

Experimental Flight Testing of a Fault-Tolerant Adaptive Autopilot for Fixed-Wing Aircraft

Joonghyun Lee, John Spencer, Siyuan Shao, Juan Augusto Paredes, Dennis S. Bernstein, Ankit Goel

Abstract—This paper presents an adaptive autopilot for fixed-wing aircraft and compares its performance with a fixed-gain autopilot. The adaptive autopilot is constructed by augmenting the autopilot architecture with adaptive control laws that are updated using retrospective cost adaptive control. In order to investigate the performance of the adaptive autopilot, the default gains of the fixed-gain autopilot are scaled to degrade its performance. This scenario provides a venue for determining the ability of the adaptive autopilot to compensate for the degraded fixed-gain autopilot. Next, the performance of the adaptive autopilot is examined under failure conditions by simulating a scenario where one of the control surfaces is assumed to be stuck at an unknown angle. The adaptive autopilot is also tested in physical flight experiments under degraded-nominal conditions, and the resulting performance improvement is examined.

I. INTRODUCTION

Autonomous flight control of an aircraft under rapidly changing conditions requires an autopilot that can control the aircraft in uncertain environments and without detailed models. An autopilot for a fixed-wing aircraft typically consists of a set of trim commands along with low-level controllers to follow intermediate commands. The trim conditions for an aircraft can be computed by solving nonlinear algebraic equations for trim equilibria [1], but a detailed model of the aircraft aerodynamics is required. Moreover, for low-cost aircraft that are usually repaired or modified onsite, the true aerodynamic properties may be different from nominal aerodynamics. Consequently, a fixed-gain autopilot may not be able to maintain performance in a rapidly changing environment or under failure conditions such as damaged wings or faulty actuators. In this scenario, an adaptive autopilot may be able to compensate for the lost performance by updating the autopilot gains accordingly. With these motivations in mind, this paper explores the use of an in situ learning technique to modify the autopilot during the flight.

Various adaptive control techniques have been investigated for fixed-wing aircraft control [2]. A sliding mode fault-tolerant tracking control scheme was used for control of a fixed-wing UAV under actuator saturation and state constraints in [3], [4]. A backstepping algorithm was used in [5] to design a nonlinear flight controller for a fixed-wing UAV with thrust vectoring. An MRAC-based technique was

used to augment the control system to improve the dynamic performance of a fixed-wing aircraft in [6]. However, these techniques rely on the availability of a sufficiently detailed model for the control system synthesis.

In contrast, the present paper uses the retrospective cost adaptive control (RCAC) algorithm to learn the autopilot gains from the measured data in situ. RCAC is a digital adaptive control technique that is applicable to stabilization, command following, and disturbance rejection. Instead of relying on a model of the system, RCAC uses the past measured data and past applied input to recursively optimize the controller gains. RCAC is described in [7], and its extension to digital PID control is given in [8]. The application of RCAC for a multicopter autopilot are described in [9], [10].

The contribution of this paper is the development of an adaptive autopilot for fixed-wing aircraft, and a comparison of its performance with a well-tuned fixed-gain autopilot under nominal conditions, performance recovery of a degraded-nominal autopilot, and performance improvement under actuator failure. In particular, this paper presents the potential advantages of an adaptive autopilot by investigating two scenarios. In the first scenario, a well-tuned fixed-gain controller is degraded by scaling all of the gains by a small factor, and it is shown that the adaptive autopilot is able to compensate for the degraded gains by learning the necessary gains. This scenario is investigated both in simulation and in physical flight experiments. In the second scenario, the aircraft is simulated with a faulty aileron, thus emulating an actuator failure condition, and it is shown, in simulation experiments, that the adaptive autopilot improves the trajectory-tracking performance.

The paper is organized as follows: Section II defines the notation used in this paper, Section III reviews the autopilot architecture implemented in the PX4 flight stack, Section IV presents the adaptive augmentation of autopilot, Section V presents the simulation flight tests, and Section VI presents the outdoor flight tests. Finally, Section VII concludes the paper with a summary and future research directions.

II. NOTATION

Let F_E denote an Earth-fixed frame such that \hat{k}_E is aligned with the acceleration due to gravity \vec{g} . Let F_{AC} denote an aircraft-fixed frame such that \hat{i}_{AC} is aligned with the fuselage, \hat{j}_{AC} is along the wing, and \hat{k}_{AC} is chosen to complete the right-handed frame. Note that \hat{k}_{AC} points vertically down. Next, let c denote the center of mass of the aircraft, and let w be a point fixed on Earth. The coordinates of the aircraft relative to w in the Earth frame are

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

Joonghyun Lee, John Spencer, Siyuan Shao, Juan Augusto Paredes, and Dennis S. Bernstein are with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109. joonghle, spjohn, shaosy, jparedes, dsbaero@umich.edu

Ankit Goel is with the Department of Mechanical Engineering, University of Maryland, Baltimore County, MD 21250. ankgoel@umbc.edu

denoted by $r \triangleq \vec{r}_{c/w}|_E \in \mathbb{R}^3$. The velocity of the aircraft relative to w in the Earth frame is $v \triangleq \vec{v}_{c/w}|_E \in \mathbb{R}^3$. Let Ψ , Θ , and Φ denote the 3-2-1 azimuthal, elevation, and bank Euler angles of the aircraft. The angular velocity of F_{AC} relative to F_E in the aircraft-fixed frame is given by $\omega \triangleq \vec{\omega}_{AC/E}|_{AC} \in \mathbb{R}^3$. The angular acceleration of F_{AC} relative to F_E in the aircraft-fixed frame is given by $\alpha \triangleq \vec{\alpha}_{AC/E}|_{AC} \in \mathbb{R}^3$. The measurement of the variable x is denoted by x_m , and the setpoint for the variable x is denoted by x_s . Finally, let $e_3 \triangleq [0 \ 0 \ 1]^T$.

The angles Ψ , Θ , and Φ comprise a 3-2-1 sequence of Euler angles that parameterize the orientation of F_{AC} relative to F_E . The components of ω are the yaw rate, pitch rate, and roll rate, which are different from the azimuth rate, elevation rate, and bank rate. Hence, integrating the components of ω does not yield the azimuthal, elevation, and bank Euler angles. In fact, the relation between the Euler-angle rates and the components of ω is given by (4) in the following section.

III. FLIGHT CONTROL ARCHITECTURE

In this work, we consider the flight control architecture implemented in the PX4 flight stack. The control system consists of a mission planner and two cascaded controllers in nested loops as shown in Figure 1. The mission planner generates position setpoints based on user-defined waypoints.

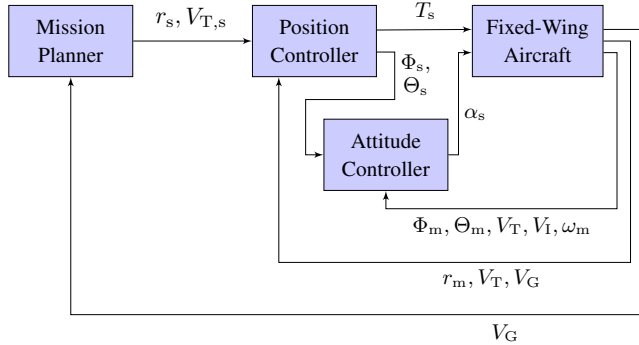


Fig. 1. Autopilot architecture.

The outer loop, also called the position controller, consists of two decoupled controllers for the longitudinal and lateral motion of the aircraft, as shown in Figure 2. The longitudinal controller is based on the total energy control system (TECS) described in [11]–[14], and the lateral controller is based on the guidance law described in [15]. The inputs to the position controller are the true airspeed setpoint $V_{T,s}$, the position setpoint r_s , the true airspeed V_T , the position measurement r_m , and the ground velocity V_G . The TECS input includes the altitude setpoint $h_s \triangleq e_3^T r_s$ and the altitude measurement $h_m \triangleq e_3^T r_m$. The longitudinal controller generates the thrust and the elevation setpoint, and the lateral controller generates the bank setpoint. The output of the position controller is thus the thrust setpoint T_s and the attitude setpoint Θ_s, Φ_s .

The inner loop, also called the attitude controller, consists of two cascaded controllers, as shown in Figure 3. The

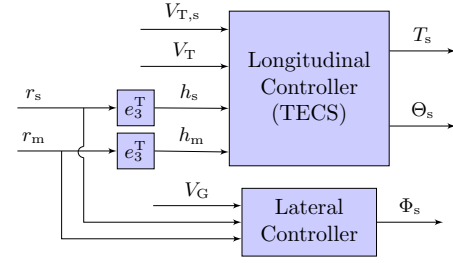


Fig. 2. Position controller architecture.

first controller uses the elevation and bank errors and a proportional control law to generate the elevation-rate and bank-rate setpoints. In particular, the elevation-rate setpoint $\dot{\Theta}_s$ and the bank-rate setpoint $\dot{\Phi}_s$ are given by

$$\dot{\Theta}_s = k_\theta(\Theta_s - \Theta_m), \quad (1)$$

$$\dot{\Phi}_s = k_\phi(\Phi_s - \Phi_m), \quad (2)$$

where k_θ, k_ϕ are the scalar gains. The azimuthal-rate is algebraically given by

$$\dot{\Psi}_s = \frac{g \tan \Phi_s \cos \Theta_s}{V_T} \quad (3)$$

to ensure coordinated turn. Finally, the body-fixed angular-velocity setpoint ω_s is given by

$$\omega_s = S(\Theta_m, \Phi_m) \begin{bmatrix} \dot{\Phi}_s \\ \dot{\Theta}_s \\ \dot{\Psi}_s \end{bmatrix}, \quad (4)$$

where

$$S(\Theta_m, \Phi_m) \triangleq \begin{bmatrix} 1 & 0 & \sin \Theta_m \\ 0 & \cos \Phi_m & \sin \Phi_m \cos \Theta_m \\ 0 & -\sin \Phi_m & \cos \Phi_m \cos \Theta_m \end{bmatrix}. \quad (5)$$

Next, a feedforward and a PI control law generates the angular-acceleration setpoint α_s . In particular, α_s is given by

$$\alpha_s = \frac{V_{T,0}}{V_T} G_{\omega,ff} \omega_s + \left(\frac{V_{I,0}}{V_I} \right)^2 G_{\omega,PI}(\mathbf{q}) (\omega_s - \omega_m), \quad (6)$$

where $G_{\omega,ff} = k_{\omega,ff}$ is a proportional control law, $G_{\omega,PI}(\mathbf{q}) = k_{\omega,P} + \frac{k_{\omega,I}}{\mathbf{q}-1}$ is a PI control law, V_I is the indicated airspeed, and $V_{T,0}$ and $V_{I,0}$ are the true airspeed and the indicated airspeed at trim conditions respectively, which are aircraft parameters. Note that \mathbf{q} is the forward-shift operator, $k_{\omega,ff}$, $k_{\omega,P}$, and $k_{\omega,I}$ are 3×3 diagonal matrices, and are thus parameterized by 9 scalar gains. Finally, using the angular-acceleration setpoint, the actuator deflections are computed using control allocation methods.

The fixed-wing autopilot thus consists of 11 gains. In practice, these 11 gains are tuned manually, which requires considerable expertise. We assume that the default gains implemented in PX4 are well tuned, and thus we refer to the autopilot with the default PX4 gains as the *nominal autopilot*.

To investigate potential improvements and demonstrate the ability of the adaptive autopilot to recover performance, the gains in the nominal autopilot are multiplied by a scalar α_d

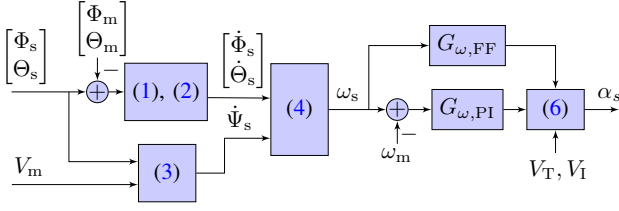


Fig. 3. Attitude controller architecture.

in order to degrade its performance. A fixed-gain autopilot with the degradation factor $\alpha_d \neq 1$ is referred to as the *degraded-nominal autopilot*. Note that $\alpha_d \neq 1$ is equivalent to the case of a poor choice of controller gains in the fixed-gain autopilot.

IV. ADAPTIVE AUTOPILOT

This section describes the adaptive autopilot, which is constructed by augmenting the nominal autopilot. The nominal autopilot is the autopilot described in Section III with fixed gains. In the adaptive autopilot, the fixed-gain control laws of the nominal autopilot are augmented with adaptive control laws, whose coefficients are updated by the retrospective cost adaptive control (RCAC) algorithm described in [7], [8]. RCAC is used to augment the fixed-gain controllers of a multicopter autopilot in [9], [10]. The output of a modified controller in the adaptive autopilot is thus given by the sum of the fixed-gain and the adaptive control law, as shown in Figure 4.

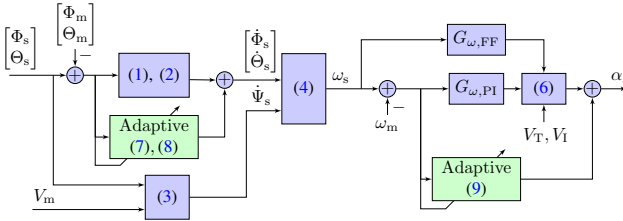


Fig. 4. Adaptive augmentation in the attitude controller.

The bank and elevation rate setpoints $\dot{\Phi}_s, \dot{\Theta}_s$ in the adaptive autopilot are given by

$$\dot{\Theta}_s = k_\theta(\Theta_s - \Theta_m) + u_\Theta, \quad (7)$$

$$\dot{\Phi}_s = k_\phi(\Phi_s - \Phi_m) + u_\Phi, \quad (8)$$

where the scalar adaptive control signals u_Θ and u_Φ are computed by RCAC. Similarly, the angular acceleration setpoint in the adaptive autopilot is given by

$$\alpha_s = \frac{V_{T,0}}{V_T} G_{\omega,ff} \omega_s + \left(\frac{V_{I,0}}{V_I} \right)^2 G_{\omega,PI}(\mathbf{q}) (\omega_s - \omega_m) + u_{\omega,PI}, \quad (9)$$

where $u_{\omega,PI}$ is computed by RCAC. Note that $u_{\omega,PI} \in \mathbb{R}^3$, and each component of $u_{\omega,PI}$ is updated by RCAC, where the error variable is the corresponding error term.

V. SIMULATION RESULTS

In this section, we investigate the performance of the adaptive autopilot and compare it to the performance of the nominal autopilot, implemented in PX4, in the Gazebo simulation environment. The aircraft dynamics simulated in Gazebo are based on the standard catapult-launched plane model¹ and are integrated in the PX4 version V1.13.0dev². We also consider the case of a faulty actuator. To simulate a fault scenario, we assume that one of the ailerons is frozen at an unknown angle.

Numerical simulations show that the aircraft performance is robust to TECS and the lateral controller gains. Therefore, in this work, we focus on augmenting only the attitude controller with the adaptive control law, and thus α_d degrades only the attitude controller in the nominal autopilot. The hyperparameters P_0 , R_u , and σ used in RCAC are shown in Table I. Furthermore, we set $R_z = 1$ in all adaptive controllers and all tests. Note that once the RCAC hyperparameters are tuned, they are fixed and thus they are not changed as α_d is varied across the simulation tests.

TABLE I

RCAC HYPERPARAMETERS FOR ALL SIMULATIONS.

Controller	P_0	R_u	σ
(7), θ_Θ	1	0.001	-0.1
(8), θ_Φ	1	0.001	-0.1

The mission waypoints are shown in Figure 5. The aircraft is assumed to be launched by a catapult from the launch point, and is commanded to fly toward the point T while climbing to an altitude of 20 m. The aircraft is then commanded to fly around point 2 in a steady-state circular flight with a radius of 30 m for one minute. Finally, the aircraft is commanded to land along the green strip.



Fig. 5. Waypoints used to construct the flight trajectory in simulation experiments.

In order to quantify and compare the performance of the autopilot, bank, elevation, and trajectory-tracking error metrics are defined as

$$J_\Phi \triangleq \sqrt{\frac{1}{N} \sum_{i=1}^N (\Phi_{s,i} - \Phi_{m,i})^2}, \quad (10)$$

¹<https://docs.px4.io/main/en/simulation/gazebo-vehicles.html>

²https://github.com/JAParedes/PX4-Autopilot/tree/RCAC-FW_UM

$$J_{\Theta} \triangleq \sqrt{\frac{1}{N} \sum_{i=1}^N (\Theta_{s,i} - \Theta_{m,i})^2}, \quad (11)$$

$$J_{\text{traj}} \triangleq \sqrt{\frac{1}{N} \sum_{i=1}^N e_{x-\text{track},i}^2}, \quad (12)$$

where N is the number of measurements during the flight, $e_{x-\text{track}}$ is the cross-track error, which is defined as the minimum distance between the current position and desired trajectory. These error metrics are computed offline.

Figures 6, 7 and 8 show the simulation results. As shown in Figure 8, the adaptive autopilot improves the performance over the nominal performance. For $\alpha_d = 0.5$, the trajectory following response degrades substantially with the degraded-nominal autopilot. In this case, the adaptive autopilot recovers the baseline performance. In fact, as the nominal controller is degraded, RCAC compensates by providing larger values of the corresponding gains. Finally, the adaptive autopilot is also able to learn the gains from a *cold start*, that is, the case where $\alpha_d = 0$, which switches off the nominal autopilot completely.

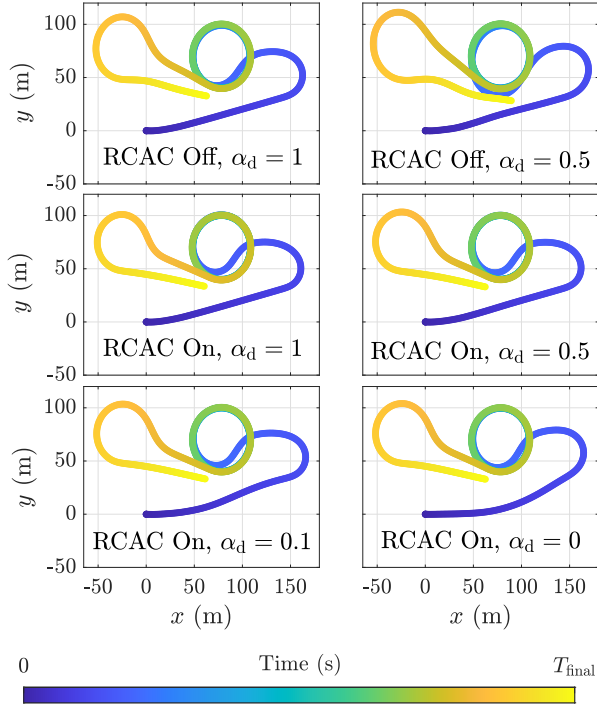


Fig. 6. **Flight simulation.** Ground trace of the aircraft with the nominal, degraded-nominal, and the adaptive autopilot for several values of the degradation factor α_d .

Next, we investigate the performance of the adaptive autopilot in the case of faulty actuators. In particular, we consider the case where an aileron is stuck at an unknown angle as shown in Figure 9. With the aileron stuck at an unknown position, we command the aircraft to follow the mission waypoints shown in Figure 5. This test is performed with both the nominal and adaptive controller. In both cases, note that $\alpha_d = 1$. Figures 10, 11 and 12 show

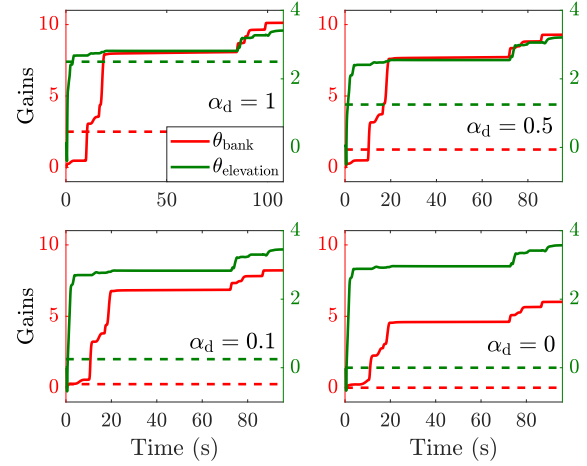


Fig. 7. **Flight simulation.** Adaptive bank and elevation controller gains optimized by RCAC in the adaptive autopilot for several values of the degradation factor α_d . In each case, the fixed gains are degraded by α_d and RCAC updates the adaptive control laws. The fixed gains are shown in dashes, and the adaptive gains are shown in solid for both the bank and elevator controller.

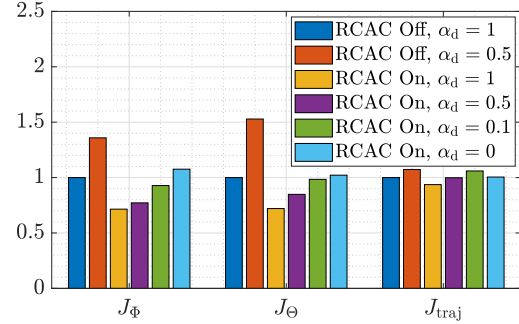


Fig. 8. **Flight simulation.** Bank, elevation, and trajectory-tracking error metrics obtained with the nominal, degraded-nominal, and adaptive autopilots for several values of α_d . Note that all metrics are normalized by the corresponding error metric obtained with the nominal fixed-gain autopilot.

the simulation results. As shown in Figure 12, the adaptive autopilot improves the trajectory-tracking error in the case of the faulty actuator and recovers the benchmark performance.

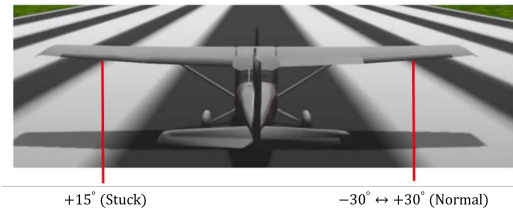


Fig. 9. **Faulty actuator.** The left aileron is stuck at an unknown angle.

VI. FLIGHT TEST RESULTS

This section presents the experimental flight results obtained with the adaptive autopilot. In this work, the flight tests are conducted with a Volantex Ranger 1600 fixed-wing aircraft, shown in Figure 13, at the Scio Flyers RC model aircraft club located at (42.298N, 83.843W). To demonstrate the performance improvements due to the adaptive autopilot, the performance of the nominal autopilot is degraded by scaling its fixed gains by the degradation factor α_d . In this work, we focus only on the attitude controller, thus the gains

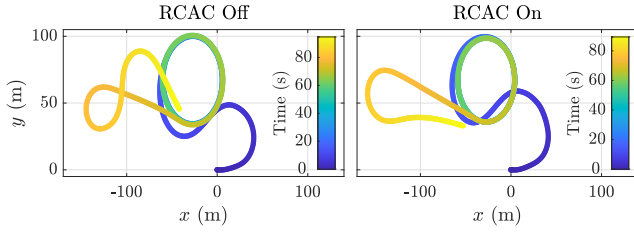


Fig. 10. **Faulty actuator.** Ground trace of the aircraft with a faulty actuator. The plots on the left and right are obtained with the nominal and adaptive autopilot, respectively. Note that, in both autopilots, $\alpha_d = 1$.

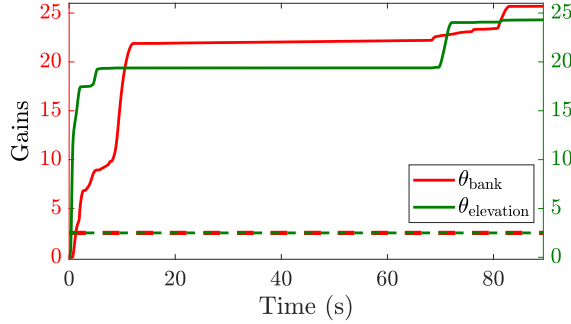


Fig. 11. **Faulty actuator.** Adaptive bank and elevation controller gains optimized by RCAC in the adaptive autopilot for the faulty actuator case. The fixed gains are shown in dashes, and the adaptive gains are shown in solid for both the bank and elevator controller.

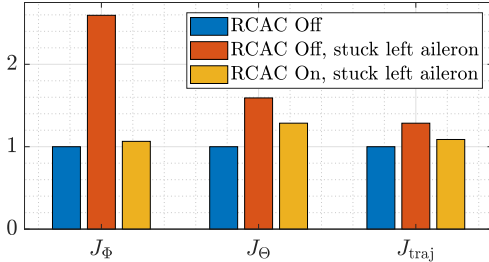


Fig. 12. **Faulty actuator.** Bank, elevation, and trajectory-tracking error metrics in the case of a faulty actuator with the nominal and adaptive autopilots. Note that the error metrics are normalized by the corresponding error metric obtained with the nominal autopilot and healthy actuator.

of only the attitude controller in the nominal autopilot are scaled. The hyperparameters P_0 , R_u , and σ used in RCAC are shown in Table II. Furthermore, we set $R_z = 1$ in all adaptive controllers and all tests. Note that once the RCAC hyperparameters are tuned, they are not changed as α_d is varied across the flight tests.

TABLE II
RCAC HYPERPARAMETERS FOR PHYSICAL FLIGHT EXPERIMENTS.

Controller	P_0	R_u	σ
(7), θ_Θ	0.1	0.001	0.1
(8), θ_Φ	0.1	0.001	-0.1

The mission waypoints are shown in Figure 14. The aircraft is launched by hand from the launch point and is commanded to fly towards point T while climbing to an altitude of 20 m. The aircraft is then commanded to fly around point 2 in a steady-state circular flight with a radius of 20 m for around 1 to 2 minutes. Finally, the aircraft is commanded to land along the green strip. During the takeoff and landing phases, the autopilot is in *stabilized mode*, in

which the bank and elevation commands are issued by a pilot. During the rest of the flight, the autopilot is in *mission mode*, in which the bank and elevation commands are issued by the outer loop of the autopilot.



Fig. 13. Volantex Ranger 1600 fixed-wing RC aircraft used in flight experiments.

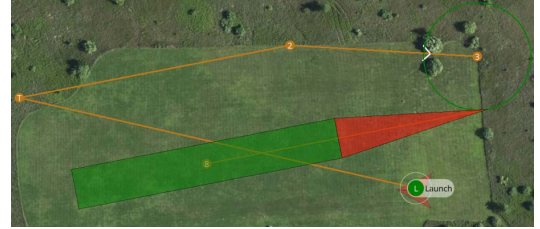


Fig. 14. Waypoints used to construct the flight trajectory in physical flight experiments.

Figures 15, 16 and 17 show the experimental flight test results. As shown in Figure 17, the augmented adaptive autopilot improves the performance over the nominal performance. For $\alpha_d = 0.5$, the trajectory following response degrades substantially with the degraded-nominal autopilot, and in this case, the adaptive autopilot recovers the baseline performance. In fact, as the fixed-gain controller is degraded, RCAC compensates by providing larger values of the corresponding gains. Finally, the adaptive autopilot is also able to learn the gains from a cold start, that is, the case where $\alpha_d = 0$, which switches off the nominal autopilot completely.

VII. CONCLUSIONS

This paper presented an adaptive autopilot that can improve an initial poor choice of controller gains in a fixed-gain autopilot, and learn the autopilot gains without any prior knowledge of the dynamics. The adaptive autopilot is constructed by augmenting the fixed-gain controllers in an autopilot with adaptive controllers. The adaptive autopilot was used to fly a fixed-wing aircraft model in the Gazebo simulator. The adaptive autopilot recovered the performance in the case where the fixed-gain autopilot was degraded and learned a set of gains in the case where the fixed-gain autopilot was completely switched off. Furthermore, the adaptive autopilot improved the trajectory-tracking performance in the case where the aileron was stuck at an unknown angle in simulation. The adaptive autopilot was also used to fly fixed-wing aircraft in flight experiments conducted outdoors. Like the simulation results, the adaptive autopilot improved the flight performance in physical flight experiments.

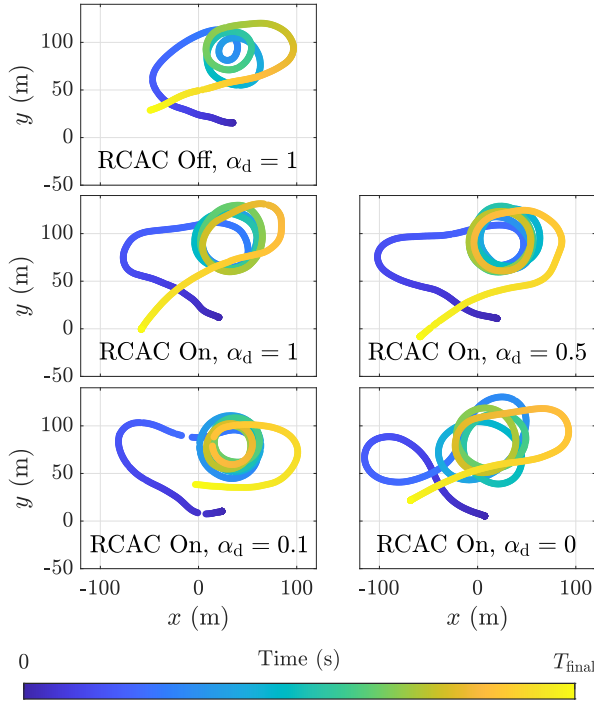


Fig. 15. **Physical flight experiments.** Ground trace of the aircraft with the nominal and adaptive autopilot for several values of the degradation factor α_d .

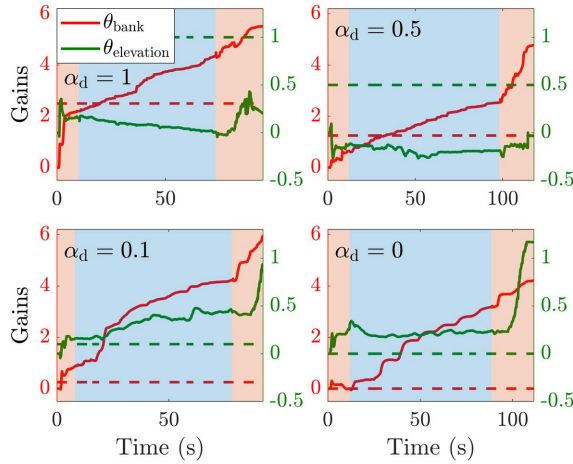


Fig. 16. **Physical flight experiments.** Adaptive bank and elevation controller gains optimized by RCAC in the adaptive autopilot for several values of the degradation factor α_d . The nominal gains are shown in dashes, and the adaptive gains are shown in solid for both the bank and elevator controller. Note that the autopilot is in mission mode and stabilized mode in the blue and orange regions, respectively.

REFERENCES

- [1] N. H. McClamroch, *Steady aircraft flight and performance*. Princeton University Press, 2011.
- [2] N. Nguyen, K. Krishnakumar, J. Kaneshige, and P. Nespeca, "Dynamics and adaptive control for stability recovery of damaged asymmetric aircraft," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2006, p. 6049.
- [3] Z. Yu, H. Badihi, Y. Zhang, Y. Ma, B. Jiang, and C.-Y. Su, "Fractional-order sliding-mode fault-tolerant neural adaptive control of fixed-wing UAV with prescribed tracking performance," in *2020 2nd International Conference on Industrial Artificial Intelligence (IAI)*, 2020, pp. 1–6.

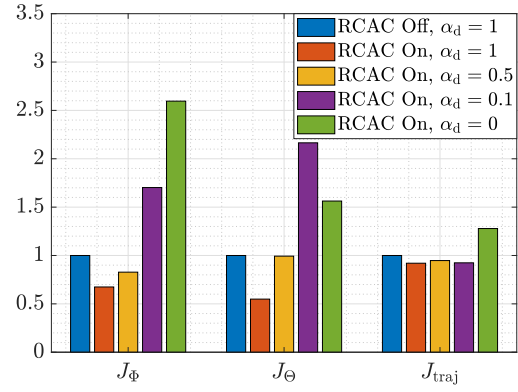


Fig. 17. **Physical flight experiments.** Bank, elevation, and trajectory-tracking error metrics obtained with the adaptive autopilots for several values of α_d and normalized by the corresponding error metric obtained with the nominal autopilot.

- [4] M. Fu, Z. Yu, and Y. Zhang, "Adaptive fault-tolerant control of fixed-wing UAV under actuator saturation and state constraints," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2021, pp. 47–52.
- [5] S. Hirano, K. Uchiyama, and K. Masuda, "Controller design using backstepping algorithm for fixed-wing UAV with thrust vectoring system," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2019, pp. 1084–1088.
- [6] J. Xiong, Y. Yang, Z. Cheng, L. Liu, Y. Wang, and H. Fan, "Observer-like model reference adaptive augmenting based fixed-wing UAV control," in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 6804–6809.
- [7] 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.
- [8] 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.
- [9] 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 *2021 American Control Conference (ACC)*, 2021, pp. 3737–3742.
- [10] J. Spencer, J. Lee, J. A. Paredes, A. Goel, and D. Bernstein, "An adaptive PID autotuner for multicopters with experimental results," in *2022 International Conference on Robotics and Automation (ICRA)*, IEEE, 2022, pp. 7846–7853.
- [11] K. Bruce, J. Kelly, and J. Person, "NASA B737 flight test results of the total energy control system," in *Astrodynamics Conference*, 1986, p. 2143.
- [12] L. Faleiro and A. Lambregts, "Analysis and tuning of a Total Energy Control System control law using eigenstructure assignment," *Aerospace science and technology*, vol. 3, no. 3, pp. 127–140, 1999.
- [13] A. A. Lambregts, "TECS generalized airplane control system design—an update," in *Advances in Aerospace Guidance, Navigation and Control*, Springer, 2013, pp. 503–534.
- [14] M. E. Argyle and R. W. Beard, "Nonlinear total energy control for the longitudinal dynamics of an aircraft," in *2016 American Control Conference (ACC)*, 2016, pp. 6741–6746.
- [15] S. Park, J. Deyst, and J. How, "A new nonlinear guidance logic for trajectory tracking," in *AIAA guidance, navigation, and control conference and exhibit*, 2004, p. 4900.