Robust Hybrid Nonlinear Control Systems for the Dynamics of a Quadcopter Drone

Fendy Santoso, *Member, IEEE*, Matthew A. Garratt, *Member, IEEE*, Sreenatha G. Anavatti, and Ian Petersen, *Fellow, IEEE*

***Abstract*—Robustness in the face of uncertainties is an important aspect in designing high performance control systems. This paper addresses the problem of accurate trajectory tracking of a small quadcopter unmanned aerial vehicle in the face of uncertainties. Accommodating the *worst*-case scenario, we propose a hybrid feedback and feedforward autopilot that has the capability to eliminate the cross-coupling disturbance between the lateral and the longitudinal loops with respect to the vertical loop as well as external disturbances (e.g., wind gusts). The proposed control system leverages on the technical benefits of both the nonlinear model predictive control and the fuzzy feedforward compensator. We highlight the efficacy of our hybrid autopilot system with respect to the performance of the conventional PD control systems through rigorous comparative studies. We also present stability analysis of our hybrid control system.**

***Index Terms*—Fuzzy-feedforward compensator, model predictive control (MPC), quadcopter drone, robustness.**

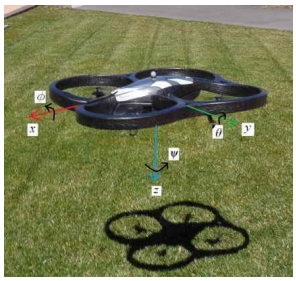

Fig. 1. AR.Drone Quadcopter UAV. The quantities of $[\phi\ \theta\ \psi]^T$ indicate the attitude of the drone, that give the rotation of the rigid body across the $[x\ y\ z]^T$ directions. The drone is equipped with two cameras: a forward and a downward looking, in addition to an ultrasound telemetry sensor, located below the central hull, for altitude measurements. The system is equipped with a 6 degree of freedom MEMS-based inertial measurement unit to stabilize the pitch, roll, and yaw and to assist with the tilting control.

I. Introduction

ROBOTIC aircraft, also known as unmanned aerial vehicles (UAVs) have become popular due to their numerous benefits, leading to increased safety, efficiency, and productivity in both military and civilian domains (e.g., agriculture monitoring [1], animal tracking [2], surveillance [3], [4], border protection [5], forest monitoring [6], and 3-D inspections [7], [8]). To support their maneuvering capability, the development of high performance flight control systems is a critical task.

Based on the configuration of their airframes, UAVs can be classified into three main types: 1) fixed-wing; 2) flapping-wing; and 3) rotary-wing or rotorcraft. For Instance, Fig. 1 depicts the AR.Drone quadcopter as our research aircraft. Unlike fixed-wing aircraft, rotorcraft utilize lift generated by rotary wings or blades and hence they have a unique capability of performing vertical take-off and landing. This ability is very useful, especially for indoor missions, where space is very limited.

Manuscript received March 28, 2018; accepted May 10, 2018. This work was supported by the University of New South Wales, Canberra, Australia. This paper was recommended by Associate Editor S. Nahavandi. *(Corresponding author: Fendy Santoso.)*

The authors are with the School of Engineering and Information Technology, University of New South Wales Canberra, Canberra, ACT 2612, Australia (e-mail: F.Santoso@adfa.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2836922

There are several types of rotorcraft, such as helicopters, quadcopters, cyclocopters, gyrocopters, and gyrodynes. Currently, the most widely implemented indoor UAV platforms are quadcopter drones, mainly due to their simple fixed-pitch motor configuration and small rotors, making them safer than helicopters for operations in confined spaces and in proximity to people [9]. However, a quadcopter is more sensitive to wind gusts than a helicopter of a similar dimension [9].

To successfully achieve a mission, it is essential to have a high level of autonomy supported with the capability of agile maneuvering. Thus, the availability of high performance onboard autopilot is essential. The first challenge in designing a high performance flight control system is to address the robustness issue to accommodate uncertainties [10], [11], such as unpredictable external air flows (e.g., severe wind gusts) and modeling errors. The motion of a small UAV can be highly vulnerable to effects of wind gusts that can force the system to depart from the desired trajectory. This phenomena can also lead to significant overshoots and considerable tracking offsets which are indeed undesirable in light of safety and efficiency issues. Payload restrictions for small UAVs can also limit the practicality of complex control algorithms.

Model predictive control (MPC) is a model-based control system that has numerous advantages as the system can deal with the challenging plant dynamics [12] (e.g., large-time delay, nonminimum phase, and open loop instability), typically present in rotorcraft. The receding horizon principle allows the MPC to predict the behavior of the plant over a finite interval,

before applying the on-line quadratic optimization technique to determine its optimal control sequence. This will lead to the capability of the closed loop control system to handle *multiple constraints* (e.g., actuator and rate limits, in addition to state and output limits), which is an important feature in small robotic platforms. Although in the past, the practical implementation of the MPC was limited to slow varying systems (e.g., chemical process control), recent improvement in computer hardware has led to promising MPC applications for robotic aircraft. However, despite their precise and accurate nature, one major drawback of employing model-based control systems is a reliance on the accuracy of the assumed mathematical model of the system.

Fuzzy logic control, on the other hand, requires no knowledge about the dynamics of the system as it can effectively deal with the imprecise nature of real world physical systems (e.g., uncertainties and ambiguities), owing to its linguistic rules [13] and [14]. Our previous work on fuzzy logic systems for modeling and control of small quadcopters can be found in [15]–[18]. While one can leverage on the advantages of MPC to handle multiple constraints in control loops, the system cannot deal with the influence of *cross-coupling* among loops, also known as the *secondary* control effects. Neglecting this undesirable phenomena is also not a good option, since it can adversely impact the performance of the system.

A. Related Work

Most current works on robust and optimal control systems, designed for small quadcopters, still focus on traditional model-based control techniques. For instance, Nemati and Kumar [19] employed a conventional PD controller to control the attitude of the quadcopter during hover. Emran *et al.* [20] developed robust adaptive control for a varying payload application by means of the model-reference technique. Model-based robust control based on the μ-synthesis technique can be found in the work of Mystkowski [21]–[24].

Meanwhile, Huynh *et al.* [25] proposed $L1$ adaptive control technique to address some robustness issue in outdoor flight environments. Some researchers have implemented intelligent control approaches. However, their approaches are not well-integrated with the model-based control approaches. For example, Drop *et al.* [26] studied a feedforward component in manual control, Fatan *et al.* [27] designed an adaptive neural network controller for attitude stabilization of a quadcopter while Maj and Butkiewicz [28] proposed a fuzzy logic autopilot for flying a multicopter. Although some researchers have investigated the feasibility of having hybrid approaches (e.g., PID/adaptive neuro fuzzy [27], sliding mode/type 2 Fuzzy [29], and PID/LQR [30]), none have explored the advantages of hybrid MPC and FIS controllers to achieve a robust solution.

Owing to the *complementary* characteristics of both the MPC and the fuzzy logic control systems, we present a *hybrid nonlinear autopilot* for trajectory tracking of the AR.Drone quadcopter, consisting of three nonlinear MPCs as the main feedback controllers and a fuzzy inference system (FIS) as a feedforward compensator. To the best of our knowledge, this paper is the first in the literature, exploring the benefits of this type of hybrid nonlinear controllers.

To guarantee the robustness of the proposed control systems, we employ the concept of the *worst*-case scenario of the uncertainties. This way, we can achieve the benefits of both control schemes as the implementation of the fuzzy-feedforward compensator can improve the transient performance of the MPC autopilots by canceling the worst-case cross-coupling signals between the pitch and the roll loops with respect to the altitude dynamics.

The rest of this paper is organized as follows. While Section II discusses dynamic mathematical modeling Section III depicts the architecture of our proposed nonlinear control systems. Meanwhile, Section IV highlights the effectiveness of the proposed control systems via extensive numerical simulations while Section V discusses about stability analysis. Furthermore, Section VI conducts a robustness and optimality study of the closed loop control system while Section VII discusses the noise rejection capability of our control system. Lastly, Section VIII concludes this paper.

II. System Dynamics

In this section, we recall the dynamics of the AR.Drone quadcopter. Designing a high performance autopilot for controlling a quadcopter drone is a rather challenging task considering its underactuated and nonlinear behaviors, coupled with cross-coupling such as between the pitch and the roll loops to the altitude loops.

A. Nonlinear Quadcopter Dynamics

The motion of the system can be represented by the following state variables $x = [p_x\ \dot{p}_x\ p_y\ \dot{p}_y\ p_z\ \dot{p}_z\ \phi\ \dot{\phi}\ \theta\ \dot{\theta}\ \psi\ \dot{\psi}]^T$, where $(p_x,\ p_y,\ p_z)$ indicates the coordinate position of the drone, $(\dot{p}_x,\ \dot{p}_y,\ \dot{p}_z)$ highlights the linear velocity across the (x, y, z) axes while $(\phi,\ \theta,\ \psi)$ indicates the orientation (i.e., roll, pitch, and yaw), respectively; and $(\dot{\phi}\ \dot{\theta}\ \dot{\psi})$ denotes the angular velocity of the drone.

To compute the thrust and the induced velocity in hover and forward flight, we employ the Glaeurt's induced flow model [31], [32] as follows. Given the relative free stream velocity of V_∞, the actuator disc deflects the airstream by the speed of V_i and this will change the downstream flow by $2V_i$. We can break down the flow into the tangential V_t and normal velocities V_n to each rotor disc by adding up the tangential and normal components of V_∞ due to the airflow created by motions of each rotor as a result of its pitching, rolling, and yawing motions.

Accordingly, the closed-form of the thrust as a function of the blade pitch θ_0 can be obtained by considering uniform inflow, that is, V_i remains unchanged with radius or azimuth. The inflow relative to the rotor disk (λ'), and the advance ratio (μ) can be determined by calculating the following element force:

$$T = \frac{\rho a(\Omega R)^2 A_b}{2}\left[\frac{1}{3}\theta_0\left(1+\frac{3}{2}\mu^2\right) - \frac{1}{2}\lambda'\right] \tag{1}$$

where ρ is the absolute density of air, Ω denotes the rotational speed of the blade, R is the blade radius, a is the

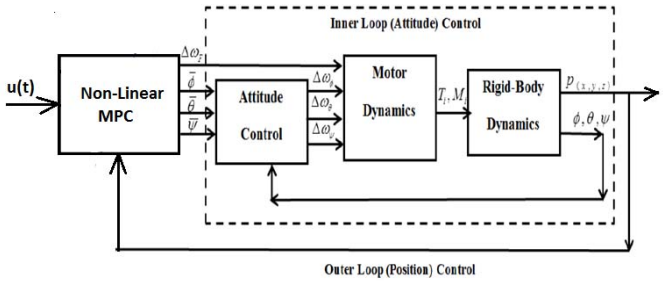

Fig. 2. General overview of our hierarchically structured feedback autopilot, showing a relation between the inner (attitude) and the outer (position) loop control systems. The manufacturer has equipped the system with a stable and fixed inner loop control system while the outer loop can be reprogrammed to meet the objectives of the designers.

lift curve slope, A_b is the total blade area (N blades) while $\lambda' = (V_i + V_n/\Omega R)$ and $\mu = (V_t/\Omega R)$.

We assume our UAV is a rigid body as with the conventional mass distribution. Thus, we can employ Newton's second law to derive the relationship between forces and moments, in addition to linear and angular accelerations. The aircraft is assumed to be symmetrical around the xz plane, so that $I_{yz} = 0,\ I_{xy} = 0$. Full equations of the rigid body dynamics can be found in the flight mechanics by Nelson [34].

B. MIMO Transfer Function

To begin with, a block diagram of the proposed feedback control system is depicted in Fig. 2. It comprises of two main control loops, namely, the inner loop for attitude control, which is made fixed by the manufacturer; and the outer loop, which can be programmed to meet the design objective. This motivates us to propose a hybrid and robust nonlinear autopilot using the complementary benefits of MPC and fuzzy logic control systems.

The AR.Drone quadcopter can be divided into three main control loops, namely, the longitudinal the lateral as well as the vertical (altitude) dynamics. In an earlier study conducted in [9], the authors derived accurate linear MIMO transfer function of the AR.Drone quadcopter by means of the *system identification* method, achieving an accuracy beyond 90% for attitude deviations less than 30 deg [9].

In most cases, however, it is sufficient to represent the nonlinear dynamics of a quadcopter using its linearized model around its *hover* point, where $\bar{\omega_1} = \bar{\omega_2} = \bar{\omega_3} = \bar{\omega_4} = \omega_h$, where $\bar{\omega_h} = mg/4k_F$ (k_F is the force constant of the motor) with $\bar{\phi} = \bar{\theta} = 0$, and $\psi = \psi_0$, leading to the following transfer function models [9].

The transfer function of the horizontal velocity v_x with respect to pitch θ_{in} in the continuous domain is given by

$$G_x(s) = \frac{v_x(s)}{\theta_{in}(s)} = \frac{0.7995}{s^2 + 4.1881s + 7.1429}. \tag{2}$$

The transfer function of the horizontal velocity v_y with respect to roll ϕ_{in} is given by

$$G_y(s) = \frac{v_y(s)}{\phi_{in}(s)} = \frac{116.24s + 67.2488}{1000s^2 + 982.664s + 301.338}. \tag{3}$$

TABLE I
INNER LOOP FLIGHT CHARACTERISTICS

TF	Poles	Damping Factor	Natural Frequency
(1)	$s_{1,2} = -2.0941 \pm 1.6607i$	0.784	2.67
(2)	$s_{1,2} = -0.491 \pm 0.245i$	0.895	0.549
(3)	$s_{1,2} = -5.1 \pm 2.73i$	0.882	5.78
	$s_{3,4} = -0.705 \pm 4.48i$	0.155	4.54.

The transfer function of the vertical velocity v_z with respect to its input variable $v_{z_{in}}$ is given by

$$G_z(s) = \frac{v_z(s)}{v_{z_{in}}(s)} = \frac{17.6902s^2 + 37.7142s + 422.62}{s^4 + 11.6091s^3 + 68.394s^2 + 257s + 688.244}. \tag{4}$$

The transfer functions in (2)–(4) lead to the following system characteristics as depicted in Table I. As can be seen, the inner loop control system is already stable and has an average damping factor around 0.8, indicating an underdamped response.

Considering the discrete nature of the MPC systems, we convert (2)–(4) to their discrete time counterparts using the zero order hold method to obtain the following equations:

$$G_x(z) = \frac{v_x(z)}{\theta_{in}(z)} = \frac{3.9223e-05(z + 0.9861)}{\left(z^2 - 1.958z + 0.959\right)} \tag{5}$$

$$G_y(z) = \frac{v_y(z)}{\phi_{in}(z)} = \frac{0.0011597(z - 0.9942)}{\left(z^2 - 1.99z + 0.9902\right)} \tag{6}$$

$$G_z(z) = \frac{v_z(z)}{v_{zin}(z)} = \frac{0.00085713(z + 0.9689)\left(z^2 - 1.977z + 0.9789\right)}{\left(z^2 - 1.9z + 0.9032\right)\left(z^2 - 1.984z + 0.9858\right)}. \tag{7}$$

It is clear that the inner loop attitude transfer functions in (5)–(7) are stable since all poles are located inside the unit circle of the z domain. However, we assume that the cross coupling transfer functions between pitch and roll with respect to altitude are not directly available. Thus, one cannot employ model-based control to cancel it.

III. NONLINEAR CONTROL SYSTEMS ARCHITECTURE

The architecture of the proposed control system is elaborated in Fig. 3. As can be seen, there are three major loops for controlling (x, y, z) position. While feedback control is performed by means of the MPC technique, the interloop cross-coupling effects are compensated by means of the fuzzy-feedforward control system since we can directly measure $\theta_{in}(kT)$ and $\phi_{in}(kT)$, $\forall k \geq 0$.

Secondary effect is an aviation term, referring to an undesirable implication of the primary effect. For instance, in the context of a fixed-wing aircraft, the secondary effect of yaw is roll. Meanwhile, in the context of the AR.Drone quadcopter, it refers to the relation of the *pitch* and the *roll* angles with respect to the *altitude drop* due to a significant loss of the vertical thrust component [9]. In addition, the presence of the *downward force* due to the *incident wings* can also exacerbate this phenomena. This adverse control effect remains uncompensated by the manufacturer and hence its effect is significant [9].

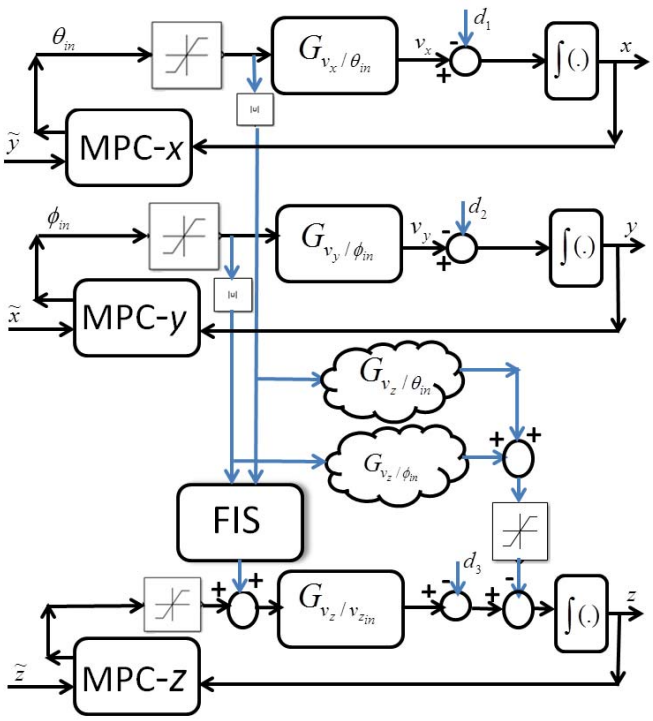

Fig. 3. Detailed architecture of the proposed control system. Given reference 3-D position $(\tilde{x}, \tilde{y}, \tilde{z})$, MPC-$(x, y, z)$ denotes the roles of each MPC autopilot in controlling the 3-D position of the drone. FIS indicates the *fuzzy inference systems* compensator, whose role is to provide feedforward compensation to the altitude loop. $G(.)$ denotes a certain transfer function, while $d_i = 1, 2, 3$ represent external disturbance, e.g., wind gusts.

Since the mathematical models of the cross-coupling effects may not be directly available, it is therefore useful to employ the concept of a model-free control system. One well-known approach to address this issue is to employ the concept of FIS. Our fuzzy-feedforward autopilot is easy to be understood in the absence of complex mathematical models. In addition, the system is also flexible and accurate to compensate the undesirable altitude drop.

Feedforward control is an effective strategy to compensate for disturbances, especially if the corresponding input variables can be directly measured. This method of compensation is related to the principle of a gust alleviation system, which is a control system in fly-by-wire aircraft attempting to compensate for the adverse effects of gusts on the aircraft. The system measures the upward acceleration of the aircraft (sensed at the center of gravity) and compares it with the acceleration set point. A control system determines the deflections of the control surfaces to counteract the accelerations caused by wind gusts. This way, the performance of a small UAV in the face of unpredictable wind gusts can be significantly improved [35]. We will discuss the control algorithms in more detail in the following section.

A. Model Predictive Control—The Receding Horizon Concept

In this section, we will design a regulatory control that employs the dynamics of the response of the process variables to force the controlled variables to a prespecified trajectory

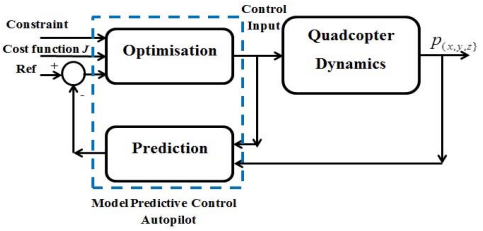

Fig. 4. Block diagram of the MPC autopilot highlighting two major processes, namely, prediction and optimization.

from their current operating point to reach a specific target. Known as finite-time-horizon prediction, MPC is a well-developed optimization process. Conceptually, there are two major processes in MPC as illustrated in Fig. 4. While the first process deals with the optimization to minimize the quadratic performance index as in (8) and (9), the second part performs prediction using a set of difference equations within a certain prediction horizon N.

In the context of the finite-horizon optimal control, one seeks to minimize the cost function as follows [36]:

$$u(k)^* = \arg \min_{u(.)} J\{u(k)\} \ \forall k \geq 0. \tag{8}$$

We define the mathematical model of our plant using a discrete state space equation $x(k+1) = f(x_k, u_k)$. Based on the concept of the finite horizon optimization, we sample and compute the states of the plant and its associated cost function for a short time window $[t, \ t+T]$. We also define the future input sequence $u_k = [u_{k|k} \ u_{k+1|k} \ \ldots \ u_{k+N-1|k}]^T$ as well as the future state trajectory $x_k = [x_{k|k} \ x_{k+1|k} \ \ldots \ x_{k+N-1|k}]^T$.

In more detail, given the current state of the system x and a fixed length of the prediction horizon N, MPC computes the future states of the systems to predict the optimal sequence of the control commands u for $\tilde{x}$ and $\tilde{u}$ that minimizes a quadratic cost function given by [7]:

$$\begin{aligned} J(x) = & \frac{1}{2} x_N^T P x_N + \frac{1}{2} \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k \\ & u_k \in U, x_o = x \\ & \text{subject to: } |u_k| \leq \tilde{u}, |x_k| \leq \tilde{x} \\ & \text{and } x_{k+1} = f(k, x_k, u_k) + g(k, x_k) \quad \forall k \end{aligned} \tag{9}$$

where N indicates the horizon, x_k are the state variables or their predicted values, u_k depicts the control input, P denotes a terminal penalty matrix, while Q indicates a state weighting matrix, and R denotes a control weighting matrix.

In practice, one can numerically solve (8) and (9) to obtain the optimum control signal $u(k)^*$ by means of *the convex quadratic programming* (QP) technique. Several numerical programming techniques can be employed to solve the QP problem [12] (e.g., the active set methods [37] or the interior point methods [38]) to perform online MPC optimization techniques.

There are multiple benefits of employing MPC autopilots for trajectory tracking applications [39]. First, the concept is straight-forward and also capable of handling numerous constraints such as *actuator saturation and bandwidth limits*, which is a very important issue for small UAVs. Second, the

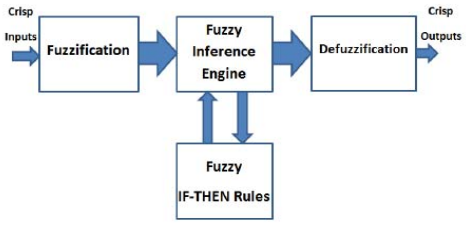

Fig. 5. Block diagram of the FIS, comprising of the fuzzification, rule-based system, and the defuzzification process.

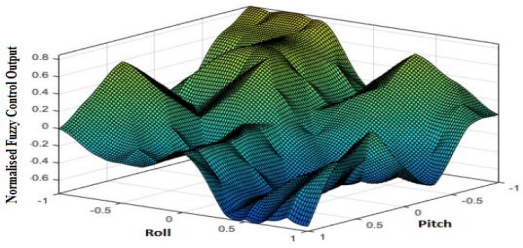

Fig. 6. Fuzzy control surface, depicting the input–output relation of the fuzzy feedforward compensator.

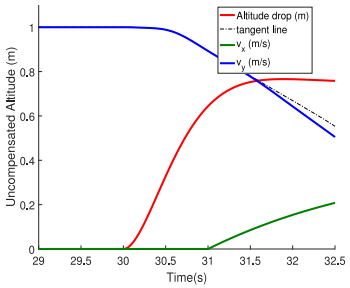

Fig. 7. Uncompensated altitude drop of the quadcopter drone. The blue curve denotes the undesirable altitude drop of the AR.drone quadcopter due to the internal cross-coupling between the lateral and the longitudinal motions; the red and green curves indicate v_x and v_y, respectively.

MPC algorithm is also easy to maintain and promotes uniformity (the same technique is used for a wide range of problems). Thus, one does not need to redesign the whole system, when the model or specification of the plant changes, and this also can be done automatically in-flight. MPC predicts the future control input to make the future response of the system well-behaved in line with the desired trajectory given in a specific time frame N corresponding to the moving horizon.

The suitability of MPC to support accurate trajectory tracking applications can be achieved through the selection of an appropriate cost function in (9) to accommodate the desired trajectory of the vehicle given actuator limitations. Thus, one can employ the cost function given in (9) to "punish" *sudden attitude changes* using the weighting matrices $Q^T \geq 0$ and $R^T > 0$ Meanwhile, a terminal penalty matrix P can be used to reduce the expended control effort [7].

B. Robust Fuzzy-Feedforward Compensator

In addition to feedback MPC, we employ a fuzzy feedforward autopilot to compensate for the undesirable cross-coupling between the pitch and the roll loops in the altitude dynamics, which ideally should remain independent. While pitch is directly related to the velocity along the x-axis, roll is used to control the velocity along the y-axis. However, creating pitch and roll will lead to a secondary control effect, also known as the cross-coupling effect to the vertical (altitude) loop. Thus, we use the pitch and the roll angles as the control inputs in our fuzzy compensator to estimate the required feedforward signal.

The reason to employ a FIS is due to its ability to handle the imprecise nature of abstract concepts such as the secondary effects of control between pitch and roll dynamics in the altitude loop, without having to know their exact mathematical models.

Fuzzy systems rely on the approximate reasoning that derives a conclusion based on a set of expert fuzzy IF-THEN rules, reflected in the form of a continuous value between 0 and 1. We employ a FIS based upon the Mamdani rules. The system consists of three fundamental constituents, namely, *fuzzification*, *rule based-inference*, and *defuzzification* (see Fig 5). The fuzzy mapping can be regarded as a mapping from *crisp inputs* into *fuzzy sets* while the *defuzzification* process is intended to reconvert the fuzzy output values from the crisp values. Given linguistic control rules, fuzzy logic imitates human reasoning and decision making using natural language.

We employ six Gaussian membership functions, equally spaced between -1 and 1, for both the pitch and the roll input variables as well as the output variable. The relationship between the input variables and the control output is represented in Fig. 6, whose purpose is to compensate the Gaussian-like *cross-coupling* signal in the vertical loop due to the lateral and the longitudinal motions.

Our control design is based on the concept of the *worst case scenario* to guarantee the *robustness* of the closed loop system. That is, we account for the worst case scenario of the cross-coupling disturbance and the noise signals. To tune the performance of our fuzzy feedforward compensator, we define a fuzzy gain G_F that can be used to set the desired magnitude of the compensator signal u_F to achieve the desired settling time t_s. We can further fine tune G_F to improve the closed loop performance of the vertical loop to achieve the desired settling time. This way, we can aim not only to cancel the cross coupling disturbance signal $\gamma(k)$ given $u_F(k) = -\gamma(k), k \geq 0$ $\forall k$, but also to expedite the transient response of the closed loop control systems by setting $|u_F(k)| > |\gamma(k)|, k \geq 0, \forall k$.

IV. Computer Simulations

In this section, we will present the performance of our closed loop control systems, highlighting the effectiveness of the proposed autopilot.

A. Inner (Attitude) Loop Response

To begin with, the open loop attitude-to-position response is depicted in Fig. 7. As can be seen, when the quadcopter drone performs pitching and rolling motions, the uncompensated altitude drops considerably. This offset is significant and hence should not be ignored.

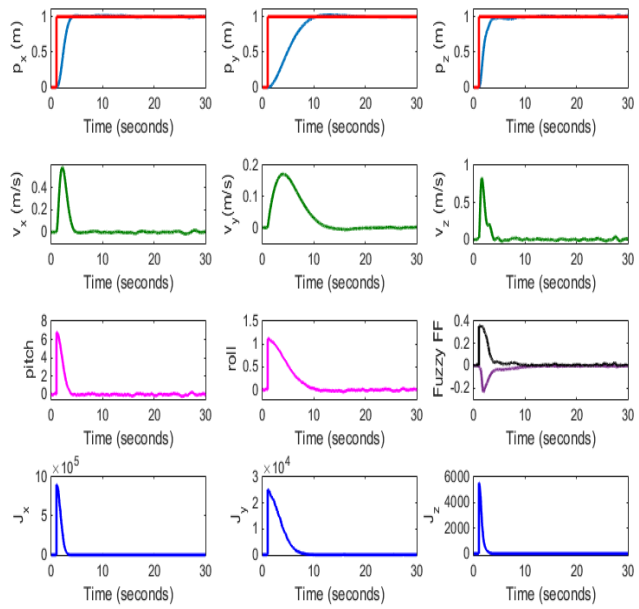

Fig. 8. Hovering performance of the closed loop control system. (a) First row: $p_{x,y,z}$. (b) Second row: $v_{x,y,z}$. (c) Third row: pitch (left), roll (middle), fuzzy feedforward control signal versus cross coupling disturbance signal (right). (d) Fourth row: performance index.

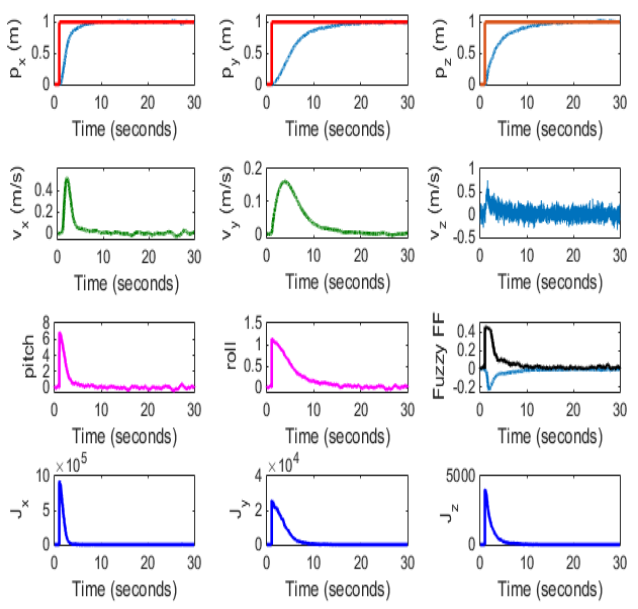

Fig. 9. Time domain performance in the face of modeling error of the hybrid control system. (a) First row: $p_{x,y,z}$. (b) Second row: $v_{x,y,z}$. (c) Third row: pitch (left), roll (middle), fuzzy feedforward control signal versus cross coupling disturbance signal (right). (d) Fourth row: performance index. In this scenario, we introduce uncertainty in the nominal transfer function by 20% from the nominal pole locations.

B. Closed Loop Hovering Performance Under Uncertainties

In what follows, we will investigate the performance of the closed loop control system in the case of hovering, where the drone is expected to hold a constant altitude and position. To make the scenario more realistic, we have included measurement noise in the form of white noise with a magnitude of $\pm(10-20)\%$ of the operating point of the system.

As can be seen from Fig. 8, the undesirable secondary effects of the attitude loop have been successfully minimized by means of the fuzzy-feedforward compensator based on the input pitch and roll measurements. It is apparent that the desired lateral position and altitude have been achieved within a reasonably short settling time.

The top figures show $p_{(x,y,z)}$ while the second row figures show $v_{(x,y,z)}$. Furthermore, the third row figures (left and middle) present the pitch and roll angle while the right figures denote the fuzzy feedforward compensation signal (light blue) compared to the undesirable secondary control effects due to the lateral motions of the drone. Next, the fourth row figures illustrate how the quadratic cost function $J_{(x,y,z)}$ is being minimized as time progresses. Lastly, the bottom figures show the optimal control sequences, $u^*_{(x,y,z)}$, for both the lateral and the longitudinal motions.

C. Closed Loop Hovering Performance in the Face of Parameter Variations

In this section, we will investigate the performance of the closed loop control system in the face of modeling error. To introduce uncertainty, we employ a perturbed dynamic model. In our perturbed models, the poles of the attitude loops have been shifted by $\Delta = \pm 20\%$ from the nominal one by altering the damping factor and natural frequency of the system. As shown in Fig. 9, the closed loop control system still demonstrates good robustness in the face of modeling error as the uncertainty does not significantly affect the performance of it similar to Fig. 8.

D. Closed Loop Trajectory Tracking Performance

The performance of the closed loop control system for multiple set points trajectory tracking is given in Fig. 10. Subsequently, we also investigate the tracking performance of the closed loop control system under the effects of a sinusoidal set point. It is clear that the closed loop control system works reasonably well to track multiple constants and varying set points as in the case of sine-wave tracking in Figs. 10 and 11.

To observe the secondary effects, we also perform multiple set point tracking with sinusoidal signals while the system hovers, meaning a constant altitude set point for the vertical loop. As can be seen from Fig. 11, the closed loop control system demonstrates reasonably good tracking performance.

E. System Performance Under Nonlinear Dynamics

We also investigated the efficacy of our control system using the nonlinear dynamics of the quadcopter, described in Section II A. We perform nonlinear simulation using the *C* code in Simulink S-function. S-function is a block diagram component in Simulink that provides access to user defined functions written in either C or MATLAB m-file format. The Simulink software calls the S-functions when the outputs and state derivatives corresponding to the S-function need to be updated.

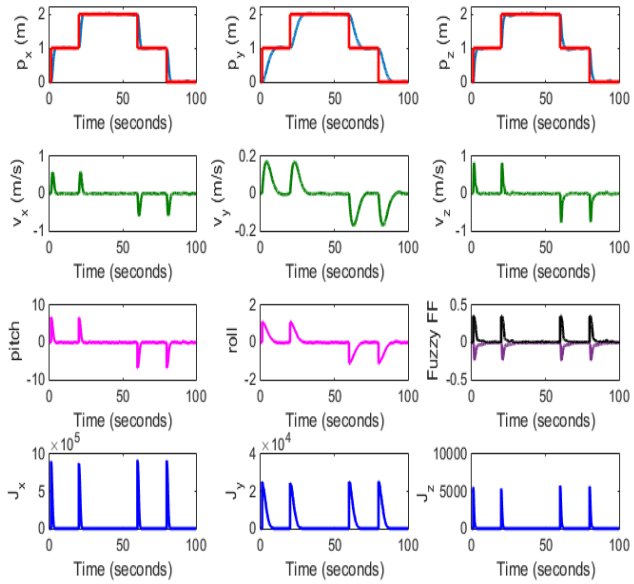

Fig. 10. Multiple set point tracking of the hybrid control system for pitch, roll, and altitude loops. (a) First row: $p_{x,y,z}$. (b) Second row: $v_{x,y,z}$. (c) Third row: pitch (left), roll (middle), fuzzy feedforward control signal versus cross coupling disturbance signal (right). (d) Fourth row: performance index.

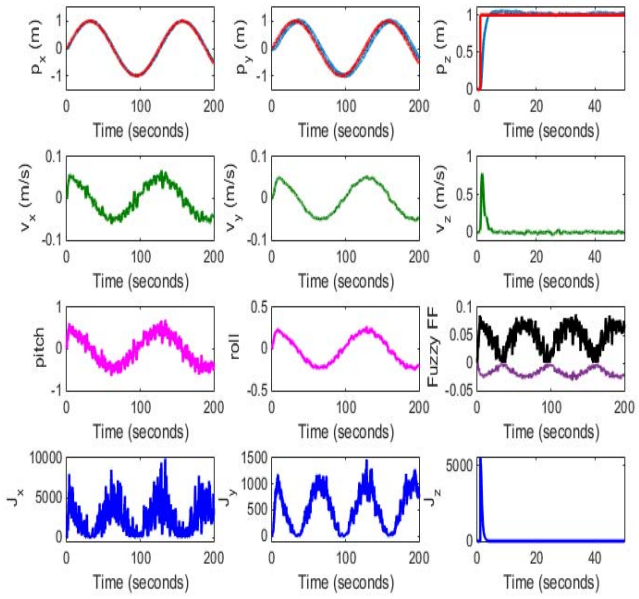

Fig. 11. Multiple set points tracking of the hybrid control system with a sine wave reference with $G_F = 0.4$. (a) First row: $p_{x,y,z}$. (b) Second row: $v_{x,y,z}$. (c) Third row: pitch (left), roll (middle), fuzzy feedforward control signal versus cross coupling disturbance signal (right). (d) Fourth row: performance index.

The states of the system dynamics are the position and the local velocity components in the body axes, rotational rates, and the quaternion attitude. Inputs to the block are the forces and moments, acting on the drone while the outputs are accelerations, local velocities, position, body angular rates, and attitude. A simple trapezoidal integration scheme executing at 2000 updates per second is used to update the states.

Fig. 12 clearly indicates that our control system can work reasonably well using nonlinear dynamics of the drone. This

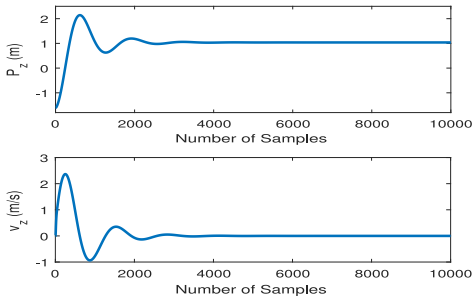

Fig. 12. Altitude holding performance using nonlinear dynamics of the quadcopter with an optimization horizon $N = 5$.

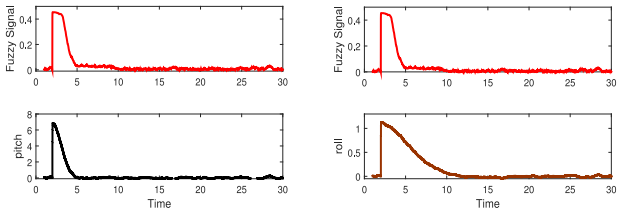

Fig. 13. Input (bottom)–output (top) fuzzy feedforward data for modeling.

also clearly highlights its robustness and suitability for real time applications. The same principle also works for the rest of the control loops.

V. Stability Analysis

This section deals with stability analysis of the closed loop control systems in the lateral, longitudinal, and vertical loops.

A. Fuzzy Feedforward Compensator

To conduct stability analysis of the fuzzy feedforward compensator, we will model our proposed fuzzy feedforward compensator using a transfer function matrix as follows:

$$u_F(z) = \begin{pmatrix} G_1(z) & G_2(z) \end{pmatrix} \begin{pmatrix} \theta_{in} \\ \phi_{in} \end{pmatrix}$$

where $G_1(z)$ indicates the relation between pitch and feedforward control signal and $G_2(z)$ denotes the relation between roll and the corresponding feedforward control signal.

To obtain $G_1(z)$ and $G_2(z)$, we employ a system identification technique, where we record the input and output experimental data. We employ a MIMO autoregressive-exogenous input technique as follows [40]:

$$A\left(z^{-1}\right)y = z^{-k}B\left(z^{-1}\right)u + e \tag{10}$$

where z is the one step forward operator $y := u_F$ and $u := [\theta_{in}\ \phi_{in}]^T$, and $k \geq 0$ denotes step delay while e indicates the modeling error. Thus, the identified transfer function is given by: $G_{1,2} = A^{-1}B(z) = (\text{adj}(A)B/|A|)$.

Typical input and output fuzzy signals for the purpose of system identification are given in Fig. 13. While Fig. 13 left indicates the relation between pitch input and output fuzzy signal, Fig. 13 right illustrates the relation between roll input and fuzzy output signal.

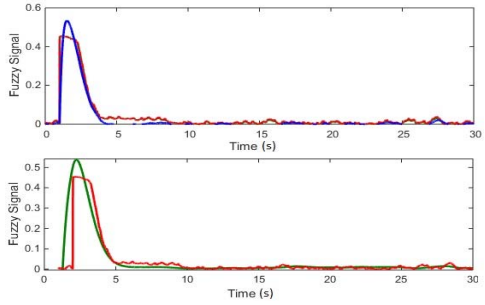

Fig. 14. Model validation: (top) $G_1(z)$, (bottom) $G_2(z)$. The validation data takes into account the worst case scenario of noise and disturbance.

Employing (10), we finally arrive at the following discrete transfer functions:

$$G_1(z) \approx \frac{0.009343z^{-1}}{1 + 0.01609z^{-1} - 0.9107z^{-2}} \tag{11}$$

which gives one zero at $z = 0$ and two real poles at $z_1 = -0.9624$ and $z_2 = 0.9463$. Based on the location of the poles of (11), it is clear that their radius ρ is within the unit circle, that is, $\rho \leq 1$.

Furthermore, the value of $G_2(z)$ is given by

$$G_2(z) \approx \frac{-1.369e-05z^{-1}}{1 - 1.98z^{-1} + 0.9802z^{-2}}. \tag{12}$$

This will give one zero at $z = 0$ and complex imaginary poles at $z_{1,2} = 0.9900 \pm 0.0100i$. The transfer function in (12) indicates a stable system, as given by $\rho \leq 1$.

To highlight the accuracy of the acquired model of our fuzzy feedforward compensator, we validate our $G_1(z)$ and $G_2(z)$ using a set of numerical data affected by the worst case noise scenario. As can be seen in Fig. 14, our model still can fit the output data reasonably well, indicating the robustness of the proposed mathematical model with respect to the modeling error.

B. MPC Systems: The Receding Horizon Control

In this section, we conduct stability analysis of our nonlinear MPC via an unconstrained optimization method. In the absence of system constraints in (9), the problem of minimizing the quadratic cost function in (8) will become a linear quadratic optimization, which can be simply mathematically formulated as follows: $(\partial J/\partial U) = 0$. This will give a good linear approximation to analyze the closed loop control systems of our nonlinear MPC.

Given a linear time-invariant mode: $x_{i+1|k} + Ax_{i|k} + Bu_{i|k}$ and under an assumption that x_k is measured at $k = 0, 1, ...$, we can rewrite the following prediction equation: $x_{0|k} = x_k$, $x_{1|k} = Ax_k + Bu_{0|k}, \ldots,$ $x_{N|k} = A^N x_k + A^{N-1}Bu_{0|k} + A^{N-2}Bu_{1|k} + ... + Bu_{N-1|k}$, leading to the general form of equation:

$$\mathbf{x}_k = \mathscr{M}x_k + \mathscr{C}\mathbf{u}_k \tag{13}$$

where

$$\mathscr{C} = \begin{pmatrix} B & 0 & \ldots & 0 \\ AB & B & \ldots & 0 \\ \vdots & \vdots & \ldots & \vdots \\ A^{N-1}B & A^{N-2}B & \ldots & B \end{pmatrix} \mathscr{M} = \begin{pmatrix} A \\ A^2 \\ \vdots \\ A^N \end{pmatrix}.$$

Accordingly, the predicted output can be formulated using the following vector equations:

$$\begin{aligned} X &= \left[\hat{y}(k+1|k)\right] = C\hat{x}(k+1|k) \\ &= CAx(k) + \sum_{j=1}^{i} CAB\hat{u}(k+1-j|k). \end{aligned} \tag{14}$$

This leads to following mathematical expression:

$$\begin{pmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \vdots \\ \hat{y}(k+N) \end{pmatrix} = \begin{pmatrix} CA \\ CA^2 \\ \vdots \\ CA^N \end{pmatrix} x(k) + \begin{pmatrix} CB & 0 & \ldots & 0 \\ CAB & CB & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \ldots & CB \end{pmatrix} \times \begin{pmatrix} \hat{u}(k) \\ \hat{u}(k+1) \\ \vdots \\ \hat{u}(k+N-1) \end{pmatrix}. \tag{15}$$

From (15), one can easily estimate the output of the system given the control signal u and horizon optimization N.

By differentiating the quadratic cost function in (8), where $J = \mathbf{u}^T H\mathbf{u} + 2x^T F^T\mathbf{u} + x^T Gx$, we finally arrive at $\nabla_u J = 2(Hu + Fx) = 0$. This results in $\mathbf{u}^* = -H^{-1}Fx$ provided $H := C^T\mathbf{Q}C + R \succ 0$, that is, if $\mathbf{R} \succ 0, (Q, P) \succ 0$ or $R \succ 0, (Q, P) \succ 0$ and C is full-rank $\Leftrightarrow (A, B)$ is controllable. The final form of receding horizon controller in state feedback form is given by

$$u_k = -[I\ 0\ \cdots\ 0]H^{-1}Fx_k \tag{16}$$

where $F := C^T\mathbf{Q}M$. For a specific prediction horizon N, this implies $k_N = H^{-1}F$ which will lead the closed loop eigen values becoming $\lambda(A + Bk_N)$. This dictates the stability and performance of the closed loop control system.

We finally arrive at the following design parameters. For the roll loop MPC, we specify $Q = 50, R = 1$ while for the pitch loop we have $Q = 300, R = 1$. Meanwhile, for the vertical loop we set $Q = 20, R = 1$. For pitch loop, we define the following constraints: $|\theta| \leq \pi/6$ rad, $|\dot{\theta}| \leq \pi/4$ rad/s, $|p_x| \leq 3$ m while for roll loop we specify $|\phi| \leq \pi/3$ rad, $|\dot{\phi}| \leq \pi/4.5$ rad/s, $|p_y| \leq 3$ m. Moreover, for the vertical motion we specify $|v_{z_{in}}| \leq 25$ m/s, $p_z \leq 3$ m.

We will investigate the effect of N with respect to the closed loop control performance. First, the values of state feedback $k_1 \cdots k_n$ with respect to N is given as follows. Fig. 15 indicates the values of the state feedback gain k_1, k_2, k_3 of the pitch loop. Fig. 16 indicates the values of the state feedback gain k_1, k_2, k_3 of the roll loop. Fig. 17 indicates the values of the state feedback gain k_1, k_2, $k_3, \ldots,$ k_5 of the vertical loop. As can be seen, higher values of N leads to bigger values of $|k_n|$, which also leads to better stability margin and time domain performance as more information can be acquired

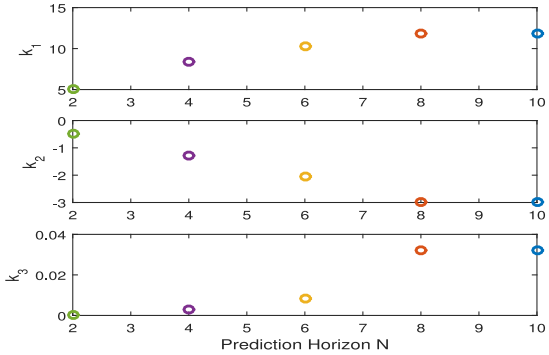

Fig. 15. State feedback gain $k = [k_1\ k_2\ k_3]^T$ as a function of the optimization horizon N for pitch loop.

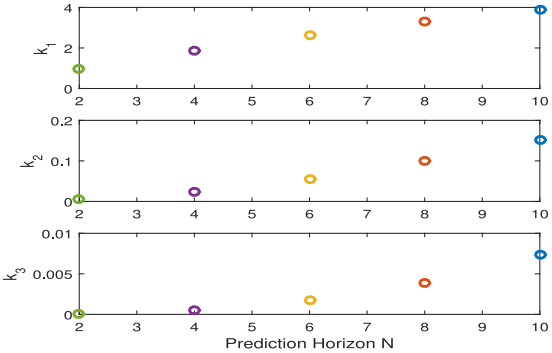

Fig. 16. State feedback gain $k = [k_1\ k_2\ k_3]^T$ as a function of horizon optimization N for roll loop.

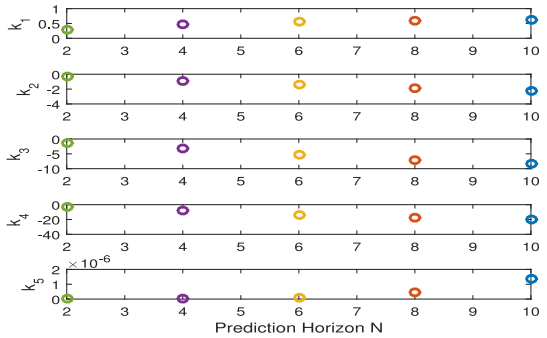

Fig. 17. State feedback gain $k = [k_1\ k_2\ k_3\ k_4\ k_5]^T$ as a function of the optimization horizon N for the vertical loop.

by the system. Conversely, narrow values of N may lead to unstable systems.

Fig. 18 depicts the location of the closed loop poles in comparison to their open loop counterparts. As can be seen, the MPC autopilots have suppressed the overshoots by minimizing the imaginary part of the closed loop control system poles. For the pitch loop, the closed loop poles $\lambda(A-Bk)$ for $N = 10$ are given by $z_1 = 1$, $z_{2,3} = 0.9791 \pm 0.0163i$ while for the roll loop, they are given by $z_1 = 0.9526$, $z_2 = 0.999$, $z_3 = 0.9998$. Furthermore for the vertical loop, the closed loop control system's poles are given by $z_{1,2} = 0.945 \pm 0.019i$, $z_{3,4} = 0.9937 \pm 0.04289i$ and $z_5 = 0.99999$, indicating negligible overshoots.

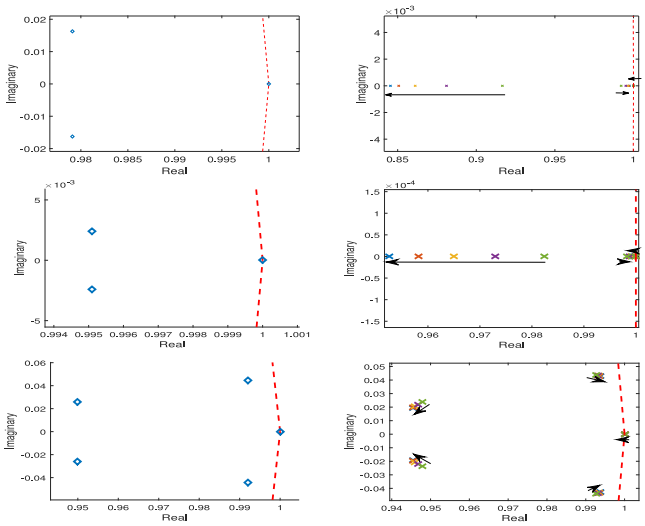

Fig. 18. Open loop (left) and closed loop poles using MPC (right). (a) Pitch loop (top), (b) roll loop (middle), and (c) altitude (vertical) loop (bottom). The arrows indicate the position of the poles (root locus) as N is increased iteratively from $N = 2$ to $N = 10$.

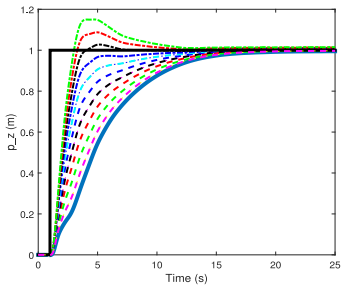

Fig. 19. Effect of fuzzy feedforward gain on the performance of the closed loop altitude control system. The smooth blue curve indicates the response of the closed loop altitude loop with an *MPC only controller* while the dotted lines indicate the response of the *hybrid* closed loop altitude loop, highlighting the effectiveness of the fuzzy feedforward compensator. The fuzzy gain varies from $G_F = 0.1$ to $G_F = 1$ with an incremental step of 0.1.

VI. Optimality and Robustness Analysis

In this section, we discuss the optimality and robustness of the proposed control schemes.

A. Fuzzy Feedforward Compensator

We fine tune the gain of our fuzzy feedforward controller G_F and measure the performance of the closed loop control system based on the quadratic index performance as follows: $J_F = \sum e_z^2(k)$, $\forall k \geq 0$. While the effect of G_F on the performance of the closed loop altitude control system is given in Fig. 19, the corresponding magnitude of u_F with respect to the worst case cross-coupling disturbance signal is given in Fig. 20. Meanwhile, an optimality study of our G_F with respect to J_F is given in Fig. 21 while a study regarding the relative improvement of the system transient response with respect to MPC only control is given in Fig. 21 right and Table II.

Fig. 21 clearly indicates that the optimum steady state response, that is, the one that has minimum area of the error signal, can be achieved at $G_F = 0.9$, with around 70% improvement compared to the performance of the closed loop

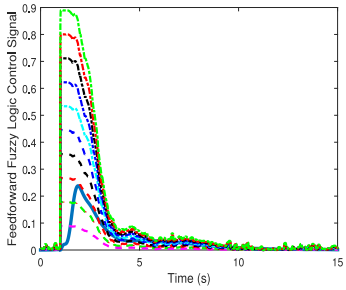

Fig. 20. Worst case scenario study: comparison between the magnitude of the fuzzy feedforward control signals $u_F(t)$ (represented by dotted lines) with respect to the secondary control effects, acting as a disturbance due to pitch and roll deflections $\gamma(t)$ (represented by the smooth blue curve). The system was investigated under varying fuzzy gains, starting from $G_F = 0.1$ to $G_F = 1$ with an incremental step of 0.1.

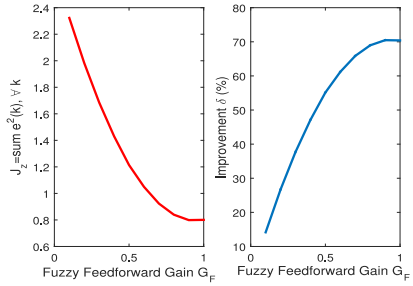

Fig. 21. Optimality analysis of the fuzzy logic feedforward compensator. While the left figure indicates how the error signal decreases as the fuzzy gain increases, the right figure denotes the percentage of improvement in transient response with respect to MPC only feedback control.

TABLE II
FUZZY GAIN VERSUS ERROR SIGNAL

No	G_F	$\int e(t)^2 dt$	$\int \lvert u_F(t)\rvert dt$	α	$\delta(\%)$
1	0.0	2.705698	0	0	0
2	0.1	2.322857	0.194212	0.3833	14.15
3	0.2	1.982756	0.388423	0.7666	26.72
4	0.3	1.685396	0.582635	1.1498	37.71
5	0.4	1.430775	0.776847	1.5331	47.12
6	0.5	1.213794	0.971058	1.9164	55.14
7	0.6	1.049756	1.165270	2.2997	61.20
8	0.7	0.923357	1.359482	2.6830	65.87
9	0.8	0.839698	1.553693	3.0663	68.97
10	0.9	0.798780	1.747905	3.4495	70.48
11	1.0	0.800602	1.942117	3.8328	70.41

control system of the altitude loop without the fuzzy feedforward compensator. While the further increase of the fuzzy gain beyond this point will result in greater overshoots, leading to the growth of the area of the error signal, any smaller values than $G_F = 0.9$ leads to reduced performance. However, in practice, suboptimal performance given by $0.7 \leq G_F \leq 0.8$ may be desirable in the absence of overshoots in altitude leading to safer flight performance.

Moreover, the magnitude of the fuzzy feedforward compensator $u_F(k)$ with respect to the undesirable cross coupling signal $\gamma(k)$, as a function of G_F, is given in Fig. 20. By choosing a certain G_F so that $u_F(k) > (\gamma(k) + \Delta(k))$, $\forall k$, where $\Delta(k)$ indicates a safety margin to accommodate the uncertainties. This notion can be used to accommodate the worst case scenario of the cross coupling disturbance signal, so that one can guarantee the robustness of the closed loop control

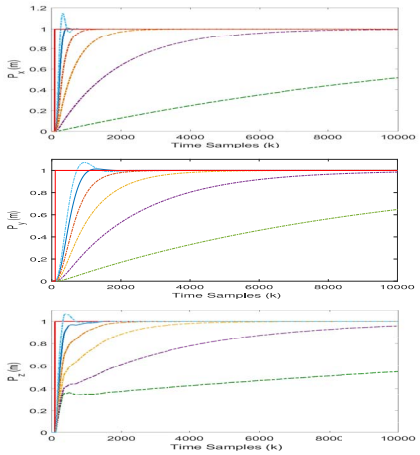

Fig. 22. Linear approximation of our MPC system in the *absence* of disturbance given various optimization horizons N. The light blue curve indicates $N = 12$, blue for $N = 10$, red for $N = 8$, yellow for $N = 6$, purple for $N = 4$, and green for $N = 2$.

system against disturbances. In what follows, we will demonstrate the optimality of the system with respect to the quadratic performance index $J = \sum e^2(k)$, $k \geq 0$. We will also measure the performance of our fuzzy feedforward compensator with respect to MPC only feedback control as in Fig. 21 and Table II.

Table II depicts the behavior of the error signal as a function of G_F. We also depict the relative magnitude of the u_F with respect to cross coupling signal γ as defined by $\alpha = \int u_F(t)dt / \int \gamma(t)dt$ while $\delta = \int e_{(F+\text{MPC})}(t)dt / \int e_{\text{MPC}}(t)dt$, where $e_{(F+\text{MPC})}(t)$ denotes the error signal when fuzzy feedforward compensator takes part in the control loop and $e_{\text{MPC}}(t)$ indicates the error signal of the MPC only autopilot in the feedback altitude loop.

B. Constrained MPC

This section deals with the optimality and robustness analysis of our constrained MPC, which can be regarded as a subset of the nonlinear control systems. We will first analyze the behavior of our nonlinear MPC by relaxing the imposed constraints so that we can analyze the closed loop control system using its *linear* approximation. First, the performance of the closed loop control systems under various prediction horizon is depicted in Fig. 22. As can be seen, our chosen estimation horizon $N = 10$ provides a delicate balance between achieving rapid settling time in the absence of overshoots.

While a small value of N could lead to poor time domain performance, a larger value of $N > 10$ may lead to undesirable overshoots. This way, we can optimize our optimization window N to achieve a responsive time domain response in the absence of overshoots. The performance of our nonlinear controllers against external disturbance (e.g., the unpredictable wind gusts and the noise in the measurements) represented in the form of white noise is given in Fig. 23. The system demonstrates sufficient robustness against any offset caused by external disturbances, in addition to process and measurement noise.

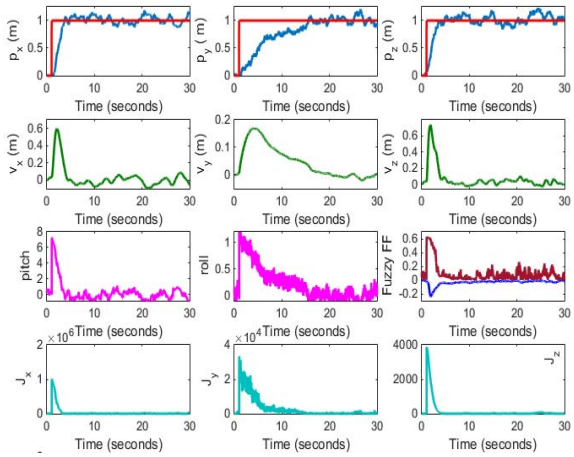

Fig. 23. System robustness against external disturbances and noise in the measurements with $G_F = 0.7$. (a) First row: $p_{x,y,z}$. (b) Second row: $v_{x,y,z}$. (c) Third row: pitch (left), roll (middle), fuzzy feedforward control signal versus cross coupling disturbance signal (right). (d) Fourth row: performance index.

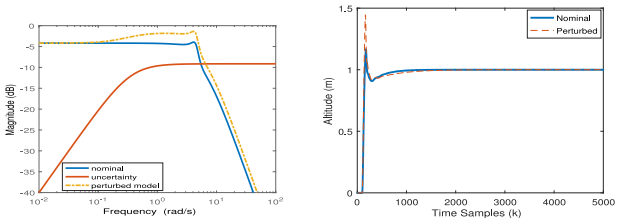

Fig. 24. Comparison between the performance of the altitude control system using nominal model of the plant with respect to its perturbed model. (a) Left: bode plot. (b) Right: time domain response.

Given stable closed loop poles for feedback MPC control $z_{1,2,\ldots,n}$, $|z_n| \leq 1$ $\forall n$, where n denotes the order of the system, as depicted in Section V; in addition to stable fuzzy feedforward compensator poles as in z_{F_m}, $|z_{F_m}| \leq 1$, $\forall m$ as in (11) and (12), it is apparent that the stability of the closed loop control system is guaranteed.

C. Robustness Analysis Using Multiplicative Uncertainty

Adopting the *multiplicative* uncertainty model, we define the uncertainty of our UAV (as in [22]). For our altitude control loop, we define its uncertainty to be as follows $W_a = [0.3491s/(s + 0.3491)]$. As seen in Fig. 24, while the Bode plot of the open loop system shows uncertainty of more than 25% (as shown in the peak of the Bode diagram), the response of our closed loop control system (nominal mode) does not differ significantly compared to the response of the same system using perturbed model. Despite an increase in overshoot, the settling time and the final value of the closed loop control system using the nominal model is nearly the same as the one from the perturbed model. This clearly shows the robustness of the controller in the face of uncertainty.

VII. Noise Rejection Performance

Lastly, we will also investigate the noise rejection capability of the closed loop control systems and compare it with the

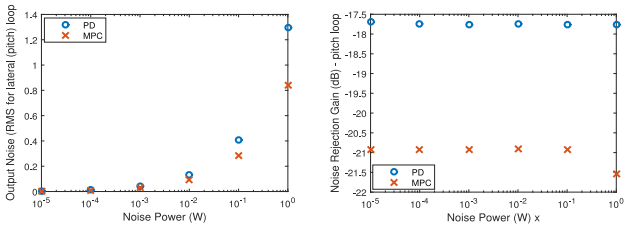

Fig. 25. Noise rejection performance of our MPC autopilot compared to a conventional PD controller ($K_p = 3.142$, $K_d = 3.013$, $K_F = 0.7113$) for the pitch loop. (a) Output RMS noise. (b) Noise rejection gain.

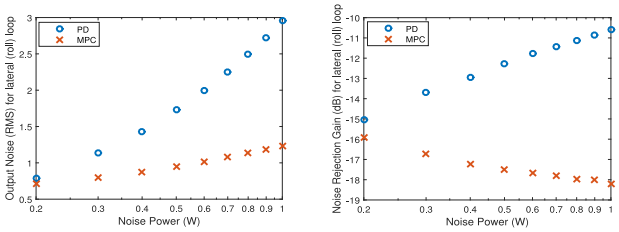

Fig. 26. Noise rejection performance of our MPC autopilot compared to the conventional PD controller ($K_p = 27.453$, $K_d = 86.638$, $K_F = 1151.2$) for the roll loop. (a) Output RMS noise. (b) Noise rejection gain.

performance of traditional proportional differential (PD) position control equipped with an additional low pass filter for the derivative term in order to limit the high frequency gain and noise. The mathematical equation of the controller is given as follows: $K_p + K_d[K_f s/(s + K_f s)]$ [41].

We compare the noise at the controlled variables (3-D position) with respect to the input disturbance noise in terms of its root mean square (RMS) value and noise rejection gain in decibels (dB) which can be mathematically defined as follows: $G_n = 20\log_{10}([N_p(\text{RMS})]/[N_i(\text{RMS})])$, where N_p denotes the RMS noise at the controlled variables, i.e., $p_{x,y,z}$ and N_i indicates the RMS of input noise. We investigate the performance of the system against the noise power in Watts.

The noise rejection performance of the closed loop MPC autopilot for the pitch loop lateral direction can be depicted in Fig. 25. Also Fig. 25 left denotes the RMS of the noise at the x-position, given the same amount of the input noise power, the right figure indicates the noise rejection gain in dB of both controllers. As can be seen, our MPC autopilot outperforms traditional PD position control of similar settling time with an average improvement gain around -3 dB.

Likewise, the noise rejection performance of our MPC autopilot for the roll loop with respect to traditional PD controller of similar settling time is depicted in Fig. 26. It is apparent that the noise rejection capability of the closed loop control system of our MPC autopilot varies as the power increases. Moreover, the noise rejection capability of our MPC autopilot for the altitude motion with respect to conventional PD controller is depicted in Fig. 27. Overall, our hybrid control system demonstrates superiority in terms of noise rejection capability with respect to conventional PD position control.

Lastly, the noise rejection performance of our fuzzy logic feedforward compensator for both the pitch and the roll channels is depicted in Fig. 28. It is apparent that our fuzzy logic

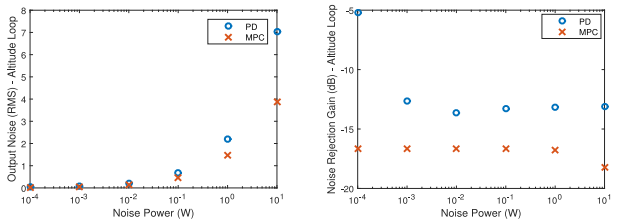

Fig. 27. Noise rejection performance of our MPC autopilot compared to the conventional PD controller ($K_p = 0.155$, $K_d = 0.01$, $K_F = 100$) for the altitude loop. (a) Output RMS noise. (b) Noise rejection gain.

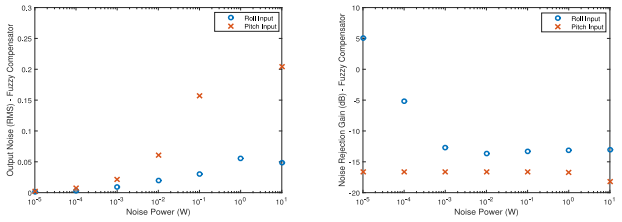

Fig. 28. Noise rejection performance of our fuzzy logic feedforward compensator. (a) RMS as a function of the noise power. (b) Gain as a function of the noise power.

feedforward autopilot has outperformed the performance of both the MPC and PD autopilots in terms of noise filtering. This capability is essential and serves to complement the performance of the MPC autopilot in feedback loop.

VIII. Conclusion

We have explored the performance of the hybrid feedback MPC/fuzzy feedforward autopilots. Through extensive computer simulations, we have shown the ability of the proposed control system to perform accurate trajectory tracking in the face of uncertainties.

We have also conducted the stability analysis in addition to the optimality analysis of the overall closed loop control systems to highlight the efficacy of the proposed control systems. In addition, we have also performed a comparative study to point out the relative merit of the proposed control algorithms with respect to conventional PD position control. It turns out that both feedback MPC and feedforward fuzzy compensator can be a good complement. Their performance successfully outperforms the traditional PD controller, which has been widely implemented as a standard position control in robotics.

We envisage that robust hybrid autopilot systems, specifically designed for small UAVs will become widely studied in the future. This is mainly due to the complementary nature of both control systems. Also, as the speed of the on-board computers have progressively advanced, one can implement robust MPC systems, supported by an intelligent control approach such as a fuzzy control system.

References

[1] E. A. George, G. Tiwari, R. N. Yadav, E. Peters, and S. Sadana, "UAV systems for parameter identification in agriculture," in *Proc. Glob. Humanitarian Technol. Conf. South Asia Satellite (GHTC-SAS)*, Thiruvananthapuram, India, 2013, pp. 270–273.

[2] F. Körner, R. Speck, A. H. Göktogan, and S. Sukkarieh, "Autonomous airborne wildlife tracking using radio signal strength," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Taipei, Taiwan, 2010, pp. 107–112.

[3] A. Pahsa, P. Kaya, G. Alat, and B. Baykal, "Integrating navigation & surveillance of unmanned air vehicles into the civilian national airspaces by using ADS-B applications," in *Proc. Integr. Commun. Navig. Surveilance Conf. (ICNS)*, Herndon, VA, USA, 2011, pp. J7-1–J7-7.

[4] F. Santoso, M. A. Garratt, and S. G. Anavatti, "Visual–inertial navigation systems for aerial robotics: Sensor fusion and technology," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 260–275, Jan. 2017.

[5] J. Blazakis, *Border Security and Unmanned Aerial Vehicles*. Washington, DC, USA: Congressional Res. Service, 2004, pp. CRS1–CRS6.

[6] D. W. Casbeer, R. W. Beard, T. W. McLain, S.-M. Li, and R. K. Mehra, "Forest fire monitoring with multiple small UAVs," in *Proc. Amer. Control Conf.*, vol. 5. Portland, OR, USA, 2005, pp. 3530–3535.

[7] J. Nikolic *et al.*, "A UAV system for inspection of industrial facilities," in *Proc. Aerosp. Conf.*, Mar. 2013, pp. 1–8.

[8] F. Santoso, M. A. Garratt, M. R. Pickering, and M. Asikuzzaman, "3D mapping for visualization of rigid structures: A review and comparative study," *IEEE Sensors J.*, vol. 16, no. 6, pp. 1484–1507, Mar. 2016.

[9] G. T. Martin, "Modelling and control of the Parrot AR.Drone," *J. Undergraduate Eng. Res.*, vol. 5, no. 1, pp. 1–12, 2012.

[10] F. Santoso, M. Liu, and G. Egan, "H_2 and H_∞ robust autopilot synthesis for longitudinal flight of a special unmanned aerial vehicle: A comparative study," *IET Control Theory Appl.*, vol. 2, no. 7, pp. 583–594, Jul. 2008.

[11] F. Santoso, M. Liu, and G. K. Egan, "Robust μ-synthesis loop shaping for altitude flight dynamics of a flying-wing airframe," *J. Intell. Robot. Syst.*, vol. 79, no. 2, pp. 259–273, 2015.

[12] B. Kouvaritakis and M. Cannon, *Model Predictive Control: Classical, Robust, and Stochastic*. Cham, Switzerland: Springer, 2016.

[13] R. Cui, C. Yang, Y. Li, and S. Sharma, "Adaptive neural network control of AUVs with control input nonlinearities using reinforcement learning," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 6, pp. 1019–1029, Jun. 2017.

[14] S. Wen, M. Z. Q. Chen, Z. Zeng, X. Yu, and T. Huang, "Fuzzy control for uncertain vehicle active suspension systems via dynamic sliding-mode approach," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 1, pp. 24–32, Jan. 2017.

[15] F. Santoso, M. A. Garratt, and S. G. Anavatti, "Fuzzy logic-based self-tuning autopilots for trajectory tracking of a low-cost quadcopter: A comparative study," in *Proc. Int. Conf. Adv. Mechatronics Intell. Manuf. Ind. Autom.*, 2015, pp. 64–69.

[16] F. Santoso, M. A. Garratt, and S. G. Anavatti, "Adaptive neuro-fuzzy inference system identification for the dynamics of the AR.Drone quadcopter," in *Proc. 4th Int. Conf. Sustain. Energy Eng. Appl.*, IEEE, Oct. 2016, pp. 55–60.

[17] F. Santoso, M. Garratt, and S. G. Anavatti, "Fuzzy system identification for the dynamics of the AR.Drone quadcopter," in *Proc. Aust. Conf. Robot. Autom.*, Brisbane, QLD, Australia, Dec. 2016.

[18] F. Santoso, M. A. Garratt, and S. G. Anavatti, "State-of-the-art intelligent flight control systems in unmanned aerial vehicles," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 2, pp. 613–627, Apr. 2018.

[19] A. Nemati and M. Kumar, "Modeling and control of a single axis tilting quadcopter," in *Proc. Amer. Control Conf.*, Portland, OR, USA, 2014, pp. 3077–3082.

[20] B. J. Emran, J. Dias, L. Seneviratne, and G. Cai, "Robust adaptive control design for quadcopter payload add and drop applications," in *Proc. Chin. Control Conf. (CCC)*, Hangzhou, China, 2015, pp. 3252–3257.

[21] A. Mystkowski, "An application of mu-synthesis to control of a small air vehicle–simulation results," *J. Vibroeng.*, vol. 14, no. 1, pp. 79–86, Mar. 2012.

[22] A. Mystkowski, "Robust control of the micro UAV dynamics with an autopilot," *J. Theor. Appl. Mech.*, vol. 51, no. 3, pp. 751–761, Jan. 2013.

[23] A. Mystkowski, "Implementation and investigation of a robust control algorithm for an unmanned micro-aerial vehicle," *Robot. Auton. Syst.*, vol. 62, no. 8, pp. 1187–1196, Aug. 2014.

[24] A. Mystkowski, "Robust optimal control of MAV based on linear-time varying decoupled model dynamics," in *Mechatronic Systems and Materials IV* (Solid State Phenomena), vol. 198. Dürnten, Switzerland: Trans Tech, 2013, pp. 571–576.

[25] M. Q. Huynh, W. Zhao, and L. Xie, "L1 adaptive control for quadcopter: Design and implementation," in *Proc. 13th Int. Conf. Control Autom. Robot. Vis. (ICARCV)*, Singapore, 2014, pp. 1496–1501.

[26] F. M. Drop, D. M. Pool, H. J. Damveld, M. M. van Paassen, and M. Mulder, "Identification of the feedforward component in manual control with predictable target signals," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1936–1949, Dec. 2013.
[27] M. Fatan, B. L. Sefidgari, and A. V. Barenji, "An adaptive neuro PID for controlling the altitude of quadcopter robot," in *Proc. Methods Models Autom. Robot. (MMAR)*, Międzyzdroje, Poland, 2013, pp. 662–665.
[28] W. S. Maj and B. Butkiewicz, "Flying n-copter with fuzzy logic control," in *Proc. Signal Process. Symp. (SPS)*, 2013, pp. 1–6.
[29] A. R. Firdaus and M. O. Tokhi, "Robust sliding mode—Based interval type-2 fuzzy logic observer for quadcopter UAVs," in *Proc. 19th Int. Conf. Syst. Theory Control Comput. (ICSTCC)*, Cheile Gradistei, Romania, 2015, pp. 559–564.
[30] L. M. Argentim, W. C. Rezende, P. E. Santos, and R. A. Aguiar, "PID, LQR and LQR-PID on a quadcopter platform," in *Proc. Int. Conf. Informat. Electron. Vis. (ICIEV)*, Dhaka, Bangladesh, 2013, pp. 1–6.
[31] J. M. Seddon and S. Newman, *Basic Helicopter Aerodynamics*. Chichester, U.K.: Wiley, Jun. 2011.
[32] H. Glauert, "A general theory of the autogyro," Roy. Aeronaut. Soc., London, U.K., Rep. 111, Nov. 1926.
[33] R. W. Prouty, *Helicopter Performance, Stability and Control*. Malabar, FL, USA: Krieger, 1986.
[34] R. C. Nelson, *Flight Stability and Automatic Control*, vol. 2. Boston, MA, USA: McGraw-Hill, 1998.
[35] Y. Wang and D. J. Inman, "Experimental validation for a multifunctional wing spar with sensing, harvesting, and gust alleviation capabilities," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 4, pp. 1289–1299, Aug. 2013.
[36] I. R. Petersen and A. V. Savkin, *Robust Kalman Filtering for Signal and Systems With Large Uncertainties*. New York, NY, USA: Springer, 1999.
[37] R. Fletcher, *Practical Methods of Optimization*. Chichester, U.K.: Wiley, 1987.
[38] C. V. Rao, S. J. Wright, and J. B. Rawlings, "Application of interior-point methods to model predictive control," *J. Optim. Theory Appl.*, vol. 99, no. 3, pp. 723–757, 1998.
[39] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.
[40] M. Liu, G. K. Egan, and F. Santoso, "Modeling, autopilot design, and field tuning of a UAV with minimum control surfaces," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 6, pp. 2353–2360, Nov. 2015.
[41] K. H. Ang, G. C. Y. Chong, and Y. Li, "PID control system analysis, design, and technology," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 4, pp. 559–576, Jul. 2005.

Fendy Santoso (S'10–M'15) received the Ph.D. degree in electrical engineering from the University of New South Wales (UNSW), Sydney, NSW, Australia, in 2012, under a University International Postgraduate Award Scholarship.

Since 2015, he has been a Post-Doctoral Researcher with the School of Engineering and Information Technology, UNSW Canberra, Canberra, ACT, Australia. He has been a Chief Investigator of several Defence Science and Technology Group-funded research projects. His current research interests include system identification, control systems, artificial intelligence, and their applications in aerial robotics.

Dr. Santoso has been a reviewer for many high impact control and robotics journals, such as the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, and IEEE ROBOTICS AND AUTOMATION LETTERS. He is an Associate Editor of the *KES Journal: International Journal of Knowledge-Based and Intelligent Engineering Systems*.

Matthew A. Garratt (M'16) received the B.E. degree in aeronautical engineering from Sydney University, Sydney, NSW, Australia, the Graduate Diploma degree in applied computer science from Central Queensland University, Rockhampton, QLD, Australia, and the Ph.D. degree in biologically inspired robotics from Australian National University, Canberra, ACT, Australia, in 2008.

He is an Associate Professor with the School of Engineering and Information Technology (SEIT), University of New South Wales Canberra, Canberra, where he is currently the Deputy Head of School (Research) in SEIT and the Chair of the CIS task force on the Ethics and Social Implications of CI. His current research interests include sensing, guidance, and control for autonomous systems with particular emphasis on biologically inspired and computational intelligence approaches.

Sreenatha G. Anavatti received the Ph.D. degree from the Indian Institute of Science, Bengaluru, India

He was an Associate Professor with the Indian Institute of Technology Bombay, Mumbai, India. He moved to Australia in 1998. He is a Senior Lecturer with the School of Engineering and Information Technology, University of New South Wales Canberra, Canberra, ACT, Australia. He has supervised a number of Ph.D. scholars over these years. His current research interests include application of artificial intelligence techniques like fuzzy and neural systems for UAVs, underwater vehicles, ground vehicles, identification and control of dynamic systems, navigation and path planning for autonomous vehicles, evolutionary fuzzy systems and their applications to dynamic systems.

Ian Petersen (S'80–M'83–SM'96–F'00) was born in Victoria, Korumburra VIC, Australia. He received the Ph.D. degree in electrical engineering from the University of Rochester, Rochester, NY, USA, in 1984.

From 1983 to 1985, he was a Post-Doctoral Fellow with Australian National University, Canberra, ACT, Australia. In 1985, he joined the University of New South Wales, Australian Defence Force Academy, Canberra, where he is currently a Scientia Professor and an Australian Research Council Laureate Fellow with the School of Engineering and Information Technology. His current research interests include robust control theory, quantum control theory, and stochastic control theory.

Dr. Petersen has served as an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, SYSTEMS AND CONTROL LETTERS, *Automatica*, and *SIAM Journal on Control and Optimization*. He is currently an Editor for *Automatica*. He is a fellow of the Australian Academy of Sciences.