Modeling, Identification, and Control of Non-minimum Phase Dynamics of Bi-copter UAVs

Yihang Li[1],Youming Qin[1],Wei Xu[1],Fu Zhang[1]

***Abstract*— Bi-copter UAVs have emerged as more agile and energetically efficient flying platforms than quadrotor UAVs. Yet, they are more challenging to control due to the use of servo motors leading to non-minimum phase roll dynamics. In this paper, we systematically investigate the modeling, identification, and control of such non-minimum phase dynamics in an integrated way. We start with deriving the first principle model, from which the cause of the non-minimum phase is clearly seen. Parameters of the derived model are identified via experiments and subsequently used to design and analyze a high-gain robust flight controller via H_∞ loop shaping techniques. The designed controller is implemented on actual systems and compared against a well-tuned PID controller. Extensive flight tests are conducted to verify the controller's effectiveness and the effect of the system's non-minimum phase nature.**

I. INTRODUCTION

In the last decades, quadrotor UAVs have attracted intensive research and commercialization interests due to their mechanical simplicity, agility, and small form factor [1]. Yet, many indoor tasks (e.g., exploration, disaster relief, constructions site monitoring & mapping) require the UAV to carry significant payload (e.g., LiDAR, IR cameras, onboard computer) and to fly through tight spaces (e.g., narrow doors, windows, hallway) for extended time. These three requirements (i.e., small size, high payload capacity, and low power consumption) are conflicting with each other and are fundamentally limited by propeller aerodynamics [2]: reducing the size or increasing the payload leads to high power consumption.

Bi-copters have emerged as an excellent platform suitable for indoor applications [3]. Shown in Fig. 1, a bi-copter utilizes two tandem rotors and hence reduces the size by half (w.r.t. a quadrotor) when moving in that direction. More formally, [3] compared the power consumption of bi-copter configuration with other common multi-rotor configurations and showed that bi-copters are more efficient than the rests. In particular, it is 30% more power or size efficient than quadrotors of the same weight. Concurrent developments in commercial sectors also proved the energy superiority of such bi-copter configuration [4].

Despite their superiority in power efficiency and size, bi-copters are very challenging to control, especially the roll dynamics due to its non-minimum phase nature. Prior work has been conducted to address these challenges. Gonçalves *et al.* [5] presented the bi-copter concept and derived the mathematical model. However, the derived model did not consider the non-minimum phase in the roll dynamics. In [6], the authors conducted more systematical work on such a bi-copter UAV from development, modeling, control to actual flight tests. A fully nonlinear model was derived, and a feedforward feedback controller was proposed and verified by experiments. Again, the presented model neglected the non-minimum phase nature in the roll dynamics, and the presented controller was more focused on the position controller. This was also the case with our prior work [3], where a cascaded PID controller was used to control the position, attitude, and angular velocity without carefully modeling the underlying roll dynamics. Other prior work, such as the model predictive control in [7], PID control in [8], and N-PID controller in [9] all neglected the non-minimum phase. The performance of these controllers was highly dependent on manual tuning and most often not optimal. More importantly, the stability margin of these tuned controllers is totally unknown when model uncertainties occur.

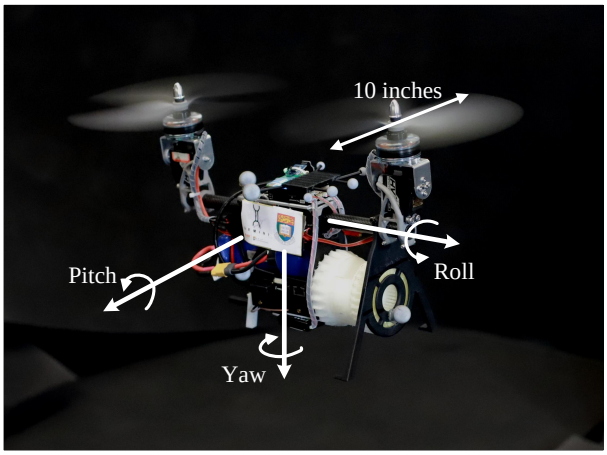

Fig. 1. Gemini: a compact yet efficient bi-copter UAV for indoor mapping. With weight up to 1.8 kg and width down to 10 inches, the UAV is able to hover for 13 minutes (see [3]) and to fly through tight spaces in indoor environments.

In this paper, we formally investigate the non-minimum phase nature of the bi-copter roll dynamics: modeling, identification, and control. We derive the first principle model and explain the cause of the non-minimum phase nature and its effect on system response. Experiments are conducted to identify the model parameters. Results show that the model with identified parameters matches very closely to the system response in the frequency domain. Then, we proceed to design a high-gain robust controller using H_∞ loop-shaping technique and compare its performance against a well-tuned PID controller. The comparison is conducted via

[1]Authors are with the Mechatronics and Robotic Systems (MaRS) Laboratory, Department of Mechanical Engineering, The University of Hong Kong, China. yhangli, qym96, xuweii, fuzhang@hku.hk

both theoretical analysis and flight experiments in normal and extreme flight conditions (e.g., drastic external disturbances and center of mass offset). Results show that the H_∞ controller achieves more disturbance attenuation and higher robustness margin while the PID controller crashed in one of the extreme tests.

This paper is organized as follows: Section II derives the first principle model of the system roll dynamics and analyzes its non-minimum phase nature. Section III presents the model identification details. Based on the identified model, the controller is designed in Section IV using H_∞ loop shaping techniques and is compared to a well-tuned PID controller. Section V shows various flight experiments, validating the performance and the stability margin of the designed controllers. Finally, Section VI draws conclusions and discusses future work.

II. SYSTEM DYNAMICS

In modeling the roll dynamics, we only retain those related physical variables to ease the presentation. Readers may refer to our prior work [3] for detailed explanation of the actuation principle and the full system dynamics. Shown in Fig. 2, the UAV tilts the two tandem rotors collectively by angle δ such that their total thrust T offsets from the center of mass and thereby produces a rolling torque τ_T, as follows:

$$\tau_T = TH\sin\delta \tag{1}$$

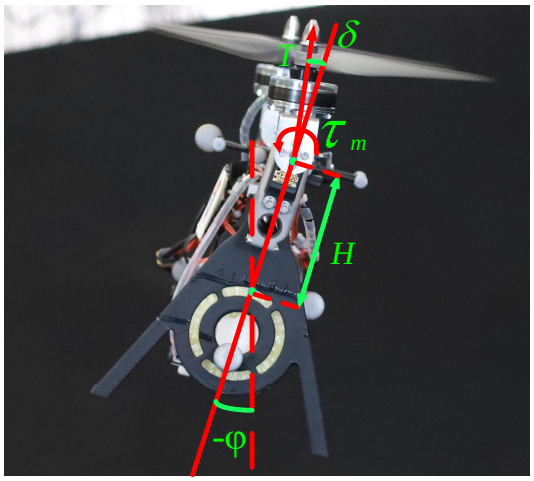

Fig. 2. Roll dynamics of our bi-copter UAV.

Besides the thrust-induced torque, the servo rotor also exerts a torque (termed as the "reactional torque τ_m") to the stator (hence the UAV body). Therefore, the UAV body dynamics is:

$$I_b\ddot{\varphi} = TH\sin\delta - \tau_m \tag{2}$$

where I_b is the moment of inertia of the UAV on roll axis. The same motor torque τ_m also applies to the servo rotor in the opposite direction according to the Newton's third law, driving the rotor angles (w.r.t. an inertial frame). The resulting rotor dynamics are:

$$I_m(\ddot{\varphi} + \ddot{\delta}) = \tau_m \tag{3}$$

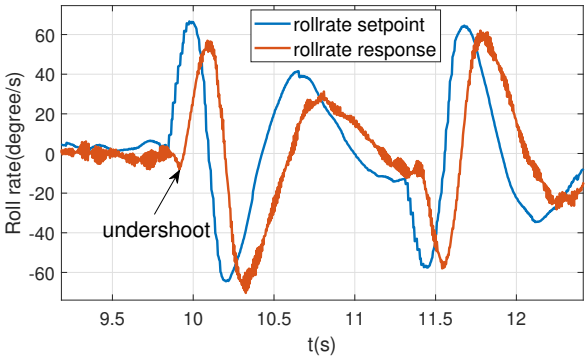

Fig. 3. Undershoot in roll rate response.

Combining (2) and (3) leads to the roll dynamics from the servo angle δ to the roll angle φ:

$$I_b\ddot{\varphi} = TH\sin\delta - I_m(\ddot{\varphi} + \ddot{\delta}) \tag{4}$$

In practice, the entire flight controller is usually a nested control structure where the internal controller is on roll rate [3]. Then linearizing the dynamics around the hovering state where $T = G = mg$ and $\sin(\delta) = \delta$, the roll rate dynamics is:

$$P_r(s) = \frac{GH - I_m s^2}{(I_b + I_m)s} \tag{5}$$

Equation (5) implies an unstable zero $z = \sqrt{GH/I_m}$, which indicates a non-minimum phase system [10]. Its effects are twofold: (1) the unstable zero will cancel the phase increment contributed by the stable zero $z = -\sqrt{GH/I_m}$, leading to a constant zero phase in the numerator. The cancellation in phase increment deteriorates the phase margin at the crossover frequency and fundamentally limits the achievable bandwidth of a linear controller [11]. More specifically, as shown in [12], the unstable zero puts an upper bound to the controller bandwidth which will be verified later in Section IV; (2) the unstable zero will cause a significant undershoot in the system response, shown in Fig. 3. While [10] gives a theoretical proof for a general non-minimum phase system, here we specifically explain the physical cause of the bi-copter UAV: when a positive roll rate is expected, the servo rotor should tilt positively (i.e., positive δ), so does the servo torque τ_m. However, this servo torque τ_m applies oppositely to the UAV body, driving the body roll rate to the negative direction. Since the servo torque always comes ahead of the tilting angle (i.e., it takes time to build up angle δ when τ_m applies) and diminishes as the servo angle converges, its effect (i.e., driving the roll rate oppositely) is always ahead of the thrust (which drives the roll rate closer to the expected value) and is gradually dominated by the latter, causing an undershoot in the response. Moreover, since the servo torque (and its adverse effect) always comes with the tilting (i.e., the roll rate actuation), this undershoot is an inherent phenomenon regardless of the controller being used. The undershoot always delays the response from the reference, explaining the limitation on controller bandwidth.

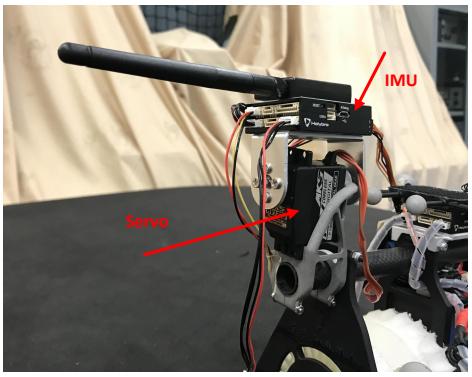

Fig. 4. Setup for identifying the servo model.

III. MODEL IDENTIFICATION

With the model in (5), this section presents the identification of the model parameters, namely, the location of zeros and poles of (5). Since the model in (5) only models the dynamics from the servo angle to the body roll rate while in practice, the command to the servo is pulse width modulation (PWM) signals, there is an additional dynamics within the servo. We therefore employ two identification steps: the first on the servo and the second on the entire system.

A. Servo model identification

To identify the servo model, we install a flight controller on the servo to provide angular rate measurements (see Fig. 4). A chirp signal with frequency range $[0.8Hz, 20Hz]$ is used to excite the servo dynamics [13]. Then the Spectral Analysis with Frequency-Dependent Resolution(SPAFDR) method [14] is applied to the collected data to obtain the gain and phase delay of the system dynamics in the frequency domain, leading to Fig. 5.

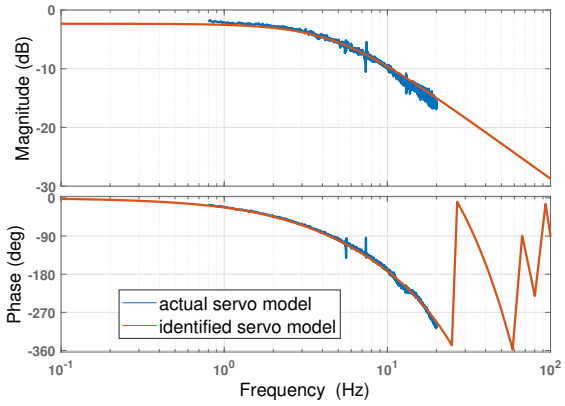

Fig. 5. Frequency response of servo dynamics and its fitted model.

Two spikes appearing around $6Hz$ and $7Hz$ are due to loss of data in the logging process, thus should be viewed as noises. The data in Fig. 5 suggests that the servo dynamics can be accurately modeled as a first-order system with a pure time delay:

$$P_m(s) = e^{-0.03s}\frac{0.764}{0.0332s+1} \tag{6}$$

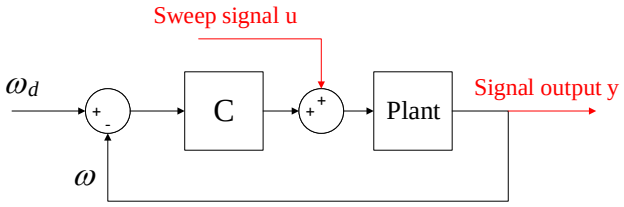

Fig. 6. The location of sweep signal in control loop.

B. Roll rate model identification

The identification of the entire roll rate model is similar to that of the servo, except that a baseline feedback controller (e.g., PID) is used to stabilize the roll dynamics such that the drone will not crash during the data collection, and that the excitation signal is injected at the controller output (see Fig. 6). The output (i.e., the roll rate) is measured by the onboard IMU. The obtained frequency response is shown in Fig. 7.

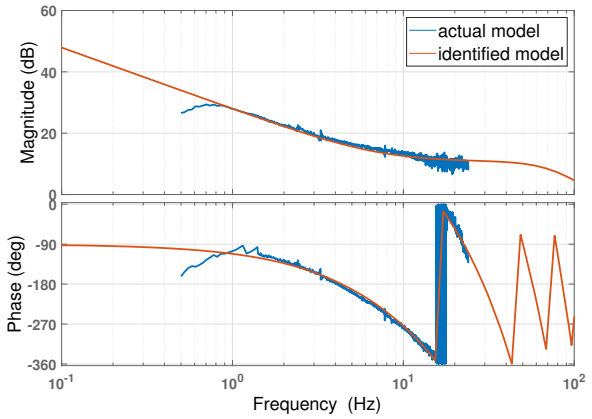

Fig. 7. Roll rate frequency response and its fitted model.

We use a parametric model to fit the measured frequency response. Recall that the entire system dynamic model is the concatenation of the servo model (6) and the first principle model (5). Additionally, the IMU data is pre-filtered by a second order Butterworth filter with $75Hz$ cutoff frequency. Hence the lumped model is

$$P(s) = \frac{Ae^{-T_d s}}{\tau s+1}\cdot\frac{GH/I_m - s^2}{(I_b/I_m+1)s}\cdot\frac{2.221\times10^5}{s^2+666.4s+2.221\times10^5} \tag{7}$$

where the parameters to be identified are the DC gain $A/(I_b/I_m+1)$, time constant τ, pure delay T_d and zero $\sqrt{GH/I_m}$. It should be noticed that the servo parameters (i.e., A, τ, T_d) are not fixed at values identified in the previous section because the servo dynamics will slightly change at different load.

We optimize the parameters in (7) via a nonlinear least square *nlinfit* in MATLAB. The resulting system model is

$$P(s) = \frac{0.0887e^{-0.032s}}{0.0256s+1}\cdot\frac{42^2-s^2}{s}\cdot\frac{2.221\times10^5}{s^2+666.4s+2.221\times10^5} \tag{8}$$

It can be seen that the identified servo delay T_d is very close to that in (6), while the time constant changes slightly due to the change of load. Other parameters are in the expected range implied by our physical system. Finally, shown in Fig. 7, the identified parametric model fits well with the actual data, validating our derived first principle model in Section II.

IV. CONTROLLER DESIGN

The identified frequency-domain model (8) is used for controller design. We use a H_∞ loop shaping technique to design a high-gain robust roll rate controller. H_∞ loop shaping is a sensible method to design flight controller [15] and has been widely used in manned and unmanned aircraft, such as an experimental Harrier aircraft [16], [17], the Bell airborne simulator [18], and the Yamaha robotic helicopter [19]. In this section, we go through the key steps of H_∞ loop shaping and focus on its application to our system.

Step 1: Choose of W_1 and W_2

To obtain good attenuation for low frequency disturbances and high frequency measurement noise, it is necessary to shape the plant model P with frequency dependent weights W_1 and W_2. As in [19], W_1 is usually a proportional-integral filter to ensure sufficient disturbance attenuation and W_2 is a low-pass filter with a lead compensator to ensure sufficient noise attenuation. In our case,

$$W_1 = \frac{0.03511(s+0.3419)}{s} \tag{9}$$

$$W_2 = \frac{307.5(s+3.816)}{(s+11.35)(s+103.4)} \tag{10}$$

W_1 and W_2 push the gain crossover frequency of the weighted plant $P_s = W_2 P W_1$ as high as possible while not destabilizing the system.

Step 2: H_∞ loop shaping

The standard H_∞ block diagram is depicted in Fig. 8. With the W_1 and W_2, H_∞ loop shaping method minimizes the H_∞ norm of the transfer function from disturbance $[w_1, w_2]^T$ to errors $[z_1, z_2]^T$ over all stable K_∞ (see Fig. 8). More formally,

$$\min_{\text{stable } K_\infty} \left\| \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \to \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \right\|_\infty = \min_{\text{stable } K_\infty} \left\| \begin{pmatrix} K_\infty \\ I \end{pmatrix} (I - P_s K_\infty)^{-1} \begin{pmatrix} I & P_s \end{pmatrix} \right\|_\infty = \gamma \tag{11}$$

where $P_s = W_2 P W_1$ denotes weighted plant. The optimal value γ indicates the performance degradation and the robustness level after H_∞ loop shaping. Typically, $\gamma < 3$ implies small performance degradation and good stability margin [19].

We use the MATLAB function *ncfsyn* to solve (11). To avoid the pure delay in P, which usually causes difficulties in the solver, we approximate it by a three order system using the MATLAB function *pade*. Then solving (11) with *ncfsyn* leads to $\gamma = 1.748$ and an optimal K_∞. This value far exceeds the required threshold, so we manually increase the controller gain (i.e., gain boosting) so that the controller bandwidth is maximized. Denoting the gain boosted controller as K_∞, the whole controller should then be $W_1 K_\infty W_2$.

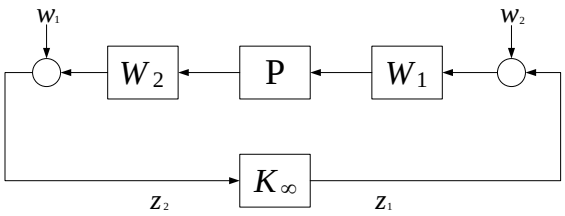

Fig. 8. Standard H_∞ loop shaping block diagram.

Step 3: Controller order reduction

A problem of the H_∞ controller is that its order is usually very high. In our case, $W_1 K_\infty W_2$ is of 12 orders. Such a high order controller is computationally expensive for embedded implementation. We therefore conduct a model reduction using the implicit balancing techniques [20] implemented in MATLAB. The reduced controller is only 5 orders, as follow:

$$C = \frac{24.02(s+54.01)(s+0.2474)(s^2+139.3s+2.624\times10^4)}{s(s^2+24.14s+8544)(s^2+232.4s+7.041\times10^4)} \tag{12}$$

As shown in Fig. 9, the reduced controller is very close to the original H_∞ controller. With both the gain boosting and the order reduction, the final value of γ is 2.862, well below the required threshold.

Step 4: Robustness and performance analysis

Then we conduct frequency analysis of our final controller (12) and compare it with a well-tuned PID controller. Fig. 10 shows the loop transfer function $L = PC$. For the H_∞ controller, the phase margin is 65.8° at gain crossover frequency $1.47Hz$ and gain margin is $7.25dB$. While for the PID, the phase margin is 64.9° at $1.58Hz$ and gain margin is $6.90dB$. It can be seen that both crossover frequencies are lower than the system unstable zero at $6.68Hz$. The stability margins are comparable among the two controllers. We further consider large nonparametric model uncertainties: $P_a = P(1+\Delta)$, where Δ is the model uncertainty with arbitrary phases. As shown in [12], for the system $P_a C$ to be stable, Δ needs to

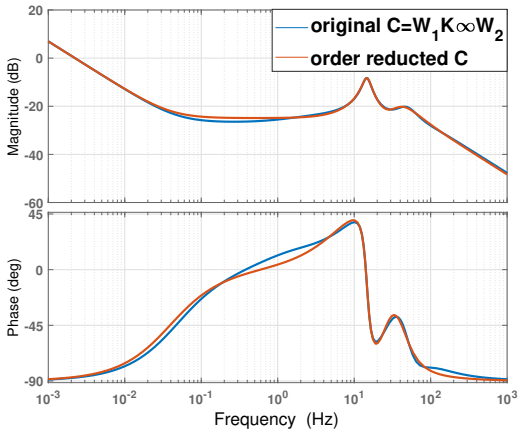

Fig. 9. Comparison of original $W_1 K_\infty W_2$ and order reduced C.

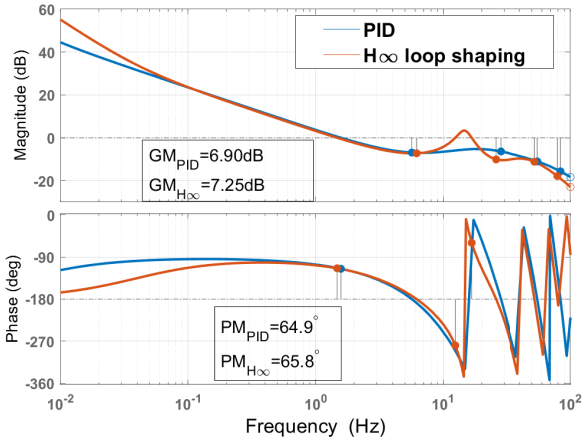

Fig. 10. The loop transfer function of H_∞ controller and PID.

meet

$$|\Delta| < \left|\frac{1+PC}{PC}\right| = \left|\frac{1}{T}\right| \tag{13}$$

where $T = PC/(1+PC)$ is the complementary sensitivity function. (13) implies smaller T allows larger model uncertainty (i.e., more stability margin). Fig. 11 shows the complementary sensitivity function of our H_∞ controller and the PID. It is seen that the PID has a notable peak around $27.8Hz$, leading to poor robustness around that frequency. On the other hands, our H_∞ controller has no such peak. Moreover, the faster decaying rate at high frequency range shows a better high frequency noise attenuation using the H_∞ controller. Finally, as shown by the sensitivity function in Fig. 12, H_∞ loop shaping controller also shows a better (i.e., more than $20dB$) low frequency disturbance attenuation.

Step 5 : Controller implementation

As discussed in [15], the H_∞ loop shaped controller $W_1K_\infty W_2$ has many different implementation forms. It could be placed on the forward path to reduce the rising time, on the feedback path to reduce overshoot, or implemented in an observer form which compromises between these two. The observer form is used on a YAMAHA helicopter in [19]. Since our bi-copter requires more agility and faster response so that the other nested outer loop controllers (e.g., attitude controller, velocity controller, and position controller as in [3]) can achieve higher gains, we place H_∞ controller in the forward path as shown in Fig. 13.

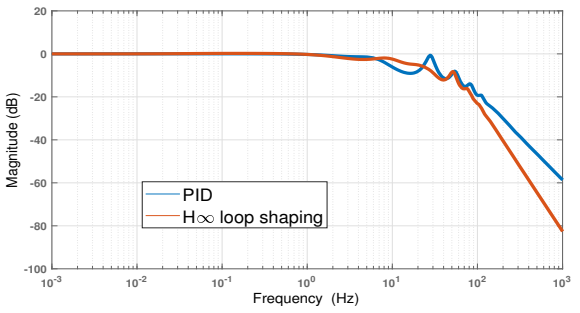

Fig. 11. Complementary sensitivity $T = PC/(1+PC)$.

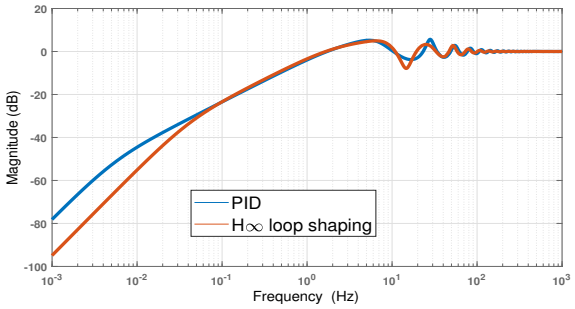

Fig. 12. Sensitivity $S = 1/(1+PC)$.

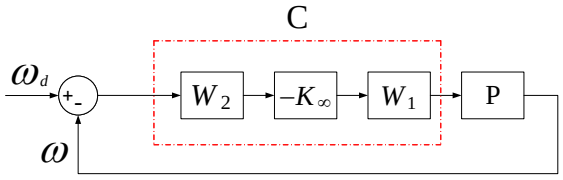

Fig. 13. Control structure of roll rate.

Finally, the controller is discretized by the Tustin method [21] at a sampling rate of $250Hz$. The discretized controller is implemented on the Pixhawk flight controller [22]. Other considerations, such as integrator windup has also been resolved in the implementation.

V. EXPERIMENTS

To verify the performance and robustness of the designed controllers, extensive experiments are conducted on our bi-copter UAV prototype. Besides the roll rate controller, the onboard flight controller also runs other angular rate controllers, attitude controllers, and position controllers, so the UAV's full position is under active control. Readers may refer to our prior work [3] for details of the prototype development and the full control architecture. We conducted three groups of experiments as below. The experiment video is also available at `http://youtu.be/jSepJtBDNYs`.

A. Normal flight

The first test is conducted on our nominal bi-copter prototype. After the UAV takes off and hovers at the target position, a test pilot commands different lateral velocity via a remote controller to excite the roll dynamics. The roll rate response of the two controllers is shown in Fig. 14. Roll rate commands are tracked well with both the H_∞ and PID controller. Nevertheless, the H_∞ controller has better noise attenuation. The PID controller leads to noisy response around $125Hz$ with $15^\circ/s$ amplitude due to the IMU noise. This difference in noise attenuation agrees with the complementary sensitivity function in Fig. 11.

B. Hanging disturbance

In our second test, we hang a $1m$ pendulum with a $134g$ mass attached to the end as shown in Fig. 15. After the UAV takes off the ground and hovers at the target position,

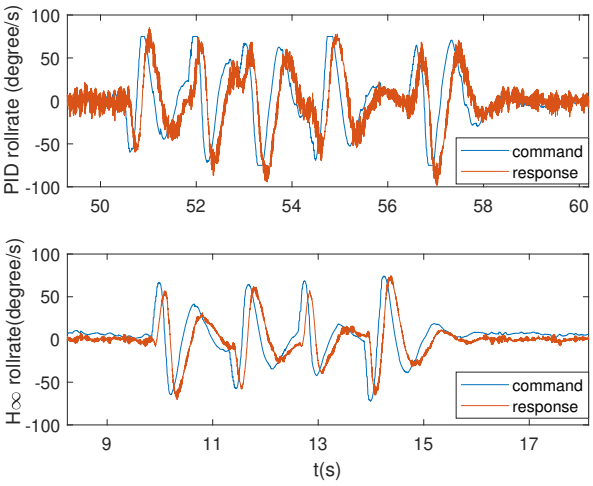

Fig. 14. Roll rate response in normal condition.

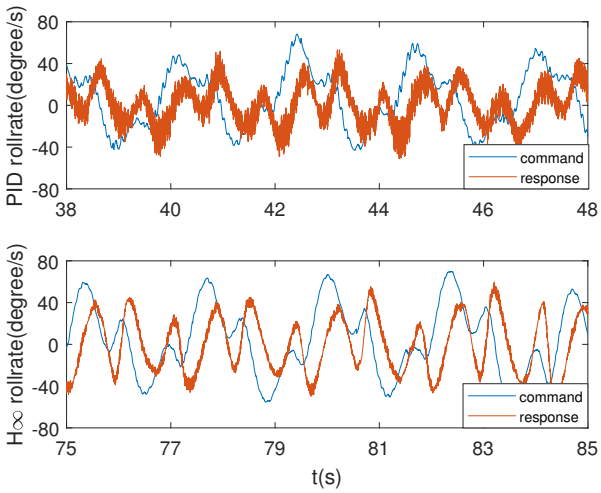

Fig. 16. Roll rate response when hanging a pendulum.

the pendulum is lifted to a height and then released to swing at $0.5Hz$, causing a disturbance at that frequency. Inspecting the data in Fig. 16, we find that the roll rate command is mainly concentrated on $0.5Hz$ and $1Hz$ due to the reaction of the outer loop position controller and the possible excitation of nonlinear effect. Under such roll rate command, the H_∞ controller and the PID controller achieve comparable tracking performance, which agrees with the sensitivity function in the previous section.

C. Center of mass offset

We finally study the controller stability with both model mismatch and disturbances. Shown in Fig. 17, we attach a water bottle of $324g$ on one side of the UAV, causing a constant disturbance and a mismatch in the system model (e.g., inertia is changed). Then the UAV is commanded to hover at a prescribed position right after taking off. The response of the H_∞ controller is shown in Fig. 18, the roll rate response converges nicely to the reference. After convergence, the tracking performance is almost the same as the normal condition (see Fig. 14). This verifies that our

Fig. 17. The UAV carries a $324g$ water bottle on one side leading to a constant disturbance torque and model mismatch.

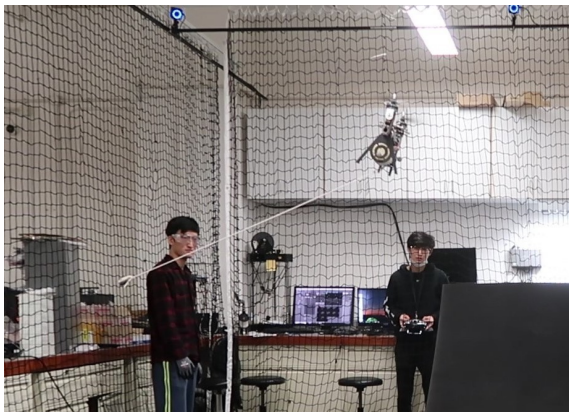

Fig. 15. The UAV is attached by a pendulum ($134g$ mass) leading to a sinusoidal disturbance torque.

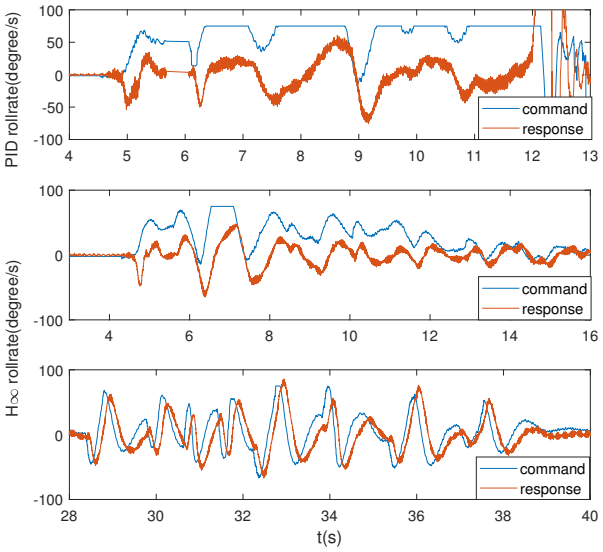

Fig. 18. Roll rate response when carrying a water bottle. Top: PID controller; Middle: H_∞ controller before convergence; Bottom: H_∞ controller after convergence.

designed controller can effectively compensate the constant disturbance and is robust to model uncertainties. In contrast, the PID controller fails to track the reference and crashes at $t = 12s$. When inspecting the data in Fig. 18, we can see that the gap between the reference and the response is more or less constant (slightly increasing), meaning that the crash can be caused by either the slow convergence or the insufficient stability margin of the PID controller. Either way, it agrees to the theoretical analysis conducted in the previous section.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we precisely derive and identify the model of a bi-copter UAV roll dynamics, which is a non-minimum phase system. Based on the identified model, a robust and well-performed controller is designed using H_∞ loop shaping techniques. Robustness is analyzed in the frequency domain and compared with a well-tuned PID controller. Finally, flight tests are conducted, which verifies that the H_∞ loop shaping controller has better performance on noise and disturbance attenuation. In the presence of large model uncertainty, the H_∞ also exhibits better robustness and can stabilize the system well.

Future work will focus on the outer loop controller design, especially the lateral position/velocity control, to achieve precise hovering with the bi-copter UAV.

REFERENCES

[1] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics and Automation magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[2] G. J. Leishman, *Principles of helicopter aerodynamics with CD extra*. Cambridge university press, 2006.

[3] Y. Qin, W. Xu, A. Lee, and F. Zhang, "Gemini: A compact yet efficient bi-copter uav for indoor applications," *IEEE Robotics and Automation Letters*, pp. 1–1, 2020.

[4] Z. Z. Robotics, "Bizarre "bicopter" drone uses two tilting rotors for 50-minute flights," Dec 2019. [Online]. Available: https://newatlas.com/drones/v-coptr-falcon-drone-tilting-50/

[5] F. Gonçalves, J. Bodanese, R. Donadel, G. Raffo, J. Normey-Rico, and L. Becker, "Small scale uav with birotor configuration," in *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2013, pp. 761–768.

[6] A. Sanchez, J. Escareno, O. Garcia, and R. Lozano, "Autonomous hovering of a noncyclic tiltrotor uav: Modeling, control and implementation," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 803–808, 2008.

[7] C. Papachristos, K. Alexis, G. Nikolakopoulos, and A. Tzes, "Model predictive attitude control of an unmanned tilt-rotor aircraft," in *2011 IEEE International Symposium on Industrial Electronics*. IEEE, 2011, pp. 922–927.

[8] Q. Zhang, Z. Liu, J. Zhao, and S. Zhang, "Modeling and attitude control of bi-copter," in *2016 IEEE International Conference on Aircraft Utility Systems (AUS)*. IEEE, 2016, pp. 172–176.

[9] L. Hrečko, J. Slačka, and M. Halás, "Bicopter stabilization based on imu sensors," in *2015 20th International Conference on Process Control (PC)*. IEEE, 2015, pp. 192–197.

[10] M. Vidyasagar, "On undershoot and nonminimum phase zeros," *IEEE Transactions on Automatic Control*, vol. 31, no. 5, pp. 440–440, 1986.

[11] K. J. Åström, "Fundamental limitations of control system performance," in *Communications, Computation, Control, and Signal Processing*. Springer, 1997, pp. 355–363.

[12] K. J. Åström, "Model uncertainty and robust control," *Lecture Notes on Iterative Identification and Control Design*, pp. 63–100, 2000.

[13] A. Novak, L. Simon, F. Kadlec, and P. Lotton, "Nonlinear system identification using exponential swept-sine signal," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 8, pp. 2220–2229, 2009.

[14] L. Ljung, *Version 6 of the system identification toolbox*. Linköping University Electronic Press, 2003.

[15] G. Papageorgiou and K. Glover, "h_∞ loop shaping: Why is it a sensible procedure for designing robust flight controllers?" in *1999 Guidance, Navigation, and Control Conference and Exhibit*, 1999, pp. 1–14.

[16] R. A. Hyde, *H_∞ aerospace control design: a VSTOL flight application*. Springer Science & Business Media, 2013.

[17] G. Papageorgiou, K. Glover, G. D'Mello, and Y. Patel, "Taking robust lpv control into flight on the vaac harrier," in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, vol. 5. IEEE, 2000, pp. 4558–4564.

[18] I. Postlethwaite, A. Smerlas, D. Walker, A. Gubbels, S. Baillie, M. Strange, and J. Howitt, "h_∞ control of the nrc bell 205 fly-by-wire helicopter," *Journal of the American Helicopter society*, vol. 44, no. 4, pp. 276–284, 1999.

[19] M. L. Civita, G. Papageorgiou, W. C. Messner, and T. Kanade, "Design and flight testing of an h00 controller for a robotic helicopter," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 2, pp. 485–494, 2006.

[20] A. Varga, "Balancing free square-root algorithm for computing singular perturbation approximations," in *[1991] Proceedings of the 30th IEEE Conference on Decision and Control*. IEEE, 1991, pp. 1062–1065.

[21] A. V. Oppenheim and R. W. Schafe, "Discrete-time signal processing (third section)," 2011.

[22] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "Pixhawk: A system for autonomous flight using onboard computer vision," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 2992–2997.