Adaptive augmented torque control of a quadcopter with an aerial manipulator

Vidya Sumathy and Debasish Ghose

Abstract: A quadcopter manipulator system is an aerial robot consisting of a quadcopter with a robotic arm attached to it. The system has coupled nonlinear dynamics with uncertain time-varying parameters. The work in this paper focuses on designing an adaptive nonlinear controller to facilitate the uncertain system's trajectory tracking and stability. The novelty of the proposed work is the design and implementation of an adaptive feedback linearization controller, called adaptive augmented torque (AAT) control, for the aerial robot. The control law is based on a feedback linearization controller with model reference adaptive controller and a tracking error-based augmented term. Using the input-to-state stability concept, a bound on the parameter estimation error is also developed. In the presented methodology, the controller uses estimated values of system parameters obtained from the adaptive mechanism and the tracking error to compute the control input using the AAT control law. An adaptive law for estimating unknown parameters is obtained using the strictly positive real-Lyapunov method. The asymptotic stability of the closed-loop system is analyzed via the Lyapunov theory. Simulations implemented in MATLAB and ROS/Gazebo and preliminary hardware experiments are presented to validate the theoretical results and to corroborate the performance of the AAT control law.

Key words: quadcopter manipulator, feedback linearization controller, model reference adaptive control, lyapunov stability, parameter estimation error.

Résumé : Un système de manipulation de quadricoptère est un robot aérien composé d'un quadricoptère muni d'un bras robotique. Le système a jumelé la dynamique non linéaire avec des paramètres incertains variables dans le temps. Le présent document porte sur la conception d'un contrôleur non linéaire adaptatif pour faciliter le suivi de la trajectoire et la stabilité du système incertain. L'originalité du projet proposé est la conception et la mise en œuvre d'un contrôleur adaptatif de linéarisation par rétroaction, appelé commande adaptative de couple augmenté (« adaptive augmented torque, AAT »), pour le robot aérien. La loi de commande est basée sur un contrôleur de linéarisation par rétroaction avec un contrôleur adaptatif de référence de modèle et un terme augmenté basé sur les erreurs de suivi. En utilisant le concept de stabilité d'entrée à état (« input-to-state stability, ISS »), une limite sur l'erreur d'estimation des paramètres est également développée. Dans la méthodologie présentée, le contrôleur utilise les valeurs estimées des paramètres du système obtenues à partir du mécanisme adaptatif et de l'erreur de suivi pour calculer l'entrée de commande à l'aide de la loi de commande AAT. Une loi adaptative pour estimer des paramètres inconnus est obtenue en utilisant la méthode Lyapunov réelle strictement positive. La stabilité asymptotique du système en boucle fermée est analysée par le truchement de la théorie de Lyapunov. Des simulations mises en œuvre dans MATLAB et ROS/Gazebo et des expériences matérielles préliminaires sont présentées pour valider les

Received 17 June 2021. Accepted 4 November 2021.

V. Sumathy and D. Ghose. Guidance, Control, Decision and Systems Laboratory (GCDSL) and Mobile Robotics Laboratory (MRL), Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India.

Corresponding author: Vidya Sumathy (e-mail: vidyasumathy@iisc.ac.in).

© 2021 The Author(s). This work is licensed under a [Creative Commons Attribution 4.0 International License \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

résultats théoriques et corroborer les performances de la loi de commande AAT. [Traduit par la Rédaction]

Mots-clés : manipulateur quadricoptère, contrôleur de linéarisation par rétroaction, commande adaptative de référence de modèle, stabilité de Lyapunov, erreur d'estimation des paramètres.

1. Introduction

Aerial robotics is one of the most cutting-edge topics of interest in both industry and academic research. The integration of autonomous aerial vehicles (UAVs) with robotics has led to the development of robots with multifaceted applications like load transportation (Villa et al. 2020), canopy sampling (Kutia et al. 2018; Charron et al. 2020), valve turning (Orsag et al. 2014), bridge inspection (Jimenez-Cano et al. 2017), civil engineering applications (Freeman et al. 2021), drawer operation (Kim et al. 2015), and natural gas leak detection (Barchyn et al. 2017).

Quadcopters are some of the most widely used aerial vehicles both for research and in industry. This vehicle is under-actuated as it has four control inputs for all six degrees of freedom (DoF) and exhibits nonlinear dynamics. A comprehensive assessment of recent trends in UAV or drones research and applications can be found in Chabot (2018) and Otto et al. (2018). The research work of Hoffmann et al. (2007) focuses on three aerodynamic effects that impact the altitude and attitude control of a quadcopter and other interference caused by the various components. A study on the dynamics and control of a quadcopter with variable geometry arms is presented in Barbaraci (2015) where two controllers — an linear quadratic regulator controller with integrator gains and a PID controller — are implemented in the system to reduce tracking error and achieve stability. Many research works in the literature focus on developing controllers for aerial manipulators. A nonlinear hierarchical controller for an aerial manipulator system with model reference adaptive controller (MRAC) to generate control inputs for the quadcopter and a nonlinear adaptive control algorithm to tackle the system's change in mass inertia is proposed in Yilmaz et al. (2019). An adaptive control scheme is proposed in Ali and Li (2020) based on RST controller incorporated with MRAC for an aerial manipulator having a quadcopter with two-DoF manipulator integrated with a gripper and camera for precise grasping and stabilization. In Ali and Li (2019), a dual control approach where an MRAC, augmented with an optimal baseline controller consisting of adaptive control and linear quadratic regulator with proportional-integral feedback connection, is used for controlling and stabilizing a quadcopter with a gripper. A back-stepping controller based on super-twisting extended state observer is designed in Jiao et al. (2020) to attenuate wind gust and model uncertainties in attitude control of a UAV equipped with a robotic arm used in an end-to-end human-UAV interaction system. In Baraban et al. (2020), an MRAC controller ensures trajectory tracking even while payload pickup and maintains flight stability is proposed. An MRAC with an integrator (MRACI) is presented in Ali and Zhangang (2021) having reference dynamics like an observer for a hex-rotor UAV with a cable-driven robotic arm for grabbing application. In Ali and Xinde (2021), MRACI together with a RST control algorithm for a hex-rotor UAV equipped with a 4-DoF manipulator, to eliminate external noise and to stabilize nonlinear dynamics of the aerial manipulator, is presented. A nonlinear multi-input multi-output MRAC for inner loop attitude control is provided in Maki et al. (2020) to dynamically compensate for variations in rotational inertia and centre of gravity owing to the payload of an aerial robot. A detailed literature survey on aerial robots can be found in the literature (Ruggiero et al. 2018; Khamseh et al. 2018; Xilun et al. 2019; Meng et al. 2020; Mohiuddin et al. 2020). Incorporating an adaptive methodology to the feedback linearization controller

(FLC) can significantly improve its efficiency, especially for a real-time system with parameter uncertainties. A nonlinear control strategy for an aerial manipulator consisting of a passive nonlinear dynamic controller for the UAV, an integral kinematic multi-task controller for the manipulator, and an optimizer to coordinate the relative movement between the UAV and the robotic manipulator is designed in [Acosta et al. \(2020\)](#). To ensure practical, collision-free trajectory tracking in a hinged door opening task, a model predictive control is applied on the combined dynamics between the autonomous aerial manipulator and the hinged door with constraints in [Lee et al. \(2020\)](#). In [Anderson et al. \(2020\)](#), the MRAC law for unknown nonlinear switched plants and switched reference models using the Caratheodory and Filippov frameworks is described. Adaptive FLCs are widely used in research, as proposed in [Choi et al. \(2000\)](#), [Shojaei et al. \(2011\)](#), and [Mukherjee and Waslander \(2012\)](#). In this approach, two nonlinear feedback linearization laws are used to cancel the nominal terms in each subsystem. An $\mathcal{L}_1$ adaptive control methodology is then applied to compensate for all model uncertainties and disturbances. The state of art in the field of aerial robotics with respect to the presented work is given in [Table 1](#).

In our paper, a unified dynamics for the quadcopter manipulator system (QMS), with a three-DoF manipulator attached at the bottom of a quadcopter, is developed. This setup is similar to the one described in [Lippiello and Ruggiero \(2012\)](#), in which a Cartesian impedance controller is implemented on a quadcopter with a three-DoF robot arm, subjected to contact forces and environmental disturbances. In [Pierri et al. \(2018\)](#), a hierarchical control scheme for an aerial manipulator is proposed, in which a cascade controller with an adaptive term is designed to mitigate uncertainty and ensure trajectory tracking control. In the presented work, the control law, designed based on FLC, uses estimated system parameters and tracking error to achieve stability and efficient trajectory tracking. The novelty of our work lies in the design and implementation of the proposed “adaptive augmented torque (AAT)” control law for an aerial robot. The presented FLC has an adaptive mechanism based on MRAC to provide better estimates of nonlinear uncertainty parameters in plant dynamics. In addition, an augmented tracking error-based term is included in the control law to provide extra torque to compensate for any internal disturbances induced by nonlinearities that were not taken care of by the cancellation. Thus, the computed torque law has an estimated dynamics component to take care of the nonlinear dynamics of the QMS and an augmented tracking error component to provide additional torque. The integration of the manipulator into the drone can cause disturbances to the drone during manipulator motion and vice versa. The augmented tracking error component is used to counteract these disturbances, which are not adequately taken care of by the nonlinear dynamics cancellation. An adaptive law is obtained using the strictly positive real (SPR)-Lyapunov method, and all the elements in the dynamics equations are estimated for better performance. Since the computed AAT is based on the estimated parameters, the convergence of the estimation error is critical. Another contribution of this work is the mathematical formulation for the bound on estimation error that is developed based on input to state stability analysis. The formulation is intended for flying and articulation of the manipulation without end effector contact. The performance of the controller with adaptive mechanism is validated using ROS/Gazebo and MATLAB simulations. Initial hardware experiments, considering only the motion of manipulator with quadcopter fastened to rods, is also performed for validation.

The rest of the paper is organized as follows. The problem formulation and solution approach with aerial manipulator’s combined kinematics and dynamics are given in Section 2. The proposed nonlinear adaptive controller AAT and the adaptive law are designed in Section 3. Section 4 gives the stability analysis using the Lyapunov theory. The bound on parameter estimation error is derived in Section 5. Finally, Section 6 presents

Table 1. State of-the-art in aerial robotics.

Research work	Application and system	Methodology
Pierri et al. (2018)	Vehicle–arm coordination and management of complex multi-task missions with an aerial manipulator.	An adaptive hierarchical control scheme.
Yilmaz et al. (2019)	Trajectory tracking of aerial manipulation system for a quadrotor endowed with a two-DoF manipulator.	A hierarchical nonlinear control scheme with MRAC in the outer loop and a nonlinear adaptive control in the inner loop.
Jiao et al. (2020)	Field exploration assistance using a hexacopter UAV equipped with a robotic arm.	A back-stepping controller based on super-twisting extended state observer.
Ali and Li (2020)	Precise grasping using aerial manipulator with a gripper and a camera for a quadcopter with two-DoF manipulator.	Adaptive control scheme based on RST controller with MRAC.
Baraban et al. (2020)	Pickup and transport of payload by a quadcopter with two-DoF arm having a magnetic gripper.	An adaptive model reference controller.
Maki et al. (2020)	Payload transport by an aerial robot.	Nonlinear multi-input multi-output MRAC.
Acosta et al. (2020)	Gripping and payload grasping and release by an aerial manipulator.	A nonlinear control strategy consisting of a passive nonlinear dynamic controller for UAV, an integral kinematic multi-task controller for manipulator, and an optimizer to coordinate the relative movement between the two.
Lee et al. (2020)	Opening a hinged door by an autonomous aerial manipulator.	A model predictive control is developed for the combined dynamics between the aerial manipulator and the hinged door.
Anderson et al. (2020)	Mounting a camera of unknown inertial properties to a vertical surface by an aerial robot.	An MRAC law for unknown nonlinear switched plants.
Ali and Xinde (2021)	Clutching the target object during hovering and normal flight operation for a hex-rotor UAV equipped with a four-DoF manipulator.	MRAC with an integrator together with a RST control algorithm.
Ali and Zhangang (2021)	Grabbing and dropping objects from a hex-rotor UAV with a cable-driven robotic arm.	An MRACI with reference dynamics like an observer.
Present work	Capturing images of hard-to-reach places by a quadcopter with a three-DoF robot arm.	Adaptive augmented torque control.

the simulation results, while Section 7 concludes the paper with a summary of the contributions of this work to the body of knowledge.

2. Problem formulation and solution approach

A quadcopter manipulator is an aerial robot consisting of a robotic arm attached to the bottom of a quadcopter. The two subsystems are combined and considered as a single system in this work. The kinematics and dynamics of the integrated system are developed as

Fig. 1. The QMS and the frame of reference attached to each component of the system.

given in our previous work (Sumathy et al. 2018). The aerial robot's mathematical model provides a highly nonlinear plant model.

2.1. Kinematics

In this section, the kinematics of the aerial robot is developed. The aerial vehicle quadcopter is an under-actuated system with four control inputs for executing all six DoF. A three-link serial manipulator connected by revolute joints having three DoF is connected to the quadcopter. In the manipulator, two one-DoF revolute joints connect the three links of the arm with each other and one revolute joint having one DoF connects the base link to the holder that attaches the arm to the drone. Figure 1 shows a schematic of the front view of the aerial robot. Let F_I , F_Q , and F_M denote the inertial, quadcopter, and manipulator frames of reference, respectively, as given in Fig. 1. The generalized coordinate q is defined as $q = [P_Q, \Phi_Q, \eta]^T$ where $P_Q = [x, y, z]^T$ denotes the position of the center of gravity of the quadcopter, $\Phi_Q = [\psi, \theta, \phi]^T$ is the yaw-pitch-roll Euler angles of the UAV and $\eta = [\eta_0, \eta_1, \eta_2]^T$ defines the joint angles of the three-link robotic arm in frames F_{M_0} , F_{M_1} , and F_{M_2} , respectively. The arm is connected to the quadcopter's base plate by a holder on which the motor M_0 is placed. The holder, denoted by L_d , is of length l_d . The base frame for the entire manipulator is an imaginary frame F_M that serves as a transition between F_Q and F_{M_0} . The reference frame attached to the link L_d is considered to have the same orientation as the frame F_M . Link L_0 is connected to the shaft of M_0 . Servo motors connect the manipulator's three links L_0 , L_1 , and L_2 , which have lengths of l_0 , l_1 , and l_2 , respectively, as shown in Fig. 1. The rotary joints are defined with respect to its own corresponding frames F_{M_0} , F_{M_1} , F_{M_2} .

The linear and angular velocities of the system can be derived following steps similar to Lippiello and Ruggiero (2012). The orientation of the body frame F_Q with respect to the inertial frame F_I is given by the rotation matrix $R_{IQ} \in SO(3)$. The linear velocity $\dot{P}_Q$ of the drone can be obtained in the inertial frame. The angular rate in body frame ω_Q is obtained by differentiating the absolute angles θ , ϕ , and ψ , which are defined in their own corresponding vehicle frames (Beard 2008) and using the transformation matrix T_A , as given in

Lippiello and Ruggiero (2012). The angular rates $\omega_Q = [p, q, r]$ defined in the body frame and the rotational velocity with respect to F_Q are

$$(1) \quad R_{IQ} = \begin{bmatrix} \cos \theta \cos \psi & \sin \theta \sin \phi \cos \psi - \sin \psi \cos \phi & \sin \theta \cos \phi \cos \psi - \sin \psi \sin \phi \\ \cos \theta \sin \psi & \sin \theta \sin \phi \sin \psi - \cos \psi \cos \phi & \sin \theta \cos \phi \sin \psi - \cos \psi \sin \phi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix}$$

$$(2) \quad T_A = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix}$$

$$(3) \quad \dot{P}_I = R_{IQ} \dot{P}_Q$$

$$(4) \quad \omega_Q = T_A \dot{\Phi}_Q$$

The linear velocity ($\dot{P}_j$) and the angular velocity (ω_j) of the manipulator are related to the joint velocity ($\dot{\eta}$) through the Jacobian matrix J . The Jacobian matrix J can be partitioned into linear J_{p_j} and angular J_{o_j} sub-matrices. The manipulator's angular velocity is obtained using the Jacobian matrix J_{o_j} . Since all three joints are revolute joints, the angular part of the Jacobian matrix

$$(5) \quad J_{o_j} = z_j$$

where z_j is the unit vector of joint j axis (Siciliano et al. 2010). The matrix J_{o_j} can be computed on the basis of direct kinematics of the manipulator. Let P_I and P_j^Q denote the position of the quadcopter in the F_I frame and arm in F_Q frame with subscript $j = [0, 1, 2]$, respectively. Since link L_d is fixed, it has only translational velocity, computed separately using an equation similar to

$$(6) \quad \dot{P}_j = \dot{P}_I + \dot{R}_{IQ} P_j^Q + R_{IQ} \dot{P}_j^Q$$

with P_d^Q instead of P_j^Q . The translation and rotational velocity of the manipulator links are defined as

$$(7) \quad \omega_j = R_{IQ} \omega_Q + R_{IQ} J_{o_j} \dot{\eta}_j$$

The end-effector of the manipulator is attached to link L_2 and is assumed to have the same orientation as L_2 . This assumption is made to simplify the problem. In the present work, the end-effector is considered as a point mass located at the tip of L_2 . In applications like pick-and-place, an external payload is added to the end-effector. The payload's position and orientation are assumed to be identical to the end-effector in such a scenario.

2.2. Dynamics

In this section, the equations of motion of the combined system are derived using the Euler-Lagrange formulation. The total kinetic energy (K_T) and total potential energy (P_T) of the QMS is

$$(8) \quad K_T = K_Q + \sum_{j=0}^2 K_j$$

$$(9) \quad P_T = m_Q g e_3^T P_I + \sum_{j=0}^2 m_j g e_3^T (P_I + R_{IQ} P_j^Q)$$

where

$$(10) \quad K_Q = \frac{1}{2} \dot{P}_1^T m_Q \dot{P}_1 + \frac{1}{2} \dot{\omega}_Q^T T_A^T R_{IQ} I_Q R_{IQ}^T T_A \dot{\omega}_Q$$

$$(11) \quad K_j = \frac{1}{2} \dot{P}_j^T m_j \dot{P}_j + \frac{1}{2} \omega_j^T (R_{IQ} R_j) I_j (R_{IQ} R_j)^T \omega_j$$

where m_Q and m_j denote the mass, and I_Q and I_j denote the moment of inertia of the quadcopter and the j th link, respectively, where $j \in [0,1,2]$, corresponds to three-DoF manipulator. The energy-based term Lagrangian (L) of the QMS is defined as $L = K_T - P_T$. Once the Lagrangian is obtained, the Euler–Lagrange modeling equation

$$(12) \quad \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau$$

is solved to get the equations of motion of the aerial robot, where L is the Lagrangian and q is the generalised coordinate of the system. The equations of motion of the aerial robot can be expressed similarly to those given in [Lippiello and Ruggiero \(2012\)](#)

$$(13) \quad M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

where the system matrix $M(q)$, known as the mass or inertia matrix, has coefficients corresponding to acceleration terms. Thus, $M(q)$ can be obtained by gathering the coefficients of $\ddot{q}$ in [eq. \(12\)](#). The matrix $C(q,\dot{q})$ involves $\dot{q}_i \dot{q}_j$ and $\dot{q}_j^2$ terms, which are basically the Coriolis and centrifugal force terms. The term $C(q,\dot{q})$ is obtained from the $M(q)$ matrix using

$$(14) \quad C_{ij} = \sum_{k=1}^n C_{ijk} \dot{q}_k$$

where C_{ijk} are known as the Christoffel symbols.

$$(15) \quad C_{ijk} = \frac{1}{2} \left(\frac{\partial M_{ij}}{\partial q_k} + \frac{\partial M_{ik}}{\partial q_j} - \frac{\partial M_{kj}}{\partial q_i} \right)$$

The gravity matrix, denoted by $G(q)$, is the coefficient matrix of gravity terms. In other words

$$(16) \quad G(q) = \frac{\partial (P_T)}{\partial q_j}$$

where the generalized coordinate $q \in \mathbb{R}^{n \times 1}$. In the equation, the system matrices $[M(q)] \in \mathbb{R}^{n \times n}$, $[C(q,\dot{q})] \in \mathbb{R}^{n \times n}$, $[G(q)] \in \mathbb{R}^{n \times 1}$, and $[\tau] \in \mathbb{R}^{n \times 1}$ represent the mass–inertia matrix, Coriolis and centrifugal force matrix, gravity matrix, and the control input matrix, respectively, wherein $n = 9$. According to [eq. \(13\)](#), τ is a 9×1 vector where the first six elements correspond to the quadcopter and the remaining three correspond to the manipulator control inputs. The quadcopter is an under-actuated system and can take only four control inputs. Hence, τ is converted to another vector f using the conversion given in

$$(17) \quad \tau = \Xi f$$

Thus, even though the input τ makes the system look fully actuated as in [eq. \(13\)](#), τ is converted to force vector f to take care of the under-actuation of the quadcopter.

The control input τ applied to the QMS is converted using eq. (17), where $f = [f_Q, f_M]$, $f_Q \in \mathbb{R}^{4 \times 1}$ is the control input to the quadcopter, and $f_M \in \mathbb{R}^{3 \times 1}$ corresponds to each of the motors in the robotic arm. The matrix Ξ is obtained as given in Lippiello and Ruggiero (2012), $\Xi = N_1 N_2$ where $N_1 = \text{diag}(R_{IQ}, T_A^T R_{IQ}, I_3) \in \mathbb{R}^{9 \times 9}$ and $N_2 = \text{diag}(\chi, I_3) \in \mathbb{R}^{9 \times 7}$ and $\chi \in \mathbb{R}^{6 \times 4}$. Here,

$$(18) \quad \chi = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & d & 0 & -d \\ -d & 0 & d & 0 \\ c & -c & c & -c \end{bmatrix}$$

where d is the distance between the motor and the quadcopter center, and $c > 0$ is the drag factor. The matrix $\Xi^T \Xi$ is invertible. For notation, we will omit all dependencies of system matrices and denote them as

$$(19) \quad M\ddot{q} + C\dot{q} + G = \tau$$

The equations of motion of the system have the following structural properties:

1. M is a positive definite symmetric bounded matrix.
2. The matrix term $[\dot{M} - 2C]$ is skew-symmetric. Hence, $x^T [\dot{M} - 2C] x = 0$.
3. The sum $[C + C^T] = \dot{M}$.

The unified system dynamics are highly coupled because of the inter- and intra-coupling within and between the two subsystems. This coupling can be seen, for example, in the representation of the mass matrix M ,

$$(20) \quad M(q) = \begin{bmatrix} M_{pp} & M_{p\Phi} & M_{p\eta} \\ M_{\Phi p} & M_{\Phi\Phi} & M_{\Phi\eta} \\ M_{\eta p} & M_{\eta\Phi} & M_{\eta\eta} \end{bmatrix}$$

wherein the term $M_{p\eta}$ denotes the product of inertia and mass between the states p and η of the QMS, which are the position coordinates of the drone and the joint angles of the manipulator, respectively.

In the QMS dynamics, the two subsystems (quadcopter and the robotic arm) are highly coupled. This coupling can create unmodelled moments or counter-torques on each subsystem due to the motion of the other. Also, the coupling effect can make the parameter estimation less accurate. So the objective is to formulate a control input τ for the QMS to make its output track the reference model output and an adaptive law to estimate the unknown system parameters with better accuracy. The solution approach proposed in this work is to design a FLC to cancel the nonlinearities in the plant dynamics, wherein the estimates of the plant parameters are obtained using MRAC. Also, a tracking error-based term is augmented to the control law to take care of any disturbances caused due to improper cancellation of the nonlinearities.

3. Adaptive FLC

The QMS has a coupled nonlinear dynamics, as stated in the second-order dynamic equation (13). One of the most widely used nonlinear controllers is the FLC. The controller cancels out the plant's nonlinear natural dynamics and converts it to linear error dynamics. However, for this exact cancellation to happen, precise knowledge of the system's nonlinearities is required. The nonlinear matrices $[M, C, G]$, which are functions of the

Fig. 2. Block diagram of proposed adaptive FLC for the aerial robot.

generalized coordinate q , are specific to the system. Also, the system matrices are time-varying. Thus, achieving accurate cancellation of nonlinear dynamics of the system is difficult. An adaptive mechanism, called the model reference adaptive mechanism, is added to the FLC to overcome this problem. The overall nonlinear control architecture for the adaptive FLC with QMS and reference plant is shown in Fig. 2. In this approach, the FLC will have estimated system parameter values, wherein the adaptive mechanism provides the estimated values. The control architecture utilizes a reference plant model that gives the desired response for the external command signal. The adaptive law uses the tracking error, defined as the difference between actual plant and reference model outputs. Thus, the controller has an inner loop to reduce the tracking error and an outer loop to yield the estimated system parameters, as seen in Fig. 2. The blocks in Fig. 2 also give the equation corresponding to each block and these equations will be presented later in the paper.

3.1. Control law

Consider the proposed control law τ

$$(21) \quad \tau = \hat{M}\nu + \hat{C}\dot{q} + \hat{G} - \hat{M}\lambda\dot{e} - \hat{C}z$$

where the signal ν is

$$(22) \quad \nu = [\ddot{q}_r - K_D\dot{e} - K_P e]$$

The dynamics of the reference model are given by

$$(23) \quad \ddot{q}_r + \lambda_1\dot{q}_r + \lambda_2q_r = q_d$$

where λ_1 and λ_2 are diagonal matrices. The quadcopter is an under-actuated system and can only control the position coordinates x, y, z and Euler angle ψ (yaw angle). It cannot directly control Euler angles ϕ (roll angle) and θ (pitch angle). Hence, only the desired position coordinates (x, y, z) and yaw angle can be input to the quadcopter. Thus, the desired input q_d to the QMS is $q_d = [x_d, y_d, z_d, \psi_d, \eta_{0d}, \eta_{1d}, \eta_{2d}]$. The reference values for pitch and roll angles are then computed from the tracking error in position coordinates x and y , respectively. The tracking error $e \in \mathbb{R}^{n \times 1}$ is defined as $e = q - q_r$. The proportional and the derivative gains, K_P and K_D , respectively, are positive diagonal constant matrices $\in \mathbb{R}^{n \times n}$. We define a new quasi-state z as

$$(24) \quad z = \dot{e} + \lambda e$$

where design parameter λ is a positive constant. The quasi state z is obtained by a linear transformation function applied on the tracking error. The quasi-state is introduced to aid analysis of the plant model obtained after applying the proposed controller.

Let matrices $[a] = [M, C, G]$ and $[\hat{a}] = [\hat{M}, \hat{C}, \hat{G}]$ define the actual and estimated values of system parameters, respectively, where $[a] \in \mathbb{R}^{n \times (2n+1)}$ and $[\hat{a}] \in \mathbb{R}^{n \times (2n+1)}$. Then, the parameter estimation error matrix $[\tilde{a}] = [\tilde{M}, \tilde{C}, \tilde{G}] \in \mathbb{R}^{n \times (2n+1)}$, is defined as $\tilde{a} = \hat{a} - a$.

The computed control law τ is defined as the AAT and is based on the estimated values of the system parameters. The torque control law has two parts

$$(25) \quad \tau = \tau_d + \tau_e$$

where $\tau_d = \hat{M}\nu + \hat{C}\dot{q} + \hat{G}$ accounts for the feedback linearization part, and $\tau_e = -(\hat{M}\lambda\dot{e} + \hat{C}z)$ adds an additional torque based on the tracking error. As the estimation error converges, τ_d in the AAT will cancel the nonlinear dynamics in eq. (13) and give a second-order error equation. The second term τ_e , which is a function of tracking error, will add to the error equation's coefficients, whose proportional and derivative gains can be tuned to get better tracking error convergence. Thus, the computed torque control law is designed to take care of both nonlinear dynamics and disturbances between the subsystems.

3.2. Applying the control law

The proposed nonlinear control law eq. (21) is applied to the plant. By substituting for τ in eq. (19)

$$(26) \quad M\ddot{q} + C\dot{q} + G = \hat{M}\nu + \hat{C}\dot{q} + \hat{G} - \lambda\hat{M}\dot{e} - \hat{C}z$$

Rearranging the terms we get,

$$(27) \quad M\ddot{q} = \hat{M}\nu + \hat{C}\dot{q} + \hat{G} - \lambda\hat{M}\dot{e} - \hat{C}z - C\dot{q} - G$$

By adding and subtracting the terms $M\nu$, $\lambda M\dot{e}$, and Cz to the above equation, and using the definition of parameter estimation error

$$(28) \quad M\ddot{q} = \tilde{M}\nu + \tilde{C}\dot{q} + \tilde{G} - \lambda\tilde{M}\dot{e} - \tilde{C}z + M\nu - \lambda M\dot{e} - Cz$$

Collecting all estimation error terms $(\tilde{M}, \tilde{C}, \tilde{G})$

$$(29) \quad M\ddot{q} - M\nu + \lambda M\dot{e} + Cz = \tilde{M}\nu + \tilde{C}\dot{q} + \tilde{G} - \lambda\tilde{M}\dot{e} - \tilde{C}z$$

Replacing the signal ν by eq. (22) and applying $\ddot{e} = \ddot{q} - \ddot{q}_r$ gives

$$(30) \quad M[\ddot{e} + (K_D + \lambda I_n)\dot{e} + K_P e] + Cz = \tilde{M}(\nu - \lambda\dot{e}) + \tilde{C}(\dot{q} - z) + \tilde{G}$$

Let $\Omega = [\nu - \lambda\dot{e}, \dot{q} - z, 1]^T$. Then

$$(31) \quad \tilde{a}\Omega = \tilde{M}(\nu - \lambda\dot{e}) + \tilde{C}(\dot{q} - z) + \tilde{G}$$

By choosing $K_D = \lambda I_n$ and $K_P = \lambda^2 I_n$, where $I_n \in \mathbb{R}^{n \times n}$ and using eq. (31) in eq. (30) gives

$$(32) \quad M[\ddot{e} + 2\lambda\dot{e} + \lambda^2 e] + Cz = \tilde{a}\Omega$$

Applying the definition of quasi-state z in the above equation

$$(33) \quad M\dot{z} + (\lambda M + C)z = \tilde{a}\Omega$$

The above equation can be rewritten as

$$(34) \quad \dot{z} = -(\lambda I_n + M^{-1}C)z + M^{-1}\tilde{a}\Omega$$

Re-writing the equation in the state-space form $\dot{z} = A_z z + B_z u$, where the state matrix $A_z = -(\lambda I_n + M^{-1}C)$ and the control input matrix $B_z = M^{-1}$. Since the plant's output is the same as its states, the output matrix $C_z = I_n$. The control input $u = \tilde{a}\Omega$. The control input has quasi-state z in the second term in the Ω vector. This term can be rewritten using the definitions of quasi-state z and tracking error e as

$$(35) \quad \Omega = [\nu - \lambda\dot{e} \quad \dot{q} - z \quad 1]^T$$

$$(36) \quad \Omega = [\nu - \lambda\dot{e} \quad \dot{q} - (\dot{e} + \lambda e) \quad 1]^T$$

$$(37) \quad \Omega = [\nu - \lambda\dot{e} \quad \dot{q} - (\dot{q} - \dot{q}_r + \lambda e) \quad 1]^T$$

$$(38) \quad \Omega = [\nu - \lambda\dot{e} \quad \dot{q}_r - \lambda e \quad 1]^T$$

Thus, the control input $u = \tilde{a}\Omega$ in [eq. \(34\)](#) is not dependant on the quasi-state z .

3.3. Adaptive law

The parameter estimation error should ideally go to zero to achieve the desired response from the closed-loop system. That is, the estimated parameter values should become equal to the actual values of the system. The FLC can easily cancel out all system nonlinearities and convert them to linear tracking error dynamics. Nevertheless, in a practical system, it is challenging. In this section, an adaptive update law to find the estimated values of system parameters is developed using the SPR-Lyapunov approach ([Shojaei et al. 2011](#), [Ioannou and Datta 1991](#)). In this method, a dynamic equation is initially derived in terms of tracking error and parameter estimate error. Then, by using the Lyapunov analysis, the adaptive law is developed. The dynamics equation of the quasi-state z , as in [eq. \(34\)](#)

$$(39) \quad \dot{z} = -(\lambda I_n + M^{-1}C)z + M^{-1}\tilde{a}\Omega$$

The dynamics [equation \(39\)](#) can be written in the state-space form as

$$(40) \quad \dot{z} = A_z z + B_z u$$

where the state matrix A_z is

$$(41) \quad A_z = -(\lambda I_n + M^{-1}C)$$

All eigenvalues of system matrix A_z have $\text{Re}[\sigma_i] < 0$. Thus, A_z is Hurwitz. The state-space [eq. \(39\)](#) can be written in transfer function form as $z = G(s)\tilde{a}\Omega$, where $G(s)$ is a strictly positive real transfer function. Now, by using the SPR-Lyapunov analysis method, the parameter update law for $\tilde{a}$ can be obtained as

$$(42) \quad \dot{\tilde{a}} = -\gamma z \Omega^T$$

Then, by the results from [Slotine and Li \(1991\)](#), for a positive constant γ , signals z and $\tilde{a}$ are globally bounded. Also, the quasi-state z goes to zero for a bounded Ω value as time tends to infinity. Hence, the tracking error e will also become zero eventually.

$$(43) \quad z \rightarrow 0 \quad \text{as} \quad t \rightarrow \infty$$

Thus, eq. (42) gives the update rule for the parameter estimation error.

4. Lyapunov stability analysis

In this section, the closed-loop system stability analysis is done with the proposed control law and developed adaptive law using the Lyapunov method.

Theorem 1 Consider the plant dynamics eq. (19), with the structural properties mentioned in structural properties in Subsection 2.2. Let the external command $[q_d, \dot{q}_d, \ddot{q}_d]$ given to the system be measurable and bounded. We can now define the torque given to the plant as in eq. (21)

$$(44) \quad \tau = \hat{M}[\ddot{q}_r - K_D \dot{e} - K_P e] + \hat{C}\dot{q} + \hat{G} - \lambda \hat{M}\dot{e} - \hat{C}z$$

and an adaptive law as in eq. (42)

$$(45) \quad \dot{\hat{a}} = -\gamma z \Omega^T$$

Under the proposed control law τ and the developed adaptive law, the closed-loop system is asymptotically stable, and the tracking errors $[e, \dot{e}, \ddot{e}] \rightarrow 0$ as $t \rightarrow \infty$.

Proof. Consider the candidate Lyapunov function V as

$$(46) \quad V = \frac{1}{2} z^T M z + \frac{1}{2} \text{Tr} \left(\frac{1}{\gamma} \tilde{a} \tilde{a}^T \right)$$

The candidate Lyapunov function's two terms can be denoted using V_1 and V_2 as $V_1 = (1/2) z^T M z$ and $V_2 = (1/2) \text{Tr}[(1/\gamma) \tilde{a} \tilde{a}^T]$. Upon differentiating V_1

$$(47) \quad \dot{V}_1 = z^T \dot{M} z + \frac{1}{2} z^T \dot{M} z$$

Using the definition $\dot{z} = \ddot{e} + \lambda \dot{e}$ and expanding the tracking error term

$$(48) \quad \dot{V}_1 = z^T M [(\ddot{q} - \ddot{q}_r) + \lambda \dot{e}] + \frac{1}{2} z^T \dot{M} z$$

Adding and subtracting $z^T C z$ and re-arranging the equations gives

$$(49) \quad \dot{V}_1 = z^T (M\ddot{q} - M\ddot{q}_r) + z^T \lambda M \dot{e} + \frac{1}{2} z^T (\dot{M} - 2C) z + z^T C z$$

Applying the skew-symmetric property, $z^T (\dot{M} - 2C) z = 0$

$$(50) \quad \dot{V}_1 = z^T [M\ddot{q} - M\ddot{q}_r + \lambda M \dot{e} + C z]$$

Substituting system dynamics equation (19) and applying proposed control law eq. (21) gives

$$(51) \quad \dot{V}_1 = z^T [\hat{M}\nu + \tilde{C}\dot{q} + \tilde{G} - \lambda \tilde{M}\dot{e} - \tilde{C}z - M\ddot{q}_r]$$

Adding and subtracting the term $M\nu$ and applying the estimation error definition to the mass matrix, gives

$$(52) \quad \dot{V}_1 = z^T [\tilde{M}\nu + \tilde{C}\dot{q} + \tilde{G} - \lambda \tilde{M}\dot{e} - \tilde{C}z - M\ddot{q}_r + M\nu]$$

On expanding the signal v as in eq. (22) and using the definition of $\tilde{a}\Omega$ as in eq. (31), $\dot{V}_1$ can be written as

$$(53) \quad \dot{V}_1 = z^T [\tilde{a}\Omega - M(K_D \dot{e} + K_P e)]$$

Now the second term in the candidate Lyapunov function can be differentiated as

$$(54) \quad \dot{V}_2 = \frac{1}{\gamma} \text{Tr}(\dot{\tilde{a}}\tilde{a}^T)$$

Applying the linear property of trace, $(d/dt)(\text{Tr}[\cdot]) = \text{Tr}[(d/dt)[\cdot]]$ and substituting the adaptive law eq. (42)

$$(55) \quad \dot{V}_2 = \frac{1}{\gamma} \text{Tr}[\tilde{a}(-\gamma z \Omega^T)^T]$$

By using the cyclic property of the trace of a matrix, $\text{Tr}(\tilde{a}\Omega z^T) = \text{Tr}(z^T \tilde{a}\Omega)$

$$(56) \quad \dot{V}_2 = -\text{Tr}(z^T \tilde{a}\Omega)$$

Here, $z \in \mathbb{R}^{n \times 1}$, $\tilde{a} \in \mathbb{R}^{n \times (2n+1)}$, and $\Omega \in \mathbb{R}^{(2n+1) \times 1}$. Then, by dimension $[z^T \tilde{a}\Omega]$ is a scalar. Thus

$$(57) \quad \dot{V}_2 = -z^T \tilde{a}\Omega$$

The total derivative of the candidate Lyapunov function $V(z, \tilde{a})$ can then be obtained as the sum of eqs. (53) and (57).

$$(58) \quad \dot{V} = z^T [\tilde{a}\Omega - M(K_D \dot{e} + K_P e)] - z^T \tilde{a}\Omega$$

Upon simplification

$$(59) \quad \dot{V} = z^T [-M(K_D \dot{e} + K_P e)]$$

By substituting for the values K_P and K_D

$$(60) \quad \dot{V} = z^T [-M(\lambda \dot{e} + \lambda^2 e)]$$

Applying the definition of z

$$(61) \quad \dot{V} = -z^T \lambda M z$$

Using the property of norm, a bound on $\dot{V}$ can be obtained as

$$(62) \quad \dot{V}(t, z, \tilde{a}) \leq -\lambda \sigma_{\min}(M) \|z\|^2 \leq 0$$

where $\sigma_{\min}(M)$ denotes the smallest eigenvalue of the mass matrix M . The derivative of the candidate Lyapunov function being semi-negative definite, $\dot{V} \leq 0$, implies that V is a decreasing function of time. Since z and $\tilde{a}$ are bounded, by the definition of z tracking error and its derivatives are also bounded for a given bounded external command. The candidate Lyapunov function V is lower bounded, $\dot{V}$ is negative semi-definite, and $\dot{V}$ is uniformly continuous as $\ddot{V}$ is bounded. Then, using Barbalat's Lemma

$$(63) \quad \dot{V} \rightarrow 0 \quad \text{as} \quad t \rightarrow \infty$$

In other words, the tracking error will become zero as time tends to infinity. Thus, the error converges. $\square$

The elements in matrix “a” are not constant as they are functions of time-varying states. In the proposed methodology, the elements in M, C, and G matrices are estimated using an adaptive law based on the SPR-Lyapunov method, as given by eq. (42). This method ensures that the value of the estimates will always remain bounded for a positive adaptive gain. Also, in the next section, we will derive the bound on estimation error. Hence, the tracking error will converge asymptotically as long as the estimates are within the bound.

5. Bound on parameter estimation error

In this section, a bound on parameter estimation error $\tilde{a}$, an important contribution of this work, is derived. For this purpose, the following proposition from Khalil and Grizzle (2002), stated without proof, will be useful.

Proposition 1 (Khalil and Grizzle 2002) *For the linear non-autonomous system (39), there exists a continuously differentiable, positive definite, symmetric, bounded matrix $P(t)$, which satisfies the Lyapunov equation,*

$$(64) \quad \dot{P}(t) + A_z^T(t)P(t) + P(t)A_z(t) = -Q(t)$$

where the solution to the equation, $Q(t)$, is a continuous, symmetric, positive definite matrix, with

$$(65) \quad 0 < d_1 I \leq P(t) \leq d_2 I \quad \forall t \geq 0$$

$$(66) \quad Q(t) \geq d_3 I > 0 \quad \forall t \geq 0$$

Here, d_1 and d_2 are the minimum and maximum eigenvalues of matrix $P(t)$, respectively. Similarly, d_3 is the minimum eigenvalue of $Q(t)$.

For the given system eq. (39), and from eq. (40) and eq. (41)

$$(67) \quad A_z^T(t)M(t) + M(t)A_z(t) = -[\lambda I_n + M(t)^{-1} C(t)]^T M(t) - M(t)[\lambda I_n + M(t)^{-1} C(t)]$$

$$(68) \quad A_z^T(t)M(t) + M(t)A_z(t) = -\lambda M(t) - C(t)^T M(t)^{-1} M(t) - \lambda M(t) - M(t)M(t)^{-1} C(t)$$

$$(69) \quad A_z^T(t)M(t) + M(t)A_z(t) = -2\lambda M(t) - (C(t)^T + C(t))$$

Now by applying the property $(C^T + C) = \dot{M}$ as given in structural properties in Subsection 2.2,

$$(70) \quad A_z^T(t)M(t) + M(t)A_z(t) = -2\lambda M(t) - \dot{M}(t)$$

Upon rearranging, the equation can be written as,

$$(71) \quad A_z^T(t)M(t) + M(t)A_z(t) + \dot{M}(t) = -2\lambda M(t)$$

Thus, by Proposition 1, system (39) satisfies the Lyapunov equation (64) with solution $Q(t) = 2\lambda P(t)$ with $P(t) = M(t)$.

The non-autonomous system (39) can be written in the general form

$$(72) \quad \dot{z} = f(t, z, u)$$

where $f: [0, \infty) \times \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ is piece-wise continuous in t and locally Lipschitz in z and u . The input u is piece-wise continuous and bounded for all t . The system (72) can be considered as an unforced system with a perturbation, where the unforced system is given as

$$(73) \quad \dot{z} = f(t, z) = A_z(t)z$$

The bound on parameter estimation error $\tilde{a}$ is obtained using the concept of *input to state stability*. The system (72) is input to state stable if $f(t, z, u)$ is continuously differentiable and globally Lipschitz in (z, u) , uniform in t , and if its unforced system (73) has a globally exponentially stable equilibrium point at origin $z = 0$. The function $f(t, z, u)$ is continuously differentiable and globally Lipschitz in (z, u) . To ensure system (72) is input to state stable, we need to prove that system (73) has a globally exponentially stable equilibrium point at the origin $z = 0$. Below, we will show global exponential stability of the equilibrium point using the Lyapunov stability concept.

Consider the unforced state-space form of the system

$$(74) \quad \dot{z} = -[\lambda I_n + M^{-1}(t)C(t)]z$$

which can be written as $\dot{z} = A_z z$. Let $z = 0$ be an equilibrium point of the above system and $\mathbb{D}$ be a domain containing the equilibrium point, $\mathbb{D} \subset \mathbb{R}$. Consider the candidate Lyapunov function $V : [0, \infty) \times \mathbb{D} \rightarrow \mathbb{R}$, that is continuously differentiable as

$$(75) \quad V(t, z) = z^T M(t) z$$

where $M(t)$ is a positive definite symmetric matrix that satisfies the Lyapunov equation (71). Taking derivative

$$(76) \quad \dot{V}(t, z) = [\dot{z}^T M(t) z + z^T \dot{M}(t) z + z^T \dot{M}(t) z]$$

Substituting for $\dot{z}$ using the state equation gives

$$(77) \quad \dot{V}(t, z) = z^T [A_z(t)^T M(t) + M(t) A_z(t) + \dot{M}(t)] z$$

which is the Lyapunov equation, as in eq. (71), and can be substituted by its solution as

$$(78) \quad \dot{V}(t, z) = z^T [-2\lambda M(t)] z$$

Thus, the derivative of the candidate Lyapunov function is negative semi-definite and

$$(79) \quad \dot{V}(t, z) \leq -\sigma_{\min}(Q(t)) \|z\|^2$$

where $Q(t)$ is the general notation for the solution of the Lyapunov equation, which in our case is $Q(t) = 2\lambda M(t)$. Here $\sigma_{\min}(Q(t))$ represents the minimum eigenvalue of $Q(t)$. The candidate Lyapunov function is bounded as

$$(80) \quad 0 < \sigma_{\min}(M(t)) \|z\|^2 \leq V(t, z) \leq \sigma_{\max}(M(t)) \|z\|^2$$

$$(81) \quad 0 < c_1 \|z\|^2 \leq V(t, z) \leq c_2 \|z\|^2$$

Also, the derivative of $V(t, z)$ is

$$(82) \quad \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(t, z) \leq -\sigma_{\min}(Q(t)) \|z\|^2$$

Then, by Khalil and Grizzle (2002), the origin $z = 0$ is globally exponentially stable, and thus, the forced system (39) is input to state stable.

Now consider the input-driven plant equation (39) with input u . For this system, the derivative of the candidate Lyapunov function in eq. (75) becomes

$$(83) \quad \dot{V}(t, z) = \dot{M}(t)z + z^T \dot{M}(t)z + z^T \dot{M}(t)z$$

Substituting for $\dot{z}$ and using the Lyapunov [equation \(71\)](#)

$$(84) \quad \dot{V}(t, z) = z^T \{-2\lambda M(t)\}z + u^T B_z(t)^T M(t)z + z^T M(t) B_z(T)u$$

which can be simplified by collecting common terms and substituting for $B_z(t)$ as $B_z(t) = M^{-1}(t)$

$$(85) \quad \dot{V}(t, z) = z^T \{-2\lambda M(t)\}z + \{(z^T u)^T + z^T u\}$$

Here $z \in \mathbb{R}^{n \times 1}$ and $u \in \mathbb{R}^{n \times 1}$. Hence, $z^T u$ is a scalar.

$$(86) \quad \dot{V}(t, z) = z^T \{-2\lambda M(t)\}z + 2z^T u$$

Then the derivative becomes

$$(87) \quad \dot{V}(t, z) \leq -\sigma_{\min}(2\lambda M(t))\|z\|^2 + 2\|z^T\|\|u\|$$

Now for $\dot{V}(t, z) \leq 0$, the first term $\left[\sigma_{\min}(2\lambda M(t))\|z\|^2\right]$ should be greater than $\left[2\|z^T\|\|u\|\right]$. [Equation \(87\)](#) can be rewritten as

$$(88) \quad \dot{V}(t, z) \leq -c_3\|z\|^2 + 2\|z^T\|\|u\|$$

which by mathematical manipulation becomes

$$(89) \quad \dot{V}(t, z) \leq -c_3(1 - \delta)\|z\|^2 - c_3\delta\|z\|^2 + 2\|z^T\|\|u\|$$

where $c_3 = \sigma_{\min}(Q(t))$, δ is a positive constant, $0 < \delta < 1$. Then

$$(90) \quad \dot{V}(t, z) \leq -c_3(1 - \delta)\|z\|^2 \quad \forall \|z\| \geq \frac{2\|u\|}{c_3\delta}$$

Since the non-autonomous system (39) is input to state stable, the condition in [eq. \(90\)](#) should always be satisfied. That is,

$$(91) \quad \|z\| \geq \frac{2\|u\|}{c_3\delta}$$

The above condition can be modified as

$$(92) \quad \|u\| \leq \frac{1}{2}\|z\|c_3\delta$$

By substituting the control input $u = \tilde{a}\Omega$

$$(93) \quad \|\tilde{a}\Omega\| \leq \frac{1}{2}\|z\|c_3\delta$$

$$(94) \quad \|\tilde{a}\| \leq \frac{\|z\|c_3\delta}{2\|\Omega\|}$$

Thus, $\tilde{a}$ should always be bounded

$$(95) \quad \|\tilde{a}\| \leq \beta$$

Table 2. Sequence of events in the task.

Phase	QMS pose	Time (s)
Phase ₀	$[P_g, \eta_{fo}] - [P_0, \eta_{io}]$	0–40
Phase ₁	$[P_0, \eta_{io}] - [P_1, \eta_{so}]$	40–160
	$[P_1, \eta_{so}] - [P_1, \eta_{io}]$	160–220
Phase ₂	$[P_1, \eta_{io}] - [P_2, \eta_{so}]$	220–340
	$[P_2, \eta_{so}] - [P_2, \eta_{io}]$	340–400
Phase ₃	$[P_2, \eta_{io}] - [P_3, \eta_{so}]$	400–520
	$[P_3, \eta_{so}] - [P_3, \eta_{io}]$	520–580
Phase ₄	$[P_3, \eta_{io}] - [P_4, \eta_{so}]$	580–700
	$[P_4, \eta_{so}] - [P_4, \eta_{io}]$	700–760
Phase ₅	$[P_4, \eta_{io}] - [P_g, \eta_{fo}]$	760–820

where

$$(96) \quad \beta = \frac{\|z\|c_3\delta}{2\|\Omega\|}$$

Equation (96) gives the bound on parameter estimation error $\tilde{a}$, for the non-autonomous forced system to be input to state stable.

6. Simulation results

This section presents simulation results to validate the adaptive FLC for the QMS in MATLAB and the ROS/Gazebo platform. The implementation module includes an inverse kinematics block and a trajectory planner. The inverse kinematics block computes the desired position and orientation of the drone and joint angles for the given target location. Then, the trajectory planner calculates a minimum acceleration trajectory for the quadcopter and the robotic arm based on a motion planning algorithm. The algorithm considers all possible quadcopter configurations and the reachable workspace of the manipulator. It thereby ensures the feasibility of the computed trajectories. The combined dynamics of the QMS as in eq. (13) are used in MATLAB to generate the simulation. In ROS, an iris drone with a three-DoF robotic arm attached to the centre of gravity at the bottom is used as the aerial robot.

In the given case study, the QMS is commanded to perform a given assignment at four different target locations. The additional force and the reaction is not considered in this work. We assume that getting the end-effector to the target point is the main objective. The force due to actuation of the end effector and its interaction with the environment is not considered. The complete task profile is shown in Table 2. The sequence of events in the task profile is divided into six phases: phase₀, phase₁, phase₂, phase₃, phase₄, and phase₅. In each phase, the quadcopter and manipulator are required to achieve the desired transition in states within the time slot. The vector $[P_i, \eta_j]$, where $P_i = [x, y, z, \psi]$ and $\eta_j = [\eta_0, \eta_1, \eta_2]$, denotes the pose of the QMS, which is the position of the quadcopter and the manipulator's orientation at any given point in time. The pose in each phase are denoted by subscripts $i \in \{g, 0, 1, 2, 3, 4\}$ and $j \in \{fo, io, so\}$, where $P_g = [0, 0, 0, 0]$ is the ground position, $P_0 = [0, 0, 1, 0]$ is the initial position and P_1, P_2, P_3 , and P_4 denote the target locations that the quadcopter needs to visit. Similarly, the subscripts in η_j denote the folded orientation $fo = [0, 0, 0]$, intermediate orientation $io = [0, -\pi/4, 0]$ and stretched orientation $so = [\eta_0, 0, \pi/2]$ of the

Fig. 3. ROS/Gazebo simulation of the QMS inside a building with ArUco marker as the target. (a) Phase₀, (b) phase₁, (c) phase₂, (d) phase₃, (e) phase₄, and (f) phase₅. In phase₀ the QMS lifts off to 1 m altitude, and the manipulator changes to the intermediate configuration (η_{i0}). The QMS then moves to target locations P_1 , P_2 , P_3 , and P_4 in phase₁, phase₂, phase₃, and phase₄, respectively. In each phase, at the target location, the manipulator orients to stretched orientation (η_{s0}). The end-effector camera then takes the image of the target ArUco marker, as seen at the end-effector camera view. Finally, in phase₅, the QMS comes back to its home location.

robotic arm. In the stretched orientation of the arm, the value of joint angle η_0 depends on the target location.

The system first lifts off in phase₀ and moves to the desired target position P_i within the given time slot. At each target location, the manipulator orients to the desired η_j and then perform the task. In this simulation, the assigned task is to get an image of the ArUco marker on the compound wall using the WGE100 camera, attached to the tip of L_2 as the end-effector. After each task, the robotic arm folds back to its intermediate orientation (η_{i0}),

Fig. 4. The trajectory achieved by the quadcopter for the given task.

and the QMS then moves to the next target. The advantage of orienting to η_{io} is that the robotic arm joint η_0 can change to any desired quadrant without risking the arm interacting with the environment or the drone downwash. Once phase₁, phase₂, phase₃, and phase₄ are performed, the QMS changes to the initial configuration and lands back at the starting point in phase₅. In the simulation, $P_1 = [2, 0, 2, 0]$, $P_2 = [2, 2, 2, 0]$, $P_3 = [0, 2, 2, 0]$, and $P_4 = [-2, 2, 2, 0]$. Each variable in q changes one at a time and takes $t = 20$ s to achieve its desired value. No two states change simultaneously. The values of design parameters for FLC and adaptive law in the MATLAB simulation are taken as $\lambda = 1$ and $\gamma = 0.01$. The reference model has parameter values $\lambda_1 = 25$ and $\lambda_2 = 6$.¹

The corresponding implementation in the ROS/Gazebo platform is shown in Fig. 3, wherein the QMS is made to fly within a closed indoor space of $8 \text{ m} \times 8 \text{ m} \times 2.5 \text{ m}$. Figures 3a–3f show ROS/Gazebo simulation of phase₀ and phase₅, respectively. The targets are marked with ArUco markers and are placed at four different locations on the walls of the building. Two targets are placed on the front side wall, denoted Target 1 and Target 2, and two on the left side wall marked as Target 3 and Target 4. Two cameras, marked in Fig. 3 as wall camera 1 and wall camera 2, are mounted on the right and rear building walls. These two cameras are used only for recording the motion of the QMS and their

¹Complete video of the experimental simulation can be seen at: <https://youtu.be/kMPY8xfRacs>.

Fig. 5. Tracking error in (a) drone position, (b) drone orientation, and (c) manipulator joint angles.

information is not used for control. The image output from the WGE100 camera and the cameras on the building walls, along with their zoomed views, are shown in Fig. 3.

Figure 4 shows the QMS's complete path from the initial position to the final position and back. The four target points are marked as data points in the figure. At each target point, the robotic arm orients from η_{io} to η_{so} and then back to η_{io} . The different configurations of the robotic arm and the quadcopter are observed and recorded in the simulation. The initial value of parameter estimation error is taken as $\tilde{a} = 1\%$. The tracking error during one of the runs is shown in Fig. 5. The coupling effect between the subsystems can be observed at different times. The quadcopter hovers at each target location, and the manipulator then stretches out to perform the task. During the motion of the manipulator, a disturbance moment is created on the quadcopter, affecting the drone's position. The tracking error in the pose of the quadcopter increases during manipulator motion, as

Fig. 6. Effect of parameter estimation error on tracking error in (a) drone position, (b) drone orientation, and (c) manipulator joint angles.

seen in Figs. 5a and 5b. Initially, the quadcopter lifts off to 1 m, and the arm changes its orientation to η_{i0} . This arm motion creates a disturbance moment on the vehicle altitude, as shown in the e_z plot in Fig. 5a. The desired yaw angle $\psi = 0$ always, but due to disturbance, an error always persists, as seen in Fig. 5b. The motion of the quadcopter from one position to the other also causes disturbance to the manipulator orientation. When the drone changes its position from one target to another, the manipulator must maintain η_{i0} orientation. However, because of the vehicle's pitching and rolling, the robotic arm's servo motors experience external disturbances, as seen in Fig. 5c. The proposed controller's efficacy is reflected by the minimum error in the trajectory tracking error plots, even during counter-reactions. The generated augmented adaptive torque control input can minimize the reaction torques and moments between the two subsystems.

The effectiveness of the adaptive controller is analyzed by varying the initial values of the parameter estimation error. The adaptive law tries to minimize the parameter estimation error so that the estimated value is almost the same as the actual value, in

Fig. 7. The QMS hardware experimental setup. **Fig. 8.** The three orientations of the manipulator: (a) folded orientation, (b) intermediate orientation, and (c) stretched orientation.

other words the values of matrices a and $\hat{a}$ are almost equal. Let d denote the percentage error in $\tilde{a}$. Figure 6 shows the tracking error in the pose of QMS for different values of d . Figure 6a, 6b, and 6c show the tracking error in quadcopter position, orientation, and manipulator joint angles, respectively, for the task as mentioned above. The plots show that as the estimation error percentage increases, the tracking error also increases. In the ideal case at $d = 0\%$, when the computed value is equal to the actual system parameter values, the tracking error is a minimum. But since achieving the exact values is not practically possible, we consider different cases with slight variations, $d = [0.1\%, 0.5\%, 1\%, 5\%]$. As the value of $\tilde{a}$ increases, the closeness of the estimated parameter values to the actual values decreases, and thus the tracking error increases.

Fig. 9. The robotic arm joint angles trajectory tracking from HIL experimental simulation.

The hardware experimental setup for testing the proposed AAT controller is given in Fig. 7. The setup consists of a custom-made drone and a 3D-printed three-DoF manipulator attached at the bottom of the aerial vehicle. Initial results obtained from some hardware-in-the-loop (HIL) experiments are given in this paragraph. However, the complete HIL results for all the different components in the work are beyond the scope of the present paper. The experimental setup consists of the QMS fastened to rods attached to the ceiling, as in Fig. 7. In the presented experimental study, the trajectory tracking control of the manipulator is demonstrated. A Raspberry Pi 3 is used as the companion computer. The Dynamixel motors, AX-18A and MX-28T, are used as the servo motors where AX-18A is used as servo motors M_1 and M_3 and MX-28T for motor M_2 . The three motors are interconnected by daisy-chain and connected to the Raspberry Pi via a U2D2 communication converter. The arm is powered using 3s Lipo battery. The three given orientations achieved by the joints, namely, the folded orientation, intermediate orientation, and stretched orientation, are shown in Fig. 8a, 8b, and 8c, respectively. Figure 9 shows the trajectory tracked by the three motors in real-time HIL simulation, during the complete task profile. As can be seen, the three motors achieve the desired joint angles with very little error. The four instants where the manipulator achieves stretched orientation at each target can be identified by the η_2 plot with joint angle $\eta_2 = \pi/2$.

7. Conclusion

In this work, a nonlinear FLC with a proportional derivative control unit in combination with a model reference adaptive control mechanism is designed for the combined dynamics of a QMS. The computed AAT control law, having an estimated dynamics part and an augmented error part, is proposed for the aerial robot's stability and trajectory tracking control. An adaptive law for estimating unknown system parameters is obtained using the SPR-Lyapunov method. The stability analysis of the closed-loop system is done using the Lyapunov stability method. A bound on parameter estimation error is derived using the input to state stability concept. Later, the controller is implemented in ROS/Gazebo and MATLAB, and the simulation results are analyzed. The QMS trajectory tracking plots with minimal error illustrate the efficacy of the proposed adaptive FLC. The effect of parameter estimation error on the trajectory tracking control is also demonstrated. Finally, the

results obtained from initial hardware experiments are demonstrated along with the tracking plots from real-time experiment. The complete HIL results for all the different components in this work are proposed as future work along with field experiments.

References

- Acosta, J., de Cos, C., and Ollero, A. 2020. Accurate control of aerial manipulators outdoors. A reliable and self-coordinated nonlinear approach. *Aerosp. Sci. Technol.* **99**: 105731. doi: [10.1016/j.ast.2020.105731](https://doi.org/10.1016/j.ast.2020.105731).
- Ali, Z.A., and Li, X. 2019. Modeling and controlling of quadrotor aerial vehicle equipped with a gripper. *Meas. Control*, **52**: 577–587. doi: [10.1177/0020294019834040](https://doi.org/10.1177/0020294019834040).
- Ali, Z.A., and Li, X. 2020. Controlling of an under-actuated quadrotor UAV equipped with a manipulator. *IEEE Access*, **8**: 34664–34674. doi: [10.1109/ACCESS.2020.2974581](https://doi.org/10.1109/ACCESS.2020.2974581).
- Ali, Z.A., and Xinde, L. 2021. Modeling and controlling the dynamic behavior of an aerial manipulator. *Fluctuation Noise Lett.* **20**: 2150044. doi: [10.1142/S0219477521500449](https://doi.org/10.1142/S0219477521500449).
- Ali, Z.A., and Zhangang, H. 2021. Maneuvering control of hexrotor UAV equipped with a cable-driven gripper. *IEEE Access*, **9**: 65308–65318. doi: [10.1109/ACCESS.2021.3076129](https://doi.org/10.1109/ACCESS.2021.3076129).
- Anderson, R.B., Marshall, J.A., L'Afflito, A., and Dotterweich, J.M. 2020. Model reference adaptive control of switched dynamical systems with applications to aerial robotics. *J. Intell. Rob. Syst.* **100**: 1265–1281. doi: [10.1007/s10846-020-01260-7](https://doi.org/10.1007/s10846-020-01260-7).
- Baraban, G., Sheckells, M., Kim, S., and Kobilarov, M. 2020. Adaptive parameter estimation for aerial manipulation. In *2020 American Control Conference (ACC)*, IEEE. pp. 614–619.
- Barbaraci, G. 2015. Modeling and control of a quadrotor with variable geometry arms. *J. Unmanned Veh. Syst.* **3**: 35–57. doi: [10.1139/juvs-2014-0012](https://doi.org/10.1139/juvs-2014-0012).
- Barchyn, T.E., Hugenholtz, C.H., Myshak, S., and Bauer, J. 2017. A UAV-based system for detecting natural gas leaks. *J. Unmanned Veh. Syst.* **6**: 18–30.
- Beard, R.W. 2008. Quadrotor dynamics and control. *Brigh. Young Univ.* **19**: 46–56.
- Chabot, D. 2018. Trends in drone research and applications as the Journal of Unmanned Vehicle Systems turns five. *J. Unmanned Veh. Syst.* **6**: vi–xv. doi: [10.1139/juvs-2018-0005](https://doi.org/10.1139/juvs-2018-0005).
- Charron, G., Robichaud-Courteau, T., La Vigne, H., Weintraub, S., Hill, A., Justice, D., et al. 2020. The DeLeaves: A UAV device for efficient tree canopy sampling. *J. Unmanned Veh. Syst.* **8**: 245–264. doi: [10.1139/juvs-2020-0005](https://doi.org/10.1139/juvs-2020-0005).
- Choi, J.Y., Chwa, D., and Kim, M.S. 2000. Adaptive control for feedback-linearized missiles with uncertainties. *IEEE Trans. Aerosp. Electron. Syst.* **36**: 467–481. doi: [10.1109/7.845225](https://doi.org/10.1109/7.845225).
- Freeman, M.R., Kashani, M.M., and Vardanega, P.J. 2021. Aerial robotic technologies for civil engineering: Established and emerging practice. *J. Unmanned Veh. Syst.* **9**(2): 75–91. doi: [10.1139/juvs-2020-0019](https://doi.org/10.1139/juvs-2020-0019).
- Hoffmann, G., Huang, H., Waslander, S., and Tomlin, C. 2007. Quadrotor helicopter flight dynamics and control: Theory and experiment. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, AIAA Publishers, Reston, VA. AIAA-Paper-6461.
- Ioannou, P.A., and Datta, A. 1991. Robust adaptive control: A unified approach. *Proc. IEEE*, **79**: 1736–1768. doi: [10.1109/5.119551](https://doi.org/10.1109/5.119551).
- Jiao, R., Wang, Z., Chu, R., Dong, M., Rong, Y., and Chou, W. 2020. An intuitive end-to-end human-UAV interaction system for field exploration. *Front. Neurobot.* **13**: 117. doi: [10.3389/fnbot.2019.00117](https://doi.org/10.3389/fnbot.2019.00117). PMID: [32116632](https://pubmed.ncbi.nlm.nih.gov/32116632/).
- Jimenez-Cano, A., Heredia, G., and Ollero, A. 2017. Aerial manipulator with a compliant arm for bridge inspection. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 13–16 June 2017, Biscayne Bay, Fla. IEEE. pp. 1217–1222.
- Khalil, H.K., and Grizzle, J.W. 2002. *Nonlinear systems*. Volume 3. Prentice Hall, Upper Saddle River, NJ.
- Khamseh, H.B., Janabi-Sharifi, F., and Abdessameud, A. 2018. Aerial manipulation-a literature survey. *Rob. Auton. Syst.* **107**: 221–235. doi: [10.1016/j.robot.2018.06.012](https://doi.org/10.1016/j.robot.2018.06.012).
- Kim, S., Seo, H., and Kim, H.J. 2015. Operating an unknown drawer using an aerial manipulator. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 25–30 May 2015, Seattle, WA. IEEE. pp. 5503–5508.
- Kutia, J.R., Stol, K.A., and Xu, W. 2018. Aerial manipulator interactions with trees for canopy sampling. *IEEE/ASME Trans. Mechatron.* **23**: 1740–1749. doi: [10.1109/TMECH.2018.2837005](https://doi.org/10.1109/TMECH.2018.2837005).
- Lee, D., Seo, H., Kim, D., and Kim, H.J. 2020. Aerial manipulation using model predictive control for opening a hinged door. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 31 May–31 August 2020, Paris, France. pp. 1237–1242.
- Lippiello, V., and Ruggiero, F. 2012. Cartesian impedance control of a UAV with a robotic arm. *IFAC Proc. Vol.* **45**: 704–709. doi: [10.3182/20120905-3-HR-2030.00158](https://doi.org/10.3182/20120905-3-HR-2030.00158).
- Maki, T., Zhao, M., Shi, F., Okada, K., and Inaba, M. 2020. Model reference adaptive control of multirotor for missions with dynamic change of payloads during flight. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 31 May–31 August 2020, Paris, France. pp. 7433–7439.
- Meng, X., He, Y., and Han, J. 2020. Survey on aerial manipulator: System, modeling, and control. *Robotica*, **38**: 1288–1317. doi: [10.1017/S0263574719001450](https://doi.org/10.1017/S0263574719001450).
- Mohiuddin, A., Tarek, T., Zweiri, Y., and Gan, D. 2020. A survey of single and multi-UAV aerial manipulation. *Unmanned Syst.* **8**: 119–147. doi: [10.1142/S2301385020500089](https://doi.org/10.1142/S2301385020500089).
- Mukherjee, P., and Waslander, S. 2012. Direct adaptive feedback linearization for quadrotor control. In *AIAA Guidance, Navigation, and Control Conference*, 13–16 August 2012, Minneapolis, Minn. p. 4917.

- Orsag, M., Korpela, C., Bogdan, S., and Oh, P. 2014. Valve turning using a dual-arm aerial manipulator. *In* 2014 International Conference on Unmanned Aircraft Systems (ICUAS), 27–30 May 2014, Orlando, Fla. IEEE. pp. 836–841.
- Otto, A., Agatz, N., Campbell, J., Golden, B., and Pesch, E. 2018. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. *Networks*, **72**: 411–458. doi: [10.1002/net.21818](https://doi.org/10.1002/net.21818).
- Pierri, F., Muscio, G., and Caccavale, F. 2018. An adaptive hierarchical control for aerial manipulators. *Robotica*, **36**: 1527–1550. doi: [10.1017/S0263574718000553](https://doi.org/10.1017/S0263574718000553).
- Ruggiero, F., Lippiello, V., and Ollero, A. 2018. Aerial manipulation: A literature review. *IEEE Rob. Autom. Lett.* **3**: 1957–1964. doi: [10.1109/LRA.2018.2808541](https://doi.org/10.1109/LRA.2018.2808541).
- Shojaei, K., Shahri, A.M., and Tarakameh, A. 2011. Adaptive feedback linearizing control of nonholonomic wheeled mobile robots in presence of parametric and nonparametric uncertainties. *Rob. Comput. Integr. Manuf.* **27**: 194–204. doi: [10.1016/j.rcim.2010.07.007](https://doi.org/10.1016/j.rcim.2010.07.007).
- Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. 2010. *Robotics: Modelling, planning and control*. Springer Science & Business Media.
- Slotine, J.J.E., and Li, W. 1991. *Applied nonlinear control*. Vol.199. Prentice Hall, Englewood Cliffs, NJ.
- Sumathy, V., Ghose, D., and Omkar, S. 2018. Analysis of a quadcopter manipulator operating in a workspace above the altitude of the aerial vehicle. *In* 2018 AIAA Guidance, Navigation, and Control Conference, 8–12 January 2018, Kissimmee, Fla. p. 1852.
- Villa, D.K., Brandao, A.S., and Sarcinelli-Filho, M. 2020. A survey on load transportation using multirotor UAVs. *J. Intell. Rob. Syst.* **98**: 267–296. doi: [10.1007/s10846-019-01088-w](https://doi.org/10.1007/s10846-019-01088-w).
- Xilun, D., Pin, G., Kun, X., and Yushu, Y. 2019. A review of aerial manipulation of small-scale rotorcraft unmanned robotic systems. *Chin. J. Aeronaut.* **32**: 200–214. doi: [10.1016/j.cja.2018.05.012](https://doi.org/10.1016/j.cja.2018.05.012).
- Yilmaz, E., Zaki, H., and Unel, M. 2019. Nonlinear adaptive control of an aerial manipulation system. *In* 2019 18th European Control Conference (ECC), 25–28 June 2019, Naples, Italy. IEEE. pp. 3916–3921.