulum r position loop for example to show the efficiency clearly, the unmodeled dynamics f_r is calculated with the description in 4. The comparison between it and the estimated disturbance is shown in Fig.8. From the figure, we can find the observer compensate for the internal dynamic well. We can see the estimation curve does not always coincides with the calculated dynamic curve. It is because that although we have try our best to build an accurate mathematical for the system, there are still some other factors may affect the system, for example the yaw angle can not always stay at 0 and the quadrotor is actually not a mass point. All the factors will have a influence which can be descriped by this Simmechanics simulation platform and be estimated by the observer. So we can see the estimated output is always a little larger than we calculated especially during the adjusting process.

Secondly, a circular trajectory tracking of the quadrotor is conducted. The reference trajectory is a circle with 1 meter radius and the center is the origin. The external disturbance is the same as the previous simulation which was exerted at 20s. From the results in Fig.9 and Fig.10, we can see the trajectory tracking of the quadrotor while balancing the pendulum can be fulfilled by the proposed controller even when disturbances exist. The attitude angles are shown in Fig.11. From the results, we can find we did not need to calculate the proper trajectory for the pendulum,
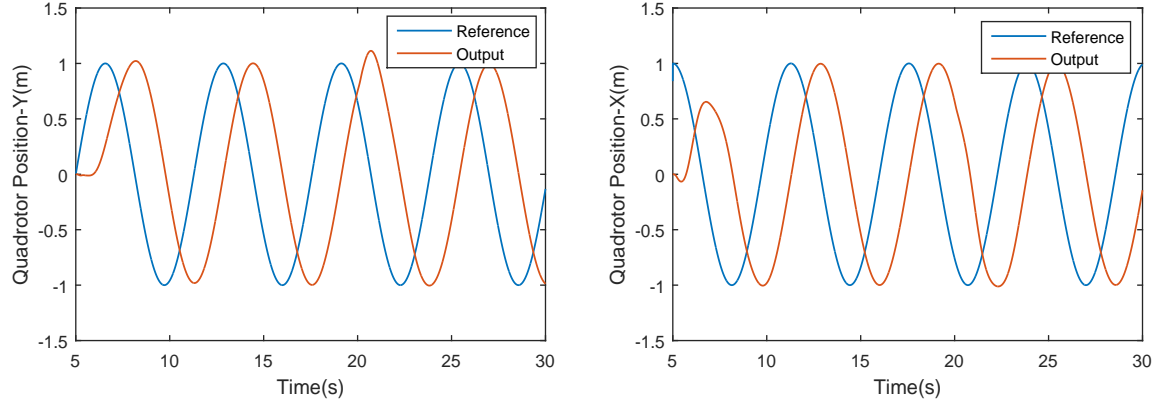
11

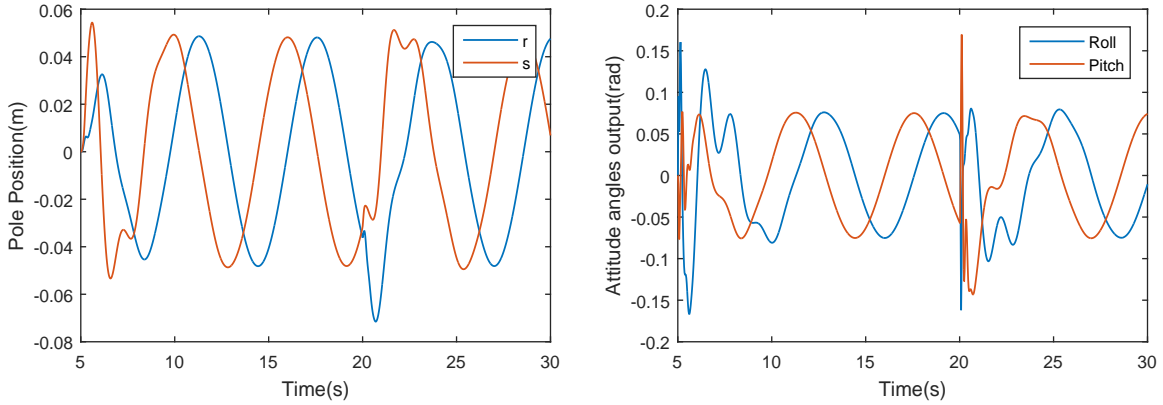Figure 9. Reference trajectory tracking outputs of quadrotor.

Figure 10. Pendulum relative position output when moving circles.

Figure 11. Quadrotor attitude angles output when moving circles.

however the cascade controller can find it itself, i.e. a small circle.

4.2. *Real Experimental Results*

4.2.1. *Real Experiment Setup*

The proposed control method was implemented in Arena Lab at our university with movement tracking system Vicon. The vehicle is a Hummingbird quadrotor from Ascending Technologies with a spherical universal joint attached to the top. A 0.7m long carbon fiber tube with only 13.5g weight is used as the inverted pendulum. The joint base is approximately 7cm above the quadrotor geometrical centre and is 0.39m away from the pendulum mass centre. Fig.12 shows the whole flying pendulum system architecture. The information exchange between the onboard attitude controller and the higher level control algorithms is a pair of wireless Zigbee modules at a frequency of 50Hz. The Vicon system is running at 100Hz and communicates with our conventional desktop via a gigabit ethernet. Fig.13 shows the quadrotor is hovering while balancing the inverted pendulum.

4.2.2. *Experiments and Analysis*

Balancing and hovering case is tested in the first. The quadrotor is hovering without balancing control in the beginning and will switch to the cascaded controller when the relative position (r, s)
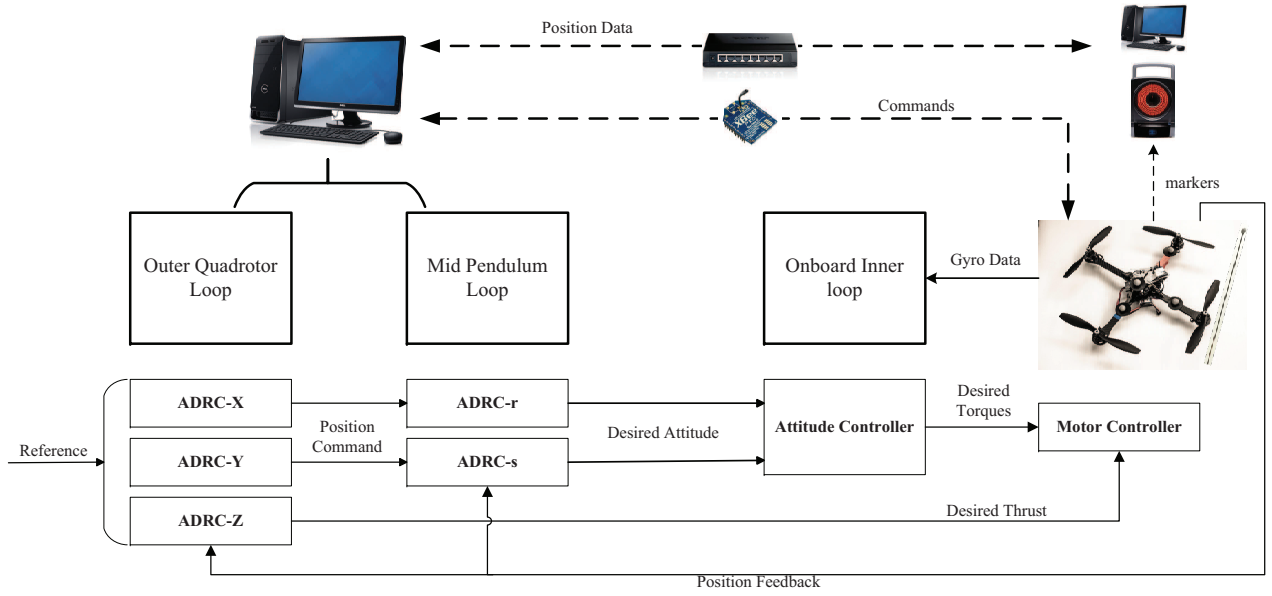
12

Figure 12. Overview of the practical flying pendulum system.

Figure 13. An inverted pendulum on a hovering quadrotor.

is smaller than 3 centimeters. The output results are shown in Fig.14 and Fig.15. The cascaded controller is switched on at approximately $t = 5s$. The pendulum and quadrotor position converge to zero at approximately $t = 13s$. After the system is in stable status, external disturbances are given twice by knocking the pendulum using a stick. From the figures, the two knocks can be found at $t \approx 60s$ and $t \approx 75s$ respectively and the system recovered to the stable state quickly.

Then a reference trajectory is given to the vehicle for path tracking task while balancing the pendulum. The system is switched to the cascaded controller at approximately $t = 7.1s$, starts hovering at $[0\ 0\ 800]^T$, then goes to a setpoint $[1000\ 0\ 800]^T$ at $t = 65.4s$, begins to move circles at $t = 78.8s$ and back to the origin in the end. The quadrotor position output is shown in Fig.16 and Fig.17 depicts the pendulum dynamic in this flight. Additionally, the effect of proposed time-delay compensation algorithm can be tested using same controller but without the compensation to do the same trajectory tracking mission. We find this method works well from outputs comparison
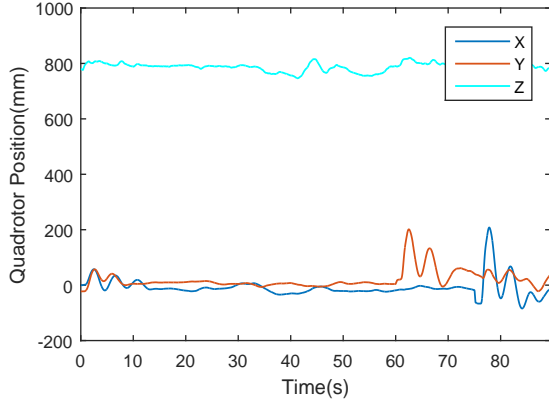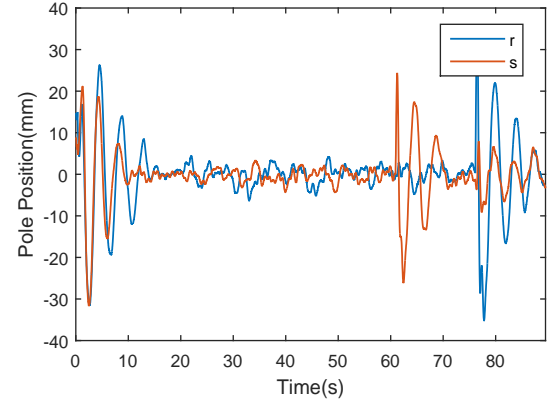
Figure 14. Quadrotor positions in practical hovering.

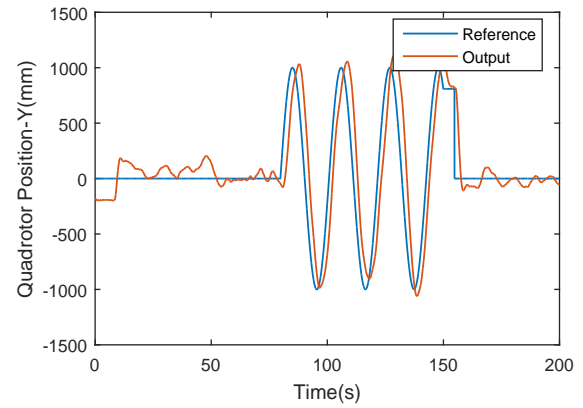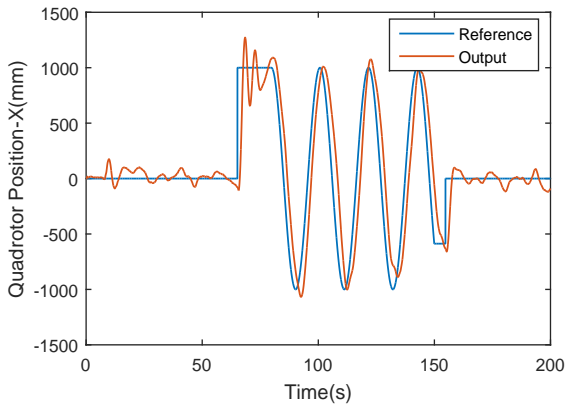Figure 15. Pendulum relative position in practical hovering.

Figure 16. Quadrotor position in practical trajectory tracking.

shown in Fig.18.

5.   Conclusion

In this paper, a cascaded ADRC controller has been proposed for the control problem of a flying inverted pendulum. After the analysis of system dynamics, a cascaded controller with three loops is designed: the inner loop is onboard for attitude control, both the middle loop and the outer loop are in an off-board computer for pendulum position control and vehicle position control respectively. Since the inverted pendulum is very sensitive to the attitude change, a two-time-scale control scheme is deployed: time-varying linear feedback in the fastest inner loop, the desired attitude generation in the middle loop and the slowest trajectory tracking in the outer loop.

Since we find b influence its performance a lot, a changing input gain (b) design from the dynamic equations is adopted instead of a constant gain to make the observer work more efficient. From the performance comparison with a LQR controller, it is clear that the proposed control method can effectively utilize system dynamics and make a better balance between two control objects. Therefore, the trajectory tracking is easily realized and works well under strong disturbances without extra tuning.

Simulation experiments were conducted by using the Matlab Simmechanics platform and an accurate quadrotor CAD file, which offered a 3D animation interface for the easy visualization of
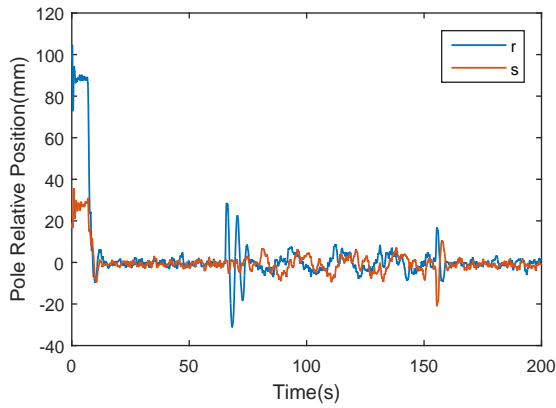
14

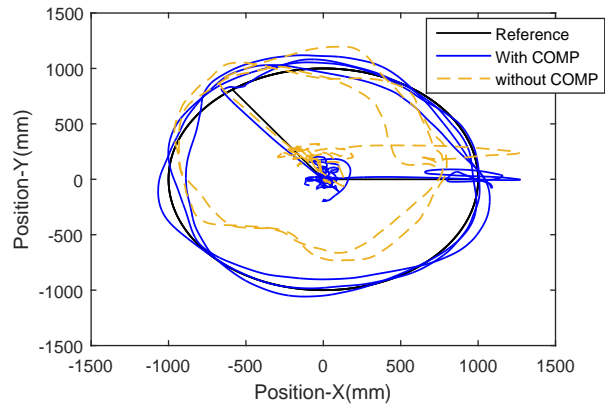Figure 17.   Pendulum relative position in practical trajectory tracking.

Figure 18.   Trajectory comparison between time-delay compensated and uncompensated.

the system dynamics. Then, the proposed control strategy was implemented on a real quadrotor in our robotics lab successfully. To handle time-delay introduced by communications, a compensation algorithm based on tracking differentiator is added between output measurement and states observer to predict the realtime output. The practical flight results show that the controller has fast response and excellent robustness in face of external disturbances.

It is worth noting that this cascaded controller can be easily expanded to other under-actuated robotics systems such as aircrafts and stand-up robots since actually it does not depend on an accurate model of the system. Both the simulation results and the practical flights can be watched clearly in the videos. Our future research work includes smooth trajectories generating as well as some formation control of the multiple quadrotors.

Acknowledgments

References

1. O. Boubaker, "The inverted pendulum benchmark in nonlinear control theory: a survey," *Int J Adv Robotic Sy*, vol. 10, no. 233, 2013.

2. F. Grasser, A. D'Arrigo, S. Colombi, and A. C. Rufer, "Joe: a mobile, inverted pendulum," *Industrial Electronics, IEEE Transactions on*, vol. 49, no. 1, pp. 107–114, 2002.

3. J.-J. Wang, "Simulation studies of inverted pendulum based on pid controllers," *Simulation Modelling Practice and Theory*, vol. 19, no. 1, pp. 440–449, 2011.

4. S. Jung, H.-T. Cho, and T. C. Hsia, "Neural network control for position tracking of a two-axis inverted pendulum system: Experimental studies," *Neural Networks, IEEE Transactions on*, vol. 18, no. 4, pp. 1042–1048, 2007.

5. A. M. Bloch, N. E. Leonard, and J. E. Marsden, "Controlled lagrangians and the stabilization of mechanical systems. i. the first matching theorem," *Automatic Control, IEEE Transactions on*, vol. 45, no. 12, pp. 2253–2270, 2000.

6. G. Cai, J. Dias, and L. Seneviratne, "A survey of small-scale unmanned aerial vehicles: Recent advances and future development trends," *Unmanned Systems*, vol. 2, no. 02, pp. 175–199, 2014.

7. J. P. How, C. Fraser, K. C. Kulling, L. F. Bertuccelli, O. Toupet, L. Brunet, A. Bachrach, and N. Roy, "Increasing autonomy of uavs," *Robotics & Automation Magazine, IEEE*, vol. 16, no. 2, pp. 43–51, 2009.

8. S. Lupashin, M. Hehn, M. W. Mueller, A. P. Schoellig, M. Sherback, and R. DAndrea, "A platform for aerial robotics research and demonstration: The flying machine arena," *Mechatronics*, vol. 24, no. 1, pp. 41–54, 2014.

9. N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav testbed," *Robotics & Automation Magazine, IEEE*, vol. 17, no. 3, pp. 56–65, 2010.

10. M. Hehn and R. D'Andrea, "A flying inverted pendulum," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 763–770, IEEE, 2011.

11. R. Figueroa, A. Faust, P. Cruz, L. Tapia, and R. Fierro, "Reinforcement learning for balancing a flying inverted pendulum," in *Proc. The 11th World Congress on Intelligent Control and Automation*, 2014.

12. C. Raimundez, J. L. Camano, and A. Barreiro, "1tabilizing an inverted spherical pendulum using a scale quad-rotor," in *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2014 IEEE 4th Annual International Conference on*, pp. 111–116, IEEE, 2014.

13. A. Abdessameud, I. Polushin, and A. Tayebi, "Motion coordination of thrust-propelled underactuated vehicles with intermittent and delayed communications," *Systems & Control Letters*, vol. 79, pp. 15–22, 2015.

14. L. Guo and S. Cao, "Anti-disturbance control theory for systems with multiple disturbances: A survey," *ISA transactions*, vol. 53, no. 4, pp. 846–849, 2014.

15. K. Ohishi, M. Nakao, K. Ohnishi, and K. Miyachi, "Microprocessor-controlled dc motor for load-insensitive position servo system," *IEEE Transactions on Industrial Electronics*, vol. 1, no. IE-34, pp. 44–49, 1987.

16. S. Li and J. Yang, "Robust autopilot design for bank-to-turn missiles using disturbance observers," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 49, no. 1, pp. 558–579, 2013.

17. D. Xing, J. Su, Y. Liu, and J. Zhong, "Robust approach for humanoid joint control based on a disturbance observer," *Control Theory & Applications, IET*, vol. 5, no. 14, pp. 1630–1636, 2011.

18. K. K. Tan, T. H. Lee, H. F. Dou, S. J. Chin, and S. Zhao, "Precision motion control with disturbance observer for pulsewidth-modulated-driven permanent-magnet linear motors," *Magnetics, IEEE Transactions on*, vol. 39, no. 3, pp. 1813–1818, 2003.

19. Q. Zheng and Z. Gao, "On practical applications of active disturbance rejection control," in *Proceedings of the 2010 Chinese Control Conference*, pp. 6095–6100, 2010.

20. M. Ramirez-Neria, H. Sira-Ramirez, R. Garrido-Moctezuma, and A. Luviano-Juárez, "Linear active disturbance rejection control of underactuated systems: the case of the furuta pendulum," *ISA transactions*, vol. 53, no. 4, pp. 920–928, 2014.

21. C. Zhang and J. Zhu, "On stabilization and disturbance rejection for the inverted pendulum," in *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pp. 3750–3754, IEEE, 2014.

22. S. Esteban, F. Gordillo, and J. Aracil, "Three-time scale singular perturbation control and stability analysis for an autonomous helicopter on a platform," *International Journal of Robust and Nonlinear Control*, vol. 23, no. 12, pp. 1360–1392, 2013.

23. G. D. Wood and D. C. Kennedy, "Simulating mechanical systems in simulink with simmechanics," *The Mathworks Report*, 2003.

24. D. Gurdan, J. Stumpf, M. Achtelik, K.-M. Doth, G. Hirzinger, and D. Rus, "Energy-efficient autonomous four-rotor flying robot controlled at 1 khz," in *Robotics and Automation, 2007*

IEEE International Conference on, pp. 361–366, IEEE, 2007.

25. F. Kendoul, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems," *Journal of Field Robotics*, vol. 29, no. 2, pp. 315–378, 2012.

26. Z. Gao, "Scaling and bandwidth-parameterization based controller tuning," in *Proceedings of the American Control Conference*, vol. 6, pp. 4989–4996, 2006.

27. I. Sa and P. Corke, "System identification, estimation and control for a cost effective open-source quadcopter," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 2202–2209, IEEE, 2012.

28. A. N. K. Nasir, M. A. Ahmad, and M. F. Rahmat, "Performance comparison between lqr and pid controllers for an inverted pendulum system," in *INTERNATIONAL CONFERENCE ON POWER CONTROL AND OPTIMIZATION: Innovation in Power Control for Optimal Industry*, vol. 1052, pp. 124–128, AIP Publishing, 2008.