

A Hammerstein-Weiner Modification of Adaptive Autopilot for Parameter Drift Mitigation with Experimental Results

Yin Yong Chee, Parham Oveissi, Siyuan Shao, Joonghyun Lee,
Juan A. Paredes, Dennis S. Bernstein, Ankit Goel

Abstract—A crucial challenge in the safe operation of adaptive controllers is the problem of parameter drift, where an underlying optimization problem, if ill-conditioned, may lead to parameter drift. This paper presents a Wiener adaptive autopilot for multicopters to mitigate instabilities caused by adaptive parameter drift and presents simulation and experimental results to validate the modified autopilot. The modified adaptive controller is obtained by including a static nonlinearity in the adaptive loop, updated by the retrospective cost adaptive control algorithm. It is shown in simulation and physical test experiments that the adaptive autopilot with proposed modifications can continually improve the fixed-gain autopilot as well as prevent the drift of the adaptive parameters, thus improving the robustness of the adaptive autopilot.

I. INTRODUCTION

Why do parameters drift? In practice, error is small after convergence, which causes the internal optimization problem to become ill conditioned due to accumulative lack of persistency. Parameters therefore may drift due to nonuniqueness of the minimizing solution.

Multicopters have found significant success in several engineering applications such as precision agriculture [1], environmental survey [2], [3], construction management [4] and load transportation [5]. However, for several reasons, including nonlinear and uncertain dynamics, unknown and uncertain operating environments, and ease of reconfigurability, control of multicopters remains a challenging engineering problem.

Several control techniques have been applied to construct stabilizing controllers for multicopters [6]–[9]. However, these techniques often require an accurate plant model and, thus, are susceptible to unmodeled dynamics and physical model parameter uncertainty [10], [11]. Several adaptive control techniques have been applied to address the problem of unmodeled, unknown, and uncertain dynamics such as model reference adaptive control [12], [13], L1 adaptive control [14], and adaptive sliding mode control [15]. In our previous work, we developed an adaptive autopilot based on the retrospective cost adaptive control (RCAC) algorithm

This research was supported in part by the Office of Naval Research under grant N00014-19-1-2273.

Yin Yong Chee, Siyuan Shao, Juan Augusto Paredes, and Dennis S. Bernstein are with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109. [cyinyong](mailto:cyinyong@umich.edu), [shaosy](mailto:shaosy@umich.edu), [jparedes](mailto:jparedes@umich.edu), dsbaero@umich.edu

Joonghyun Lee was with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109. joonghle@umich.edu

Parham Oveissi and Ankit Goel are with the Department of Mechanical Engineering, University of Maryland, Baltimore County, MD 21250. [parhamo1](mailto:parhamo1@umbc.edu), ankgoel@umbc.edu

[16], [17]. RCAC is described in [18], and is extended to digital PID control in [19].

When the sensor measurements are affected by an unknown bounded disturbance of sufficient amplitude, the parameters updated by the adaptation laws of adaptive controllers may diverge and yield an unstable controller behavior. This phenomenon is called *parameter drift* [20], [21]. To remedy the parameter drift problem in adaptive control techniques, several extensions have been proposed, such as deadzone nonlinearities, projection operator, and σ -modification and σ -modification techniques [22]–[24], controller output filtering [25], [26], and controller output averaging [27], [28]. As expected, these modifications entail a tradeoff between robustness to measurement disturbances and tracking performance.

The present paper extends the adaptive autopilot presented in [16]. In this work, it is shown that, in some circumstances, the controller gains optimized by the RCAC algorithm may diverge, eventually leading to the failure of the control system. To mitigate the instability of the control system due to adaptive parameter drift, we extend the adaptive autopilot by including a static nonlinearity before the adaptive controller updated by RCAC. The contribution of this paper is thus the extension of the adaptive autopilot to mitigate parameter drift, the investigation of the effectiveness of three deadzone nonlinearities to reduce the sensitivity of the adaptive controller to sensor noise, and experimental demonstration of the improved robustness of the modified adaptive autopilot.

The paper is organized as follows. Section II briefly reviews the architecture of the adaptive autopilot used in this work. Section III introduces the three deadzone-augmented adaptive autopilots. Section IV describes the simulation and physical flight results to validate the proposed modifications of the adaptive autopilot. Finally, Section V concludes the paper with a discussion of the results.

II. ADAPTIVE AUTOPILOT

This section briefly reviews the adaptive autopilot used in this work. The adaptive autopilot is constructed by modifying the fixed-gain autopilot implemented in the PX4 flight stack [29]. The adaptive autopilot's architecture and notation is described in detail in [16], [17]. The augmentation used for parameter drift mitigation and its implementation is presented in Section III.

The autopilot consists of a mission planner and two nested loops, as shown in Figure 1. The *mission planner* generates the position, velocity, azimuth, and azimuth rate setpoints

from the user-defined waypoints. The outer loop consists of the *position controller* whose inputs are the position setpoint and velocity setpoints as well as the measured position and measured velocity of the multicopter. The output of the position controller is the thrust vector command. Note that the thrust vector output of the position controller is expressed in the Earth fixed coordinate system. The inner loop consists of the *attitude controller* whose inputs are the thrust vector setpoint, the azimuth setpoint, and azimuth rate setpoints, as well as the measured attitude and the angular velocity measured in the body-fixed frame. The output of the attitude controller is the moment command. The magnitude of the thrust vector and the moment vector uniquely determine the rotation rate of the four propellers.

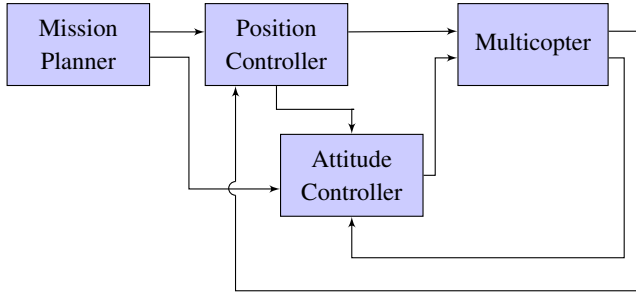


Fig. 1. Multicopter autopilot architecture.

The position controller consists of two cascaded linear controllers. The first controller G_r consists of three decoupled proportional controllers. The second controller G_v consists of three decoupled PID controllers. As shown in Figure 2, the two controllers are augmented with adaptive controllers based on the RCAC algorithm.

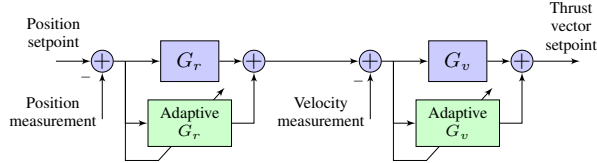


Fig. 2. The position controller in the adaptive autopilot.

Similarly, the attitude controller consists of two cascaded controllers. The first controller G_q is a nonlinear almost globally stabilizing controller [30], and the second controller G_ω consists of three decoupled PI controllers. As shown in Figure 3, the two controllers are augmented with adaptive controllers based on the RCAC algorithm.

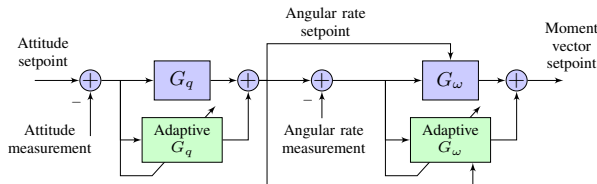


Fig. 3. The attitude controller in the adaptive autopilot.

Our prior works in [16], [17], [31] observed that some controller gains updated by RCAC in the adaptive autopilot drifted and eventually caused the adaptive autopilot to fail. Numerical investigations showed that the PI gains of the rate controller diverged even after an acceptable tracking

performance was achieved. Thus, we introduce a *deadzone* in the G_ω controller of the inner loop to reduce the sensitivity of the adaptive rate controller. The next section describes three nonlinear functions that are used to implement the deadzone in this work.

III. DEADZONE-AUGMENTED ADAPTIVE AUTOPILOT

This section describes the nonlinear functions used to implement a deadzone in the adaptive autopilot in order to improve noise robustness and prevent the onset of instabilities. To avoid undesirable updates due to noisy or small signals, the performance variable z used to define the retrospective cost minimized by the RCAC algorithm is modified, as shown below.

The performance variable z used to optimize the adaptive controller G_ω is replaced by $\mathcal{N}(z)$, where \mathcal{N} is an element-wise nonlinear function. The nonlinear function \mathcal{N} is chosen to implement a deadzone in the adaptive rate controller, thus suppressing the effect of small values of rate errors z_ω on the update of the rate controller gains θ_ω . Note that θ_ω has six components, that is, two PI gains for each direction. As a result, the inner loop controller in the adaptive autopilot is augmented with the deadzone, as shown in Figure 4. This work investigates the effectiveness of three choices of nonlinear functions to implement a deadzone, as described below.

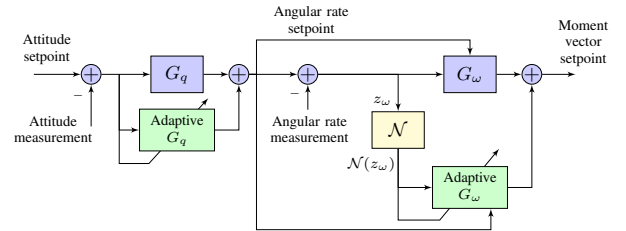


Fig. 4. The attitude controller augmented with the deadzone in the adaptive autopilot.

The first nonlinearity is given by

$$\mathcal{N}_1(x) \triangleq \begin{cases} x & x < -s, \\ 0 & x \in [-s, s], \\ x & x > s, \end{cases} \quad (1)$$

where $s > 0$. Note that \mathcal{N}_1 is a discontinuous nonlinear function whose output is zero if the input magnitude is less than s and is equal to the input if the input magnitude is greater than s . Figure 5(a) shows the output of \mathcal{N}_1 for several values of s .

The second nonlinearity is given by

$$\mathcal{N}_2(x) \triangleq \begin{cases} -\alpha(s_2 - s_1)^3 + 3\alpha(s_2 - s_1)^2(x + s_2) & x < s_2, \\ \alpha(x + s_1)^3 & x \in [-s_2, -s_1], \\ 0 & x \in [-s_1, s_1], \\ \alpha(x - s_1)^3 & x \in (s_1, s_2], \\ \alpha(s_2 - s_1)^3 + 3\alpha(s_2 - s_1)^2(x - s_2) & x > s_2, \end{cases} \quad (2)$$

where $s_1, \alpha > 0$, and $s_2 = s_1 + \sqrt{\frac{1}{3\alpha}}$. The parameter s_1 is the width of the deadzone and α affects the transition to the linear section. In particular, a large value of s_1 implies a wide deadzone, and a large value of α implies a quicker

transition to the linear section. Note that \mathcal{N}_2 introduces a bias to its output after the transition to the linear section. Figure 5(b) shows the output of \mathcal{N}_2 for $s_1 = 0.02$ and several values of α . Finally, note that \mathcal{N}_2 is a continuously differentiable nonlinear function.

The third nonlinearity is given by

$$\mathcal{N}_3(x) \triangleq \begin{cases} x & x < -s_2, \\ c_{1,3}x^3 + c_{1,2}x^2 + c_{1,1}x + c_{1,0} & x \in [-s_2, -s_1), \\ 0 & x \in [-s_1, s_1], \\ c_{u,3}x^3 + c_{u,2}x^2 + c_{u,1}x + c_{u,0} & x \in [s_1, s_2), \\ x & x > s_2, \end{cases} \quad (3)$$

where $s_2 > s_1 > 0$, and the parameters $c_{1,i}$ and $c_{u,i}$ are determined such that \mathcal{N}_3 is continuously differentiable. The parameter s_1 is the width of the deadzone and s_2 affects the transition to the linear section. Unlike \mathcal{N}_2 , the nonlinear function \mathcal{N}_3 converges to the asymptote $y = x$. Thus, the nonlinearity acts as an identity function for large input values while zeros out the small values of the input. Finally, note that $c_1 \triangleq [c_{1,0} \ c_{1,1} \ c_{1,2} \ c_{1,3}]^T = \mathcal{C}(-s_1, -s_2)$ is given by

$$\mathcal{C}(\xi_1, \xi_2) = \begin{bmatrix} 1 & \xi_1 & \xi_1^2 & \xi_1^3 \\ 0 & 1 & 2\xi_1 & 3\xi_1^2 \\ 1 & \xi_2 & \xi_2^2 & \xi_2^3 \\ 0 & 1 & 2\xi_2 & 3\xi_2^2 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ \xi_2 \\ 1 \end{bmatrix}. \quad (4)$$

The parameter $c_u = \mathcal{C}(s_1, s_2)$ is computed similarly. Note that c_1 and c_u are computed apriori. Figure 5(c) shows the output of \mathcal{N}_3 for $s_1 = 0.02$ and several values of s_2 .

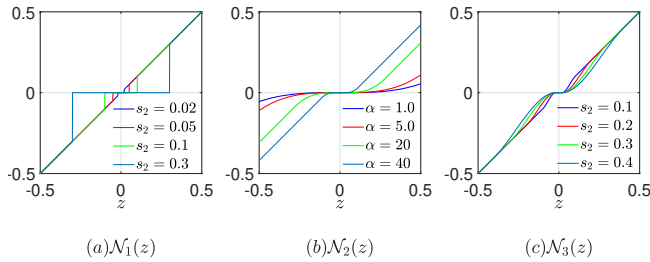


Fig. 5. Nonlinearities used to implement deadzone.

IV. FLIGHT TESTS

This section describes the numerical simulation and flight test results investigating the effectiveness of the three deadzone nonlinearities described in the previous section on mitigating the parameter drift and adaptive autopilot failure. For these tests, a quadcopter platform is considered.

A. Numerical Simulation

We first investigate the effect of the nonlinearities to mitigate the parameter drift in the adaptive autopilot with the jMAVSIM quadcopter simulation implemented in the PX4 flight stack. The three nonlinearities described in the previous section are implemented in the `mc_rate_control`¹ module in PX4.

To investigate and quantify the potential improvements in the adaptive autopilot's performance, we command the

¹https://github.com/JAParedes/PX4-Autopilot/tree/RCAC_MC_FW_dev_mavlink/src/modules/mc_rate_control

quadcopter to follow a trajectory generated using a second-order Hilbert curve. The quadcopter is commanded to fly the mission with a fixed-gain autopilot and four RCAC-based adaptive autopilots. The first adaptive autopilot does not use deadzone in the pitch rate controller. In contrast, the second, third, and fourth adaptive autopilots use \mathcal{N}_1 , \mathcal{N}_2 , and \mathcal{N}_3 nonlinear functions to implement the deadzone in the pitch rate controller, respectively.

Figure 6 shows the trajectory-following response of the quadcopter with the fixed-gain autopilot and the four adaptive autopilots. Note that all four adaptive autopilots have similar tracking performance. Note that the trajectory-following response of all four adaptive autopilots is better than the fixed-gain autopilot. However, with the first adaptive autopilot, that is, the adaptive autopilot without the deadzone, high-frequency oscillations are observed in the pitch and pitch rate response. Figures 7 and 8 show the pitch-rate error response and the pitching moment in the adaptive autopilot. Note the high-frequency oscillations between 20 and 50 seconds in this case. In contrast, the three adaptive autopilots with deadzones are able to suppress these oscillations as shown in Figures 7 and 8.

Figure 9 shows the frequency content of the pitching moment applied to the quadcopter. Note the large magnitude of the frequency content at higher frequencies generated by the first adaptive autopilot, whereas the high-frequency content is suppressed with the deadzone nonlinearities in the adaptive autopilot. Finally, Figure 10 shows the controller gains updated by the RCAC algorithm in the four adaptive autopilots. Note that without the deadzone nonlinearity, the controller gains drift as shown in the first Figure 10(a). With the deadzone nonlinearity in the adaptive autopilot, the controller gain drift is mitigated.

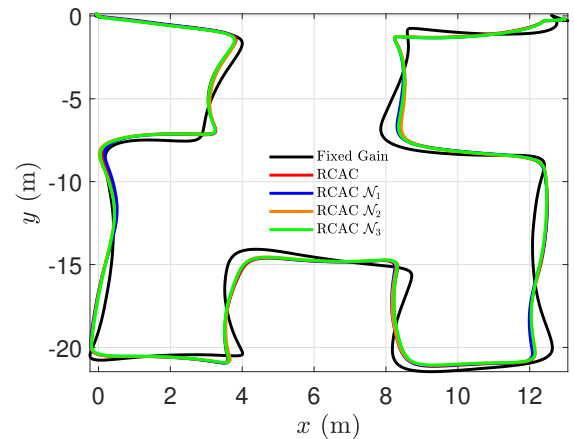


Fig. 6. **Simulation results.** Trajectory-following response of the quadcopter with the fixed-gain autopilot and the four adaptive autopilots.

B. Physical Flight Tests

Next, we repeat the experiment described in the previous subsection in a physical flight environment using a Holybro X500 quadcopter frame with a Pixhawk 6C flight computer running the PX4 flight software. The flight tests are conducted in the M-air facility at the University of Michigan, Ann Arbor.

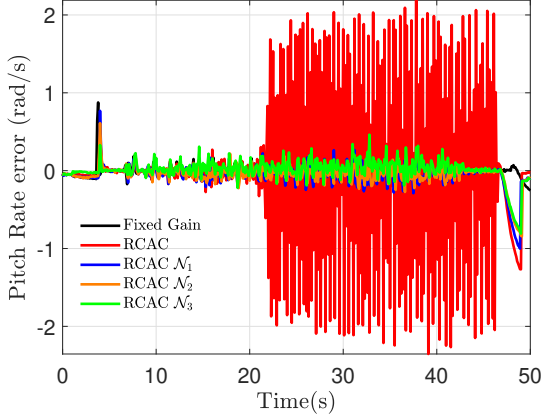


Fig. 7. **Simulation results.** Pitch-rate error response of the quadcopter with the fixed-gain autopilot and the four adaptive autopilots.

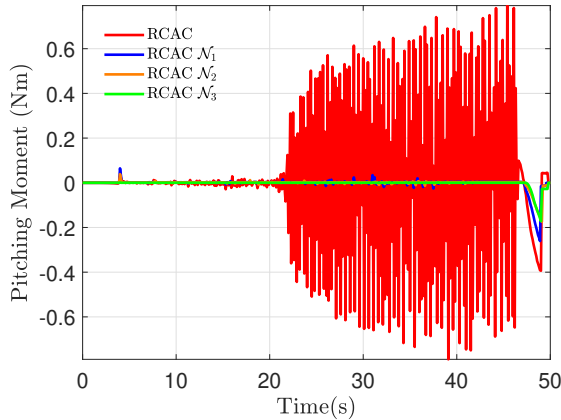


Fig. 8. **Simulation results.** The pitching moment applied to the quadcopter with the four adaptive autopilots.

Figure 11 shows the trajectory-following response of the quadcopter with the fixed-gain autopilot and adaptive autopilot. Like the simulation experiments, the trajectory-following response is similar for all adaptive autopilots. Moreover, the trajectory-following response of all four adaptive autopilots is better than the fixed-gain autopilot. Furthermore, similar to the simulation experiments, with the adaptive autopilot without the deadzone, high-frequency oscillations are observed in the pitch and pitch rate response. Figures 12 and 13 show the pitch-rate error response and the pitching moment in the adaptive autopilot. Note the high-frequency oscillations between 20 and 50 seconds in this case. In contrast, the three adaptive autopilots with deadzones are able to suppress these oscillations as shown in Figures 12 and 13.

Figure 14 shows the frequency content of the pitching moment applied to the quadcopter. Note the large magnitude of the frequency content at higher frequencies generated by the first adaptive autopilot, whereas the high-frequency content is suppressed with the deadzone nonlinearities in the adaptive autopilot. Finally, Figure 15 shows the controller gains updated by the RCAC algorithm in the four adaptive autopilots. Note that without the deadzone nonlinearity, the controller gains drift as shown in the first Figure 15a). With the deadzone nonlinearity in the adaptive autopilot, the

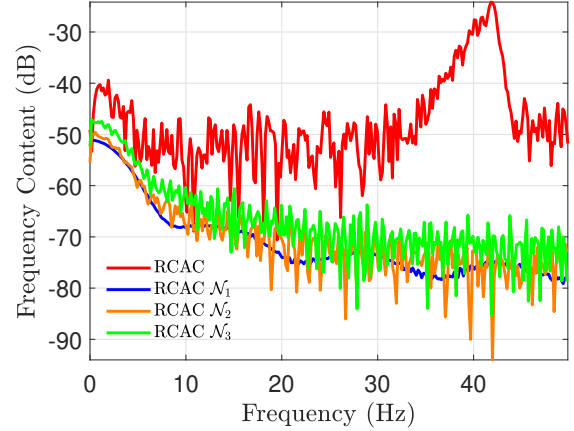


Fig. 9. **Simulation results.** The frequency content of the pitching moment applied to the quadcopter with the four adaptive autopilots.

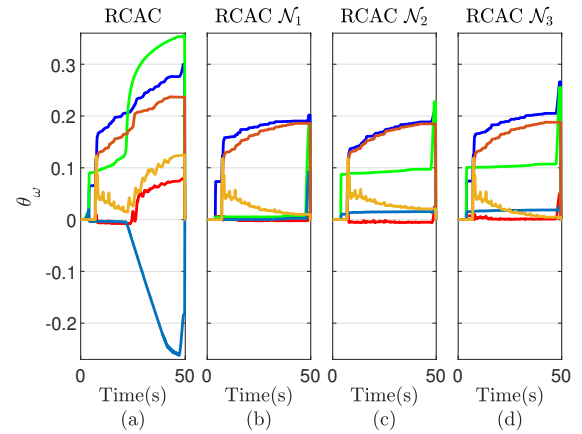


Fig. 10. **Simulation results.** The pitch rate controller gains updated by the RCAC algorithms in the four adaptive autopilots.

controller gain drift is mitigated.

To quantify the improvements with the deadzone-augmented adaptive autopilots, we compute the cost metrics J_r defined as the root mean square (RMS) value of the position error and J_ω defined as the RMS value of the pitch rate error in the mission. Note that the metrics J_r and J_ω are computed offline after completing the mission. Figure 16 shows the cost metrics comparing the performance of the four adaptive autopilots in both simulation and flight tests. Note that the position-tracking performance improves with all four adaptive autopilots in both simulation and flight tests. However, without the deadzone nonlinearity, the adaptive autopilot suffers from large pitch rate errors. The deadzone nonlinearities in the adaptive autopilot mitigate the parameter drift and thus improve the pitch rate error response, as shown by the sharp drop in J_ω .

V. CONCLUSIONS

This paper presented three deadzone-augmented adaptive autopilots to increase robustness to adaptive parameter drift caused by noise and small amplitude signals. The deadzones are implemented by three static nonlinear functions. The modified adaptive autopilots were implemented in the

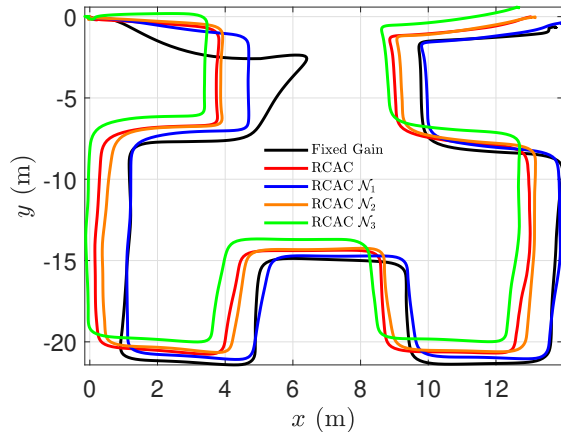


Fig. 11. **Physical flight results.** Trajectory-following response of the quadcopter with the fixed-gain autopilot and the four adaptive autopilots.

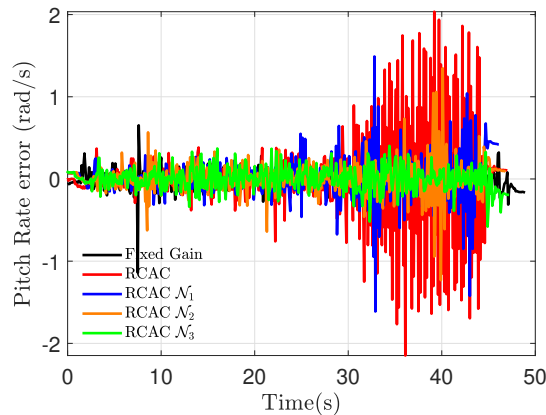


Fig. 12. **Physical flight results.** Pitch-rate error response of the quadcopter with the fixed-gain autopilot and the four adaptive autopilots.

PX4 flight stack and their performance was compared in numerical simulation as well as physical flight tests. It was shown that the deadzone nonlinearities can mitigate the high-frequency oscillations that appear in the angular rate response due to adaptive parameter drift without significantly affecting the trajectory-following performance of the adaptive controller. Furthermore, it was shown that, out of the three considered nonlinearities, augmenting RCAC with the simplest discontinuous static function, resulted in the least amount of parameter drift and the most stable flight.

REFERENCES

- [1] A. Mukherjee, S. Misra, and N. S. Raghuvanshi, "A survey of unmanned aerial sensing solutions in precision agriculture," *J. Netw. Comput. Appl.*, vol. 148, p. 102461, 2019.
- [2] A. Lucieer, S. M. d. Jong, and D. Turner, "Mapping landslide displacements using Structure from Motion (SfM) and image correlation of multi-temporal UAV photography," *Prog. Phys. Geogr.*, vol. 38, no. 1, pp. 97–116, 2014.
- [3] V. V. Klemas, "Coastal and environmental remote sensing from unmanned aerial vehicles: An overview," *J. Coast. Res.*, vol. 31, no. 5, pp. 1260–1267, 2015.
- [4] Y. Li and C. Liu, "Applications of multirotor drone technologies in construction management," *Int. J. Constr. Manag.*, vol. 19, no. 5, pp. 401–412, 2019.

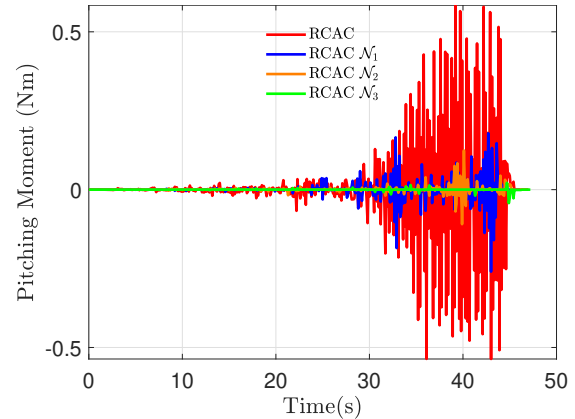


Fig. 13. **Physical flight results.** The pitching moment applied to the quadcopter with the four adaptive autopilots.

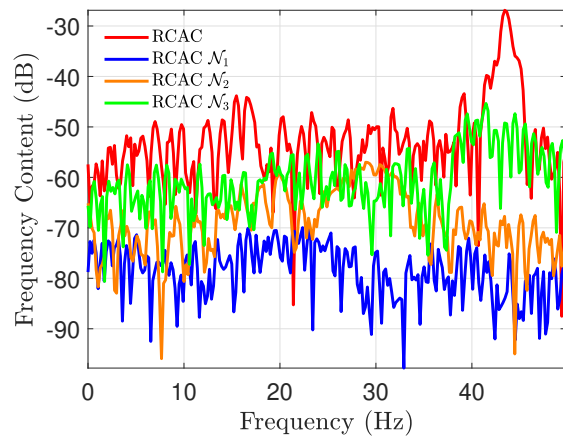


Fig. 14. **Physical flight results.** The frequency content of the pitching moment applied to the quadcopter with the four adaptive autopilots.

- [5] D. K. Villa, A. S. Brandao, and M. Sarcinelli-Filho, "A survey on load transportation using multirotor UAVs," *J. Intell. Robot. Syst.*, vol. 98, pp. 267–296, 2020.
- [6] T. P. Nascimento and M. Saska, "Position and attitude control of multi-rotor aerial vehicles: A survey," *Annu. Rev. Contr.*, vol. 48, pp. 129–146, 2019.
- [7] J. A. Marshall, W. Sun, and A. L'Affitto, "A survey of guidance, navigation, and control systems for autonomous multi-rotor small unmanned aerial systems," *Annu. Rev. Contr.*, vol. 52, pp. 390–427, 2021.
- [8] P. Castillo, A. Dzul, and R. Lozano, "Real-time stabilization and tracking of a four-rotor mini rotorcraft," *IEEE Trans. Contr. Sys. Tech.*, vol. 12, no. 4, pp. 510–516, 2004.
- [9] J. M. Portella Delgado and A. Goel, "Circumventing unstable zero dynamics in input-output linearization of longitudinal flight dynamics," in *AIAA SCITECH 2024 Forum*, 2024, p. 0321.
- [10] B. J. Emran and H. Najjaran, "A review of quadrotor: An underactuated mechanical system," *Annu. Rev. Contr.*, vol. 46, pp. 165–180, 2018.
- [11] R. Amin, L. Aijun, and S. Shamshirband, "A review of quadrotor UAV: Control methodologies and performance evaluation," *Int. J. Autom. Contr.*, vol. 10, no. 2, pp. 87–103, 2016.
- [12] B. Whitehead and S. Bieniawski, "Model reference adaptive control of a quadrotor UAV," in *AIAA Guid. Nav. Contr. Conf. Ex.*, 2010, p. 8148.

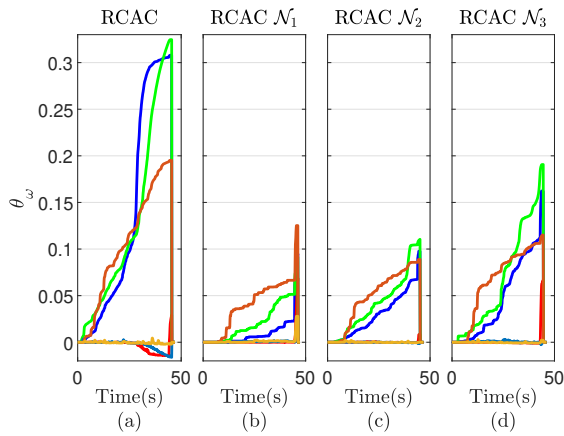
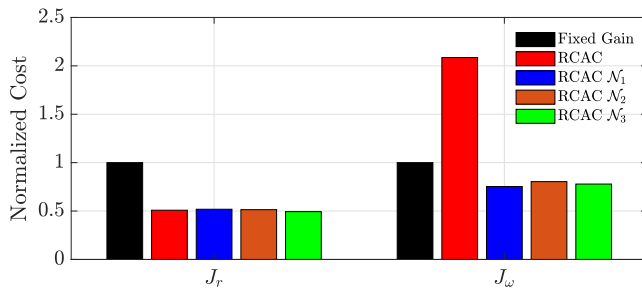
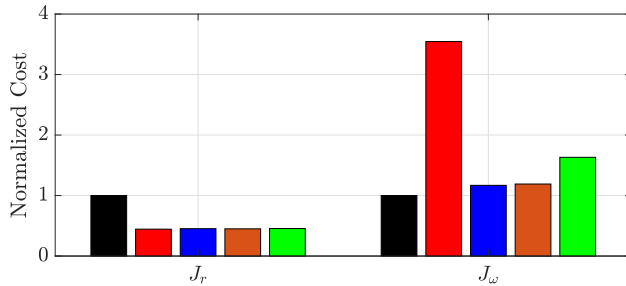


Fig. 15. **Physical flight results.** The pitch rate controller gains updated by the RCAC algorithms in the four adaptive algorithms.



(a) Simulation results



(b) Flight results

Fig. 16. Cost metrics J_r and J_ω comparing the performance of the four adaptive autopilots in (a) simulation tests and (b) flight tests.

[13] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor UAVs: A design trade study with flight evaluations," *IEEE Trans. Contr. Sys. Tech.*, vol. 21, no. 4, pp. 1400–1406, 2012.

[14] Z. Zuo and P. Ru, "Augmented L1 adaptive tracking control of quad-rotor unmanned aircrafts," *IEEE. Trans. Aerosp. Elec. Sys.*, vol. 50, no. 4, pp. 3090–3101, 2014.

[15] O. Mofid and S. Mobayen, "Adaptive sliding mode control for finite-time stability of quad-rotor UAVs with parametric uncertainties," *ISA trans.*, vol. 72, pp. 1–14, 2018.

[16] A. Goel, J. A. Paredes, H. Dadhaniya, S. A. Ul Islam, A. M. Salim, S. Ravela, and D. Bernstein, "Experimental implementation of an adaptive digital autopilot," in *Proc. Amer. Contr. Conf.*, 2021, pp. 3737–3742.

[17] J. Spencer, J. Lee, J. A. Paredes, A. Goel, and D. Bernstein, "An adaptive PID autotuner for multicopters with experimental results," in *Proc. Int. Conf. Rob. Autom.*, IEEE, 2022, pp. 7846–7853.

[18] Y. Rahman, A. Xie, and D. S. Bernstein, "Retrospective Cost Adaptive Control: Pole Placement, Frequency Response,

and Connections with LQG Control," *IEEE Control System Magazine*, vol. 37, pp. 28–69, Oct. 2017.

[19] M. Kamalidar, S. A. U. Islam, S. Sanjeevini, A. Goel, J. B. Hoagg, and D. S. Bernstein, "Adaptive digital PID control of first-order-lag-plus-dead-time dynamics with sensor, actuator, and feedback nonlinearities," *Advanced Control for Applications*, vol. 1, no. 1, e20, 2019.

[20] B. Egardt, "Stability of discrete time controllers," in *Stability of adaptive controllers*, Springer, 1979, pp. 43–86.

[21] P. A. Ioannou and J. Sun, "Robust adaptive laws," in *Robust adaptive control*, Prentice-Hall, 1996, pp. 544–554.

[22] R. Ortega and Y. Tang, "Robustness of adaptive controllers—a survey," *Automatica*, vol. 25, no. 5, pp. 651–677, 1989.

[23] B. E. Ydstie, "Transient performance and robustness of direct adaptive control," *IEEE Trans. Autom. Contr.*, vol. 37, no. 8, pp. 1091–1105, 1992.

[24] E. Lavretsky and K. A. Wise, "Robust adaptive control," in *Robust and adaptive control: With aerospace applications*, Springer, 2012, pp. 317–353.

[25] E. Kharisov, N. Hovakimyan, and K. Åström, "Comparison of several adaptive controllers according to their robustness metrics," in *AIAA Guid. Nav. Contr. Conf. Ex.*, 2010, p. 8047.

[26] N. Hovakimyan and C. Cao, *L1 adaptive control theory: Guaranteed robustness with fast adaptation*. SIAM, 2010.

[27] C. Nicol, C. Macnab, and A. Ramirez-Serrano, "Robust adaptive control of a quadrotor helicopter," *Mechatronics*, vol. 21, no. 6, pp. 927–938, 2011.

[28] C. Macnab, "Using RBFs in a CMAC to prevent parameter drift in adaptive control," *Neurocomputing*, vol. 205, pp. 45–52, 2016.

[29] PX4 User Guide, *Multicopter Control Architecture*, [Online; accessed 26 February 2023].

[30] N. A. Chaturvedi, A. K. Sanyal, and N. H. McClamroch, "Rigid-Body Attitude Control," *IEEE Control Systems Magazine*, vol. 31, no. 3, pp. 30–51, 2011.

[31] A. Goel, A. M. Salim, A. Ansari, S. Ravela, and D. Bernstein, "Adaptive digital PID control of a quadcopter with unknown dynamics," *arXiv preprint arXiv:2006.00416*, 2020.