Coping with Quadcopter Payload Variation via Adaptive Robust Control

Ahmad Kourani
Dept. of Mechanical Engineering
American University of Beirut
Beirut, Lebanon
ahk42@mail.aub.edu

Kareem Kassem
Dept. of Biomedical Engineering
Boston University
Boston, USA
knk07@bu.edu

Naseem Daher
Dept. of Electrical and Computer Engineering
American University of Beirut
Beirut, Lebanon
nd38@aub.edu.lb

***Abstract*—In this work, Adaptive Robust Control (ARC) is applied to control the motion of a quadrotor with a varying payload. ARC, based on the backstepping method, guaranties asymptotic output tracking and satisfactory transient response of the quadrotor motion in the face of uncertainties and disturbances. The control system architecture includes an outer loop where a PID controller is used to generate the position inputs, which are fed to the ARC-based attitude control system at the inner loop level. The controller design is validated against payload variation in both numerical simulation and experimentally on a quadcopter. The obtained results demonstrate the accuracy of the ARC design with a constant payload against a well-tuned fixed-gain PID controller, and the adaptation and robustness of ARC become evident when the payload is varied and tracking performance is maintained.**

***Index Terms*—Adaptive Robust Control, quadrotor, payload change, backstepping.**

I. Introduction

Considerable attention has been paid to the field of unmanned aerial vehicles (UAVs) during the past few years, especially to quadrotors. Due to their small size, superior maneuverability, and ability to hold small loads, quadcopters are widely used for applications such as photography, search missions, irrigation, broadcasting, and others. Originally, the classical proportional-integral-derivative (PID) controller was used to achieve reference trajectory tracking [1], [2]. Such controller designs provide satisfactory results in terms of trajectory tracking, but they have little potential to reject unpredicted disturbances and uncertain nonlinearities, when not accounted for in advance. This being said, advanced control strategies must be applied to guarantee robustness of quadrotors in the face of disturbances, parametric uncertainties, and unmodeled nonlinearities such as wind gusts, sudden loss of thrust in one or more propellers, varying mass and inertia tensor, amongst others.

Other controller designs have been applied to quadrotors. Among nonlinear techniques, Bouabdallah and Siegwart [3] designed a Lyapunov-based backstepping method for controlling a quadrotor. Similar backstepping based methods were developed by Madani and Benallegue [4] and by Tripathi et al [5], however experimental validation results were not included. As for other nonlinear control systems, a comparison between adaptive sliding mode control and feedback linearization was demonstrated in [6]. Dydek et al. [7] introduced an adaptive controller for quadrotor UAVs to mitigate the effects of a loss-of-thrust event. Also, an adaptive control technique was applied to compensate for dynamic changes in the center of gravity of the quadrotor [8]. A robust attitude controller was designed by Rudin et al. [9]. Combining the two latter mentioned controllers, Nicol et al. [10] proposed a robust adaptive control for quadrotors to accommodate for unknown payload and disturbances. In addition, a robust adaptive controller for quadrotors with payload add-drop applications was designed in [11].

Nonlinear control design methods can be employed to tackle the nonlinearity characteristic of quadrotors given their coupled nonlinear dynamics. Nonlinearities can be divided into two main categories: uncertain nonlinearities and parametric uncertainties [12]–[15]. Examples of parametric uncertainties include the, sometimes drastic and sudden, changes in the load carried by the quadrotor for instance. On the other hand, uncertain nonlinearities are illustrated by disturbances and aerodynamics drag forces that cannot be modelled exactly or whose functions are unknown [12], [14], [15]. Two main nonlinear control methods have been developed to account for these types of uncertainties: adaptive control (AC), and deterministic robust control (DRC).

Yao and Tomizuka proposed an adaptive robust control (ARC) approach for high performance robust control of nonlinear systems in the presence of both parametric uncertainties and uncertain nonlinearities. A general framework was formalized by Yao [14] in an attempt to present a rigorous theoretical method for the control of a one degree-of-freedom (1-DOF) system, which was developed and applied to many systems, such as hydraulic cylinder [16], DC motor [17], and active suspension system [18]. Min2011 et al. [19] proposed an altitude controller for a quadrotor based on ARC with unknown mass but without experimental validation.

This paper presents a framework to design an adaptive robust controller for quadrotors using the backstepping approach with Lyapunov function [1], [12], [13]. Section II presents the dynamical model of a quadrotor along with the linearization process. Section III includes the design process of ARC with simulation results. Section IV shows the implementation procedure and the experimental results from comparing ARC

978-1-5386-4500-0/18/$31.00 ©2018 IEEE

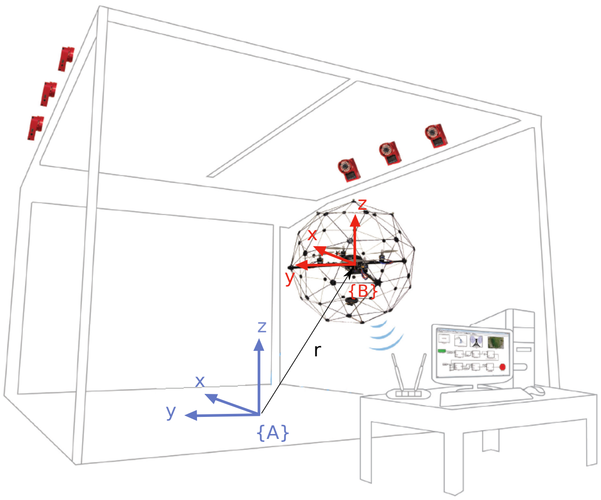

Fig. 1. QBall2 quadcopter with body coordinate frame (red) and inertial frame (blue) [20]

to a well-tuned PID controller. Section V concludes the paper with an outlook for future work.

II. Model

In this section, a high fidelity dynamic model of a quadrotor is developed after establishing appropriate frames of reference, studying the rigid body dynamics, and linearizing the model about the hovering state as an equilibrium point.

A. Coordinate Frames

It is essential to start the modelling process with proper specification of the frames of reference. As shown in Fig. 1, let the inertial frame be denoted by $\{\mathbf{A}\}$ and the body frame by $\{\mathbf{B}\}$. The quadrotor has six degrees of freedom (DOF) denoted by $\{x, y, z, \phi, \theta, \psi\}$, where the position of the quadcopters center of mass is designated by the vector $\mathbf{r} = [x \ y \ z]^T$, and ϕ, θ and ψ represent the roll, pitch, and yaw angles, respectively. The state vector, whose choice is explained in Section 3, is defined as follows:

$$X^T = [z, \dot{z}, y, \dot{y}, \phi, \dot{\phi}, x, \dot{x}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T \tag{1}$$

In this work, the $Z - X - Y$ Euler angles are used for the rotation matrix. Let R be defined as the rotation from $\{\mathbf{B}\}$ to $\{\mathbf{A}\}$ coordinates as follows:

$$R = \begin{bmatrix} c_\theta c_\psi - s_\phi s_\theta s_\psi & -c_\phi s_\psi & s_\theta c_\psi + s_\phi c_\theta s_\psi \\ c_\theta s_\psi + s_\phi s_\theta c_\psi & c_\phi c_\psi & s_\theta s_\psi - s_\phi c_\theta c_\psi \\ -c_\phi s_\theta & s_\phi & c_\phi c_\theta \end{bmatrix} \tag{2}$$

where the letters c and s represent the cosine and sine functions, respectively.

B. Rigid Body Dynamics

The translational motion of the quadrotor is written as follows:

$$m\ddot{\mathbf{r}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ F \end{bmatrix} \tag{3}$$

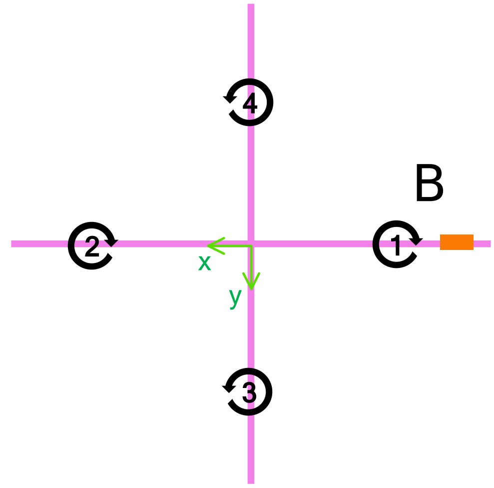

Fig. 2. Rotors Notation

where m is the quadrotors mass, g is the gravitational acceleration, and F_i is the non-conservative force of thrust induced by the i^{th} rotor. Similarly, the rotational motion of the quadrotor is modelled by:

$$I\frac{d\vec{\omega}}{dt} + \vec{\omega} \times I\vec{\omega} = \vec{\tau} \tag{4}$$

where I is the quadrotors inertia tensor, $\vec{\omega}$ is the angular velocity vector of $\{\mathbf{B}\}$ with respect to $\{\mathbf{A}\}$ expressed in $\{\mathbf{B}\}$, and $\vec{\tau}$ is the vector of non-conservative moments applied to the quadrotors airframe by the aerodynamics of the rotors [1]. The platform is assumed to be symmetric with equal distribution of masses, which reduces the inertia tensor to the following diagonal matrix:

$$R = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}. \tag{5}$$

C. Linearization

To proceed in the design of the controller using linear control theory, the equations described in the previous section are linearized about the hovering condition, expressed as follows:

$$\begin{aligned} \phi = \theta = \psi = 0 \\ \dot{\phi} = \dot{\theta} = \dot{\psi} = 0 \end{aligned} \tag{6}$$

The obtained linearized translational equations of motion are:

$$\begin{aligned} \ddot{x} &= g\theta \\ \ddot{y} &= -g\phi \\ \ddot{z} &= -g + \frac{1}{m}u_1 \end{aligned} \tag{7}$$

where u_1 is the sum of thrusts of the rotors, considered as the first of four control signals of the closed loop system.

The rotational linearized equations of motion are given by:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} u_2/I_{xx} \\ u_3/I_{yy} \\ u_4/I_{zz} \end{bmatrix}, \tag{8}$$

where $\dot{p}$, $\dot{q}$, $\dot{r}$, are the angular velocity components of the quadrotor in its body frame, and u_2, u_3, and u_4 are control signals that are components of $\vec{\tau}$ expressed in eq.(4). The angular velocity components are related to the derivatives of the roll, pitch, and yaw angles by the following rotation matrix:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} c_\theta & 0 & -c_\phi s_\theta \\ 0 & 1 & s_\phi \\ s_\theta & 0 & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{9}$$

which is approximately equal to the identity matrix for near hovering conditions.

D. Forces and Moments

The last step in the derivation of the quadrotor model is studying the aerodynamic forces of the propellers that are installed at the four corners of its rigid body. Each rotating propeller is modelled as:

$$T_i = c_T \omega_i^2, \quad i \in [1, 2, 3, 4] \tag{10}$$

where c_T is a positive constant [1] and ω_i is the angular velocity. T_i is toward the z-direction in $\{\mathbf{B}\}$. Similarly, the drag force, which is the force acting in a direction opposite to the rotation of the blades, is modelled by:

$$Q_i = c_Q \omega_i^2, \quad i \in [1, 2, 3, 4] \tag{11}$$

where c_Q is also a positive constant [1]. In practice, the motor dynamics are fast and can be neglected in modeling.

The total Force and moments acting on the quadrotor in linked to the angular velocities of the rotors by the following relation:

$$\begin{bmatrix} T_\Sigma \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} c_T & c_T & c_T & c_T \\ 0 & 0 & dc_T & -c_T \\ dc_T & -dc_T & 0 & 0 \\ c_Q & c_Q & -c_Q & -c_Q \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{12}$$

where d is the distance from the center of mass to each rotor. The PWM (Pulse Width Modulation) input to each motor is proportional to the angular velocity of its rotor.

III. Controller Design

A. Introduction to Adaptive Robust Control (ARC) Theory

The design of an adaptive robust controller, which was introduced in [14], consists of a combination of two control techniques, deterministic robust control (DRC) and adaptive control (AC). While DRC tends to suffer from chattering and a relatively large steady-state error and AC exhibits unknown transient response, ARC overcomes the drawbacks of both design methods by guaranteeing asymptotic output tracking and reducing model uncertainties through on-line parameter estimation. However, to proceed in the described manner, the conflict between both design techniques must be solved. Whilst DRC allows only bounded uncertainties, the parameter estimates provided by AC can be unbounded in the presence of nonlinear functions. This conflict can be solved by modifying the conventional adaptation law to a continuous smooth projection or discontinuous projection [21].

B. Problem Formulation

The aim of the controller is to have the quadrotor track desired reference trajectories given in terms of position and yaw angle. With $X^T = [z, \dot{z}, y, \dot{y}, \phi, \dot{\phi}, x, \dot{x}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$ as the state vector, the system is described by the following state-space representation:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -g + \theta_1 u_1 \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= -g x_5 \\ \dot{x}_5 &= x_6 \\ \dot{x}_6 &= \theta_2 u_2 \\ \dot{x}_7 &= x_8 \\ \dot{x}_8 &= g x_9 \\ \dot{x}_9 &= x_{10} \\ \dot{x}_{10} &= \theta_3 u_3 \\ \dot{x}_{11} &= x_{12} \\ \dot{x}_{12} &= \theta_4 u_4, \end{aligned} \tag{13}$$

where the vector of unknown parameters being $\Theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T = [\frac{c_T}{m} \ \frac{c_T d}{I_x} \ \frac{c_T d}{I_y} \ \frac{c_Q}{I_z}]^T$. Notice that the system can be divided into four subsystems. Each subsystem is treated as a Single-Input Single-Output (SISO) system represented in semi-strict feedback form, which the backstepping design is applied on:

- x_{1-2}: vertical acceleration (up/down)
- x_{3-6}: roll about the y-axis
- x_{7-10}: pitch about the x-axis
- x_{11-12}: yaw about the z-axis

The second and third subsystems (roll and pitch) have a combined control law: a proportional-integrator-derivative (PID) controller is used for the first two states of each subsystem, namely the position and velocity, and the rest is handled by the backstepping method.

The ARC control laws of the four subsystems are derived in a similar fashion, but not all are elaborated in this paper for conciseness purposes. Since the first subsystem (x_1, x_2) is of interest, its control law is systematically derived in detailed steps. The first subsystem is given by:

$$\begin{aligned} x_1 &= x_2 \\ x_2 &= -g + \theta_1 u_1 \end{aligned} \tag{14}$$

where the states are $[x_1 \ x_2]^T = [z \ \dot{z}]^T$.

1) Controller Design - Step 1: Since the first equation in (14) does not include uncertainties, an ARC Lyapunov function can be directly synthesized for the first two equations [12], [15]. Define a switching-function-like quantity as:

$$e_2 = x_2 - \alpha_2, \tag{15}$$

where $\alpha_2 = \dot{x}_{1d} - k_1 e_1$, $e_1 = x_1 - x_{1d}$, and x_{1d} is the desired trajectory to be tracked by x_1. Since the transfer functions $G(s) = e_1(s)/e_2(s)$ is stable, reducing e_1 produces similar effects to reducing e_2. Therefore, the remainder of the design process intends to make e_4 converge to zero.

2) Controller Design - Step 2: Differentiating eq.(15) yields the following error dynamics:

$$e_2 = x_2 - \alpha_2 = -g + \theta_1 u_1 \tag{16}$$

where u_1 is the actual control input. Defining a Lyapunov function and differentiating it yields:

$$\begin{aligned} V_2 &= \frac{1}{2}e_2^2 \\ \dot{V}_2 &= e_2\dot{e}_2 = e_2(-g + \theta_1 u_1 - \dot{\alpha}_2). \end{aligned} \tag{17}$$

The presence of parametric uncertainties can be seen in eq.(17). The control law is denoted by:

$$u_1 = u_{1a} + u_{1s1} + u_{1s2}, \tag{18}$$

where u_{1a}, u_{1s_1}, and u_{1s_2} are the model compensation, stabilizing feedback, and robust feedback terms, respectively. The control law for the actual control input is hence designed as follows:

$$\begin{aligned} u_{1a} &= \frac{1}{\theta_1}(g + \dot{\alpha}_2) \\ u_{1s1} &= -k_1 e_1 \end{aligned} \tag{19}$$

where $\hat{\theta}_1$ is the estimate of θ_1 and k_1 is a positive constant. The robust feedback u_{1s2} is chosen to satisfy the following conditions:

$$\begin{aligned} &condition1: e_1(\theta_1 u_{2s_2} - \tilde{\theta}^T \varphi_{u_1} + \tilde{d}) \leq \epsilon_1 \\ &condition2: e_1\theta_1 u_{1s2} \leq 0 \end{aligned} \tag{20}$$

where $\tilde{\theta} = \hat{\theta} - \theta$ is the estimate error, φ_{u_1} is the regressor, $\tilde{d}$ is a parameter that accounts for unknown uncertainties, and ϵ_1 is a design parameter. Condition 1 guarantees that u_{1s2} will dominate the control signal and will reduce parametric and unknown uncertainties. Condition 2 implies that the robust function is dissipative in nature so that it does not interfere with the adaptation law [12]. A possible bounding function for u_{1s2} [13] is:

$$u_{1s2} = -\frac{\|\varphi_1\|^2 \|\theta_{1,max} - \theta_{1,min}\|^2}{4\epsilon_1} e_2, \tag{21}$$

where φ_1 is a regressor defined from eq.(16) as u_{1a}. The final step is to determine the adaptation law, which is given by:

$$\dot{\hat{\theta}}_1 = Proj(\Gamma u_{1a} e_2), \tag{22}$$

where Γ is the adaptation rate.

3) Control laws of the roll, pitch and yaw subsystems: The control laws of the other subsystems are synthesized in a similar fashion by following the previous derivation process. The control laws are then:

$$\begin{aligned} u_2 &= \frac{1}{\theta_2}(\dot{\alpha}_6 - (k_6 + \frac{h_2^2}{4\epsilon_2})e_6) \\ u_3 &= \frac{1}{\theta_3}(\dot{\alpha}_{10} - (k_{10} + \frac{h_3^2}{4\epsilon_3})e_{10}) \\ u_4 &= \frac{1}{\theta_4}(\dot{\alpha}_{12} - (k_{12} + \frac{h_4^2}{4\epsilon_4})e_{12}) \end{aligned} \tag{23}$$

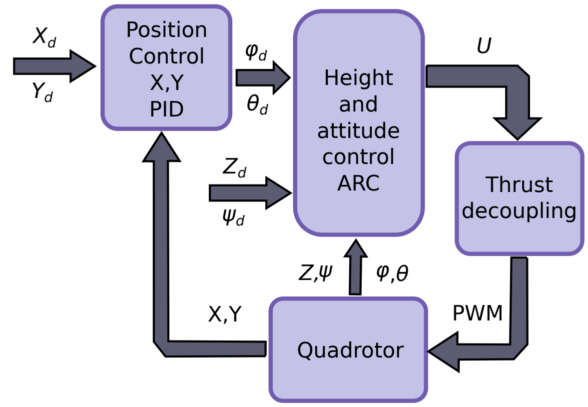

Fig. 3. Controller Structure

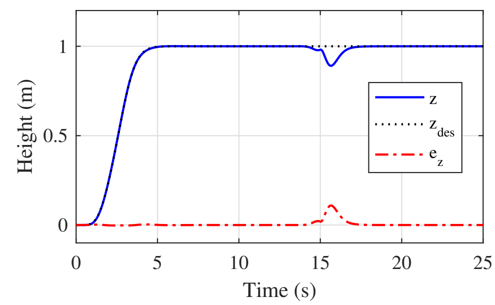

Fig. 4. Motion in the z-direction with ARC controller

where e_i, α_i, and h_i are calculated as:

$$\begin{aligned} e_i &= x_i - \alpha_i \\ \alpha_i &= \dot{x}_{i-1,d} - k_{i-1}e_{i-1} \\ h_i &= \|\varphi_i\|^2 \|\theta_{i,max} - \theta_{i,min}\|^2 \end{aligned} \tag{24}$$

Note that $x_{5,d}$ and $x_{9,d}$ are the desired roll and pitch angles respectively, and are calculated from the position PID control law as:

$$\begin{aligned} x_{5,d} &= -(k_p e_y + k_i \int e_y dt + k_d \dot{e}_y) \\ x_{9,d} &= k_p e_x + k_i \int e_x dt + k_d \dot{e}_x \end{aligned} \tag{25}$$

where $e_x = x - x_d$, $e_y = y - y_d$, and k_p, k_i and k_d are positive PID gains. A diagram that summarizes the controller structure is shown in Fig. 3

C. Simulation Results

The proposed controller is first tested in simulation to validate the design process. Fig. 4 shows the response of the quadrotor to a vertical hovering command of $1m$ as a step signal passed through an eighth order filter to make the signal well conditioned i.e., differentiable enough. A mass of 0.4 kg is suddenly added at t = 15s, the quadrotor quickly recovers ($< 2s$) and returns to the desired height. Fig. 5 shows the adaptation parameter θ_1, which converges to its true new value

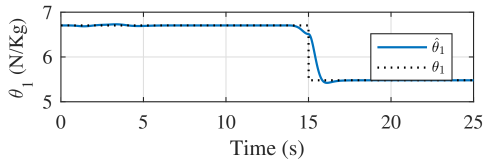

Fig. 5. Estimated Parameter θ_1

TABLE I
ARC PARAMETERS

k_1	2	k_2	2	ϵ_1	12	Γ_1	2
$k_{5,9}$	3.5	$k_{6,10}$	30	$\epsilon_{1,2}$	2	$\Gamma_{2,3}$	500
k_{11}	5	k_{12}	5	ϵ_5	0.05	Γ_4	20

in 1s after the mass is added. The controller gains for the vertical acceleration subsystem are presented in Table I, and the parameter bounds are: $\theta_{1,Max} = \frac{1.2c_T}{0.8m}$, $\theta_{1,Min} = \frac{0.8c_T}{1.2m}$, with the initial estimate set as $\hat{\theta}_{1,0} = \frac{c_T}{m}$.

IV. Experimental results

The QBall2 quadrotor [20] shown in Fig. 1 is used to test the designed controller. The quadrotor includes a 3-axis accelerometer, 3-axis gyroscope, and ARM Cortex-A9 1 GHz processor. The position feedback is generated by an OptiTrack motion capture system, and the velocities are derived from the position signal via differentiation. The angular rates are generated from the gyroscope only, and passed through a first order filter with a cutoff frequency of 60Hz to reduce the signal noise. A Kalman fitter is used to generate the angular positions. The quadrotor mass is 1.73Kg, with a diameter of 0.7m. The moments of inertia matrix is considered diagonal and its elements are: $I_{xx} = I_{yy} = 0.3kg.m^2$, and $I_{zz} = 0.4kg.m^2$ due to symmetry. The motor and propeller thrust constant is $c_T = 12N$, and the torque constant is $c_Q = 0.4N.m$, and the arm length is $d = 0.2m$.

The steps of the experiment are listed below:

1) A vertical upward motion of $1m$, during which the quadrotor lifts an object of 0.4 kg weight.
2) A horizontal motion of $1m$ in the positive $x-$direction.
3) A return to the initial horizontal position while dropping the object midway.
4) Returning to the original height.

The trajectory is generated as step functions for the z-motion and a bounded ramp functions for the x-motion with a slope of $0.2m/s$, all passed through an eighth order filter to ensure that the signals are differentiable. The lifted mass is connected to the quadrotor by a thin wire at a set distance to the quadrotor's cage, and due to this moment arm it also applies a twisting moment to the quadrotor. This experiment is tested with both controllers: ARC and PID. It is noted that PID serves as the baseline controller, which is carefully designed and well tuned by the vendor, QUANSER [20], to achieve excellent performance.

The results of the experiment are shown in Fig. 6 and Fig. 8 with both controllers overlaid on the same graph. It is noticed that at the object lift-off in the midst of the vertical flight, a very short delay is incurred when following the vertical trajectory, and a small drift in the horizontal plane occurs due to the moment about the center of gravity of the quadrotor created by the lifted load. At the object drop-off, the maximum upward drift for the ARC controller is about 18cm, while it is 25cm for the PID controller. For the motion in the vertical direction, the ARC controller showed relatively better performance compared to the PID controller in terms of maximum drift (less than $10cm$) and recovery time. From the motion in the horizontal direction, it is shown that the proposed controller is stable and able to perform the desired motion with less overshoot and smaller steady state error.

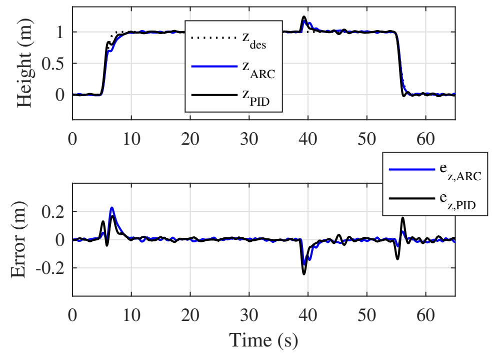

Fig. 6. Motion in the z-direction with ARC and PID controllers

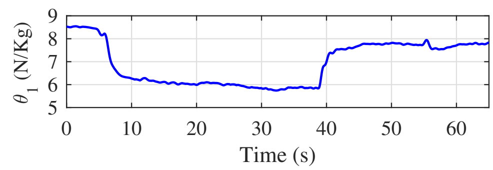

Fig. 7. Adaptation parameter θ_1

Fig. 7 shows the adaptation parameter θ_1 changing during the flight. Adaptation occurs quickly and in a way that the error in the vertical position converges to zero, and its value predicts the combined effect of the changing thrust and the varying mass.

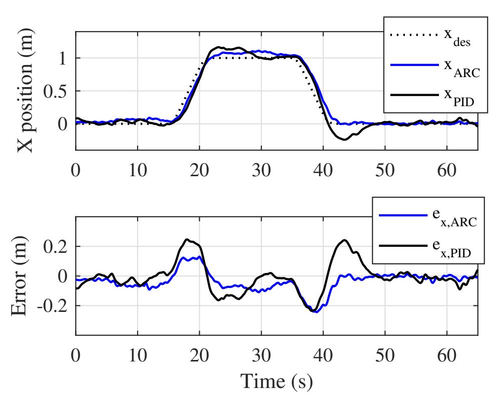

Fig. 8. Motion in the x-direction with ARC and PID controllers

V. Conclusion

In this paper, adaptive robust control (ARC) is applied to a quadrotor system. The backstepping method is used to design a controller for each of the four subsystems: vertical acceleration, roll, pitch, and yaw. A PID controller is used for the lateral position control to generate the desired roll and pitch angles. The proposed controller (ARC) showed better performance when compared to a vendor-designed PID controller in terms of stability and adaptation to mass fluctuation. Future work includes testing the performance of the controller against wind gusts and other disturbances, while implementing ARC in the outer loop of the control system architecture.

Acknowledgement

The authors would like to acknowledge the University Research Board (URB) at the American University of Beirut (AUB) for funding this work.

References

[1] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicules, modeling estimation, and control of quadrotor," *IEEE Robotics Automation Magazin*, pp. 20–32, September 2012.

[2] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav test bed," *IEEE Robotics Automation Magazine*, 2010.

[3] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *International Conference on Robotics and Automation*, pp. 2247–2252, 2005.

[4] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *International Conference on Intelligent Robots and Systems*, 2006.

[5] V. K. Tripathi and L. B. N. Verma, "Design of sliding mode and backstepping controllers for a quadcopter," in *39th National Systems Conference (NSC)*, 2015.

[6] D. Lee, H. J. Kim, and S. Sastry, "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter," *International Journal of Control, Automation, and Systems*, vol. 7, no. 3, pp. 419–428, 2009.

[7] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor uavs: A design trade study with flight evaluations," *IEEE Transactions on Control Systems Technology*, vol. 21, pp. 1400–1406, July 2013.

[8] I. Palunko and R. Fierro, "Adaptive control of a quadrotor with dynamic changes in the center of gravity," in *Proceedings of the 18th World Congress The International Federation of Automatic Control*, 2011.

[9] K. Rudin, M.-D. Hua, G. Ducard, and S. Bouabdallah, "A robust attitude controller and its application to quadrotor helicopters," in *Proceedings of the 18th World Congress The International Federation of Automatic Control*, 2011.

[10] C. Nicol, C. Macnab, and A. Ramirez-Serrano, "Robust adaptive control of a quadrotor helicopter," *Mechatronics*, vol. 21, pp. 927–938, 2011.

[11] B. J. Emran, J. Dias, L. Seneviratne, and G. Cai, "Robust adaptive control design for quadcopter payload add and drop applications," in *Proceedings of the 34th Chinese Control Conference*, 2015.

[12] E. Busquets and M. Ivantysynova, "Discontinuous projection-based adaptive robust control for displacement-controlled actuators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 33, no. 5, pp. 893–900, 2015.

[13] P. fei Qian, G. liang Tao, D. yuan Meng, and H. Liu, "A modified direct adaptive robust motion trajectory tracking controller of a pneumatic system," *Journal of Zhejiang University-SCIENCE C (Computers & Electronics)*, vol. 15, no. 10, pp. 878–891, 2015.

[14] B. Yao and M. Tomizuka, "Adaptive robust control of siso nonlinear systems in a semi-strict feedback form," *Automatica*, vol. 33, no. 5, pp. 893–900, 1997.

[15] B. Yao, F. Bu, and G. T. C. Chiu, "Nonlinear adaptive robust control of electro-hydraulic servo systems with discontinuous projections," in *37th IEEE Conference on Decision & Control*, (Tampa, Florida USA), pp. 2265–2270, 1998.

[16] B. Yao, "Adaptive robust motion control of single-rod hydraulic actuators: Theory and experiments," *IEEE/ASME Transactions on Mechatronics*, vol. 5, pp. 79–91, March 2000.

[17] J. Yao, Z. Jiao, and D. Ma, "Adaptive robust control of dc motors with extended state observer," *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 3630–3637, July 2014.

[18] W. Sun, Z. Zhao, and H. Gao, "Saturated adaptive robust control for active suspension systems," *IEEE Transactions on Industrial Electronics*, 2013.

[19] B.-C. Min, J.-H. Hong, and E. T. Matson, "Adaptive robust control (arc) for an altitude control of a quadrotor type uav carrying an unknown payloads," in *11th International Conference on Control, Automation and Systems*, 2011.

[20] Quanser, "Quanser innovate-educate, www.quanser.com," 2018.

[21] B. Yao, "High performance adaptive robust control of nonlinear systems: A general framework and new schemes," in *Proceedings of the 36th Conference on Decision & Control*, vol. 36, 1997.