



Practice article

Experimental implementation of a new non-redundant 6-DOF quadrotor manipulation system

Ahmed Khalifa^{a,*}, Mohamed Fanni^{b,1}^a Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Egypt^b Department of Mechatronics and Robotics Engineering, Egypt–Japan University of Science and Technology, Egypt

ARTICLE INFO

Article history:

Received 11 February 2019

Received in revised form 6 May 2020

Accepted 6 May 2020

Available online 11 May 2020

Keywords:

Aerial manipulation

Six-DOF positioning system

Identification

Position holding

PID with gravity compensation

ABSTRACT

The experimental validation of a new quadrotor–manipulator is tackled in this paper. In this system, a two-DOF robotic arm is attached to the bottom center of a quadcopter. The arm is designed with a certain topology such that its end-effector can follow a six-DOF desired trajectory which makes our proposed system superior over the others. It can track an arbitrary six-DOF trajectory in the task space with minimum possible actuators. To test the proposed system feasibility, a quadrotor is selected with high enough payload, then an identification experiment is carried out to estimate its parameters. The mathematical model of the whole system is built. An indoor measurement and state estimation schemes are designed and implemented to get the accurate pose of the platform. A motion control system is designed based on a nonlinear PID technique. A realistic simulation framework is built in MATLAB/SIMULINK. Furthermore, a showcase scenario is carried out by real-time tests. Results show the feasibility and efficiency of the proposed system in achieving the position holding and transferring an object to a certain target pose.

© 2020 ISA. Published by Elsevier Ltd. All rights reserved.

1. Introduction

Recently, Unmanned Aerial Vehicles (UAVs) particularly multi-rotors type receive great attention due to the capability to access regions that are inaccessible to ground vehicles. However, most of the UAV tasks are limited to surveillance and search [1]. Their distinguished mobility nominates them to be utilized for aerial manipulation that opens new application area for robotics such as inspection, maintenance, structure assembly, firefighting, rescue operation, or transportation in locations that are inaccessible, very dangerous or costly to be accessed from the ground.

In the literature, different approaches for quadrotor-based aerial manipulation have been developed by using different mechanisms attached to the quadrotor [2]. The first approach uses a gripper [3–6]. However, in such system, the attitude of the end-effector/gripper is restricted to that of the quadrotor, as a result, the gripper maneuverability is limited to four independent DOF; three translation (e.g., x -, y -, and z -axis) and one rotation (i.e., about z -axis). In [7–10] cables are attached to quadrotor(s) as a second approach. However, with such systems motion of the payload is very difficult and complex to expect/control.

A quadrotor equipped with a robotic manipulator is presented to overcome these restrictions. A quadrotor with a two-DOF manipulator is given in [11–13] but the proposed topology restricts the end-effector motion. This manipulator is designed such that the two joints axes are parallel to one plane axis of the aerial vehicle (e.g., y -axis or x -axis). Hence, the system can do orientation about the other plane axis (e.g., x -axis or y -axis), if and only if the vehicle is commanded to move horizontally.

More complicated aerial manipulators are introduced such as in [14–20]. Although the end-effector of those systems can move in a 6-DOF trajectory, they present two major drawbacks. Firstly, they use a redundant/complex manipulator/quadrotor structure to achieve the 6-DOF tracking capability. Secondly, because of the excessive manipulator/quadrotor weight, the allowable payload capacity, as well as the flight endurance, are reduced.

The authors in [21–24] present a quadrotor-based aerial manipulator in which the robotic arm has two revolute perpendicular joints with one of them is parallel to the quadrotor in-plane axis. Therefore, the arm end-effector can track a six-DOF reference trajectory without the need to move in the horizontal direction.

Achieving the position holding task and ability to track system fast dynamics are the key issues for aerial manipulation. Therefore, the accurate measurements and estimation of the system states must be achieved with low computational cost algorithms. However, the vast majority of the proposed systems [11,15,25–27] utilize a measurement scheme relies upon an exceptionally

* Corresponding author.

E-mail addresses: ahmed.khalifa@el-eng.menofia.edu.eg (A. Khalifa), Mohamed.Fanni@ejst.edu.eg (M. Fanni).¹ On leave from the Department of Production Engineering and Mechanical Design, Mansoura University, Egypt.

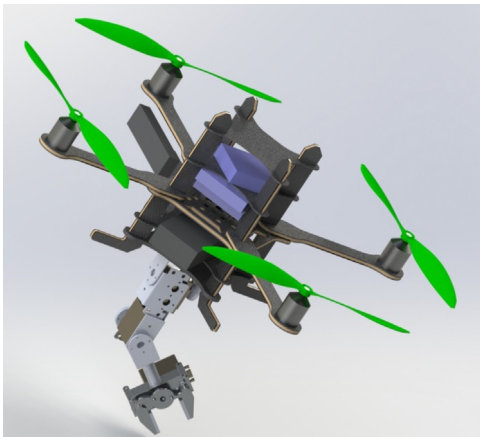


Fig. 1. 3D CAD model of the proposed system.

significant expensive 3D Motion Capture System (e.g., VICON) that confines the tasks of an aerial manipulator to be inside constrained space indoors. However, most real-world applications require to do errands in discretionary locations indoors, i.e., the aerial manipulator has to be able to navigate with installed onboard sensors. To cope up with such an issue, we use onboard sensors and low-cost computation state estimation algorithms. Other works present such a solution require expensive computational cost methods, for instance, [16].

Several quadrotor motion control algorithms are developed in the literature as in [28–30]. By rigidly connecting a dynamic robotic arm to the quadrotor makes the motion control of the entire platform is quite challenging because the quadrotor itself is inherently unstable with fast dynamics and the coupled dynamics are highly nonlinear. Therefore, there is a need to design and implement an efficient and low computational cost control technique to address these issues.

To this point, in this research, the contributions are as follows.

- An aerial manipulator with a unique topology is introduced. It provides solutions to limitations found on the currently developed aerial manipulation systems by having maximum mobility with minimum manipulator structure and weight which in turn allows to decrease the structure weight and hence increase the flight endurance/payload capacity.
- Kinematic and dynamic modeling of the system is presented that facilitates the design, simulation, and control of the system.
- An identification experiment is proposed to find the actuators coefficients.
- Measurement and estimation scheme are implemented to get the vehicle states (i.e., pose and velocity) using onboard sensors (i.e., not limited to in-place motion estimation) taking into consideration the position of the manipulator concerning the sensors and fast system dynamics.
- An efficient and low computational cost control law is designed in Quadrotor/joint space based on PID with the gravity compensation technique and then implemented in simulation and real-time.
- A real-time study is conducted to show system features.

2. Aerial manipulator description

A 3D model representing the proposed structure is given in Fig. 1 while Fig. 2 shows its geometrical frames following the Denavit–Hartenberg convention that is used to study the system kinematics. There are two revolute joints in the arm whose axes

are perpendicular. The first revolute joint axis, z_0 , is parallel to the quadrotor body-frame x -axis, as a result, at the home/extended position, the second joint axis, z_1 , is parallel to the quadrotor body-frame y -axis. Thus, the end-effector can make pitch and roll rotations without the horizontal motion of the aerial vehicle. Consequently, with this proposed configuration, the capability of manipulating objects while following an arbitrary location and orientation can be realized by utilizing the minimum number of actuators/links (i.e., non-redundant system) which in turn allows to decrease the structure weight and hence increase the flight endurance/payload capacity.

2.1. Hardware

An experimental prototype of the proposed aerial manipulator is illustrated in Fig. 3. Fig. 4 shows the proposed experimental implementation of the aerial manipulator.

A Pelican quadrotor [31], which has 650 g payload capacity, is utilized. It has a modular structure that facilitate the mounting of different external components such as onboard computer, sensors, and the manipulator and its avionics. In addition, the quadrotor is equipped by a Flight Control Unit (FCU) that contains an Inertial Measurement Unit (IMU) and two ARM-7 micro-controllers, Low-Level Processor (LLP) and High-Level Processor (HLP). The LLP is in charge of the low-level hardware communication, emergency actions, IMU data fusion (at rate of 1 kHz), and attitude and GPS-based position control. The HLP is used to implement the high-level algorithms. It also has a companion computer (Intel Atom - 1.6 GHz CPU - 1 GB RAM WiFi dongle) dedicated for the higher computational processes.

For precise object handling, we need an accurate position holding which requires a high accuracy state measurement/estimation of the system. The IMU can provide the orientation (by a low-level sensor fusion), angular rates and linear accelerations. To measure the linear position, we cannot rely on the IMU accelerations because of the resulting high integration drift over time. Therefore, we require an additional positioning scheme.

The horizontal position (i.e. x and y) can be measured by using a 2D LIDAR sensor [32] mounted at the top center of the aerial vehicle as shown in Fig. 5. It has a 4 m range, 240 degree field of view, 30 Hz frequency, and 1 mm resolution. The laser measurements are acquired by the onboard computer where the highly computational processing of laser data is carried out.

An Ultrasonic Ranger SRF04 [33] (with 3 cm to 3 m range, 3 cm resolution, and 40 Hz update rate) is used to measure the vertical position, z , see Fig. 5. To avoid the misreading of the sonar due the manipulator motion, we install it downward below one of the rotors. An Arduino board is used to acquire, process sonar data and send the z measurements to the companion computer.

A 200 g lightweight robotic arm is designed such that it has a 200 g payload capacity and a maximum reach of 22 cm to 25 cm. The manipulator components include Three DC motors, two for grasping and joint 2 of type HS-422 (0.4 N m Max torque), one for joint 1 of type HS-5485HB (0.7 N m Max torque), motors driver, wireless R/C joystick to command the motors, and Aluminum links (formed by Aluminum Tubing - 1.50 in., Connector Hub, Servo Bracket, and Long C-Bracket with Ball Bearings). An Arduino board acquires the reference motor motion from the joystick by a wireless receiver, processes them, and sends the motor command to the motors by the motor driver [34].

The main system power source is a 3S 8 Ah Li-Po battery. To power supply the manipulator motors (5 V DC) a step down DC–DC converter is used. As a low-power fail-safe mode, an algorithm is implemented to force the landing as soon as the battery voltage is under 8 V.

The ground station consists of a computer for visualization and peripheral connections, joystick to send high level commands to

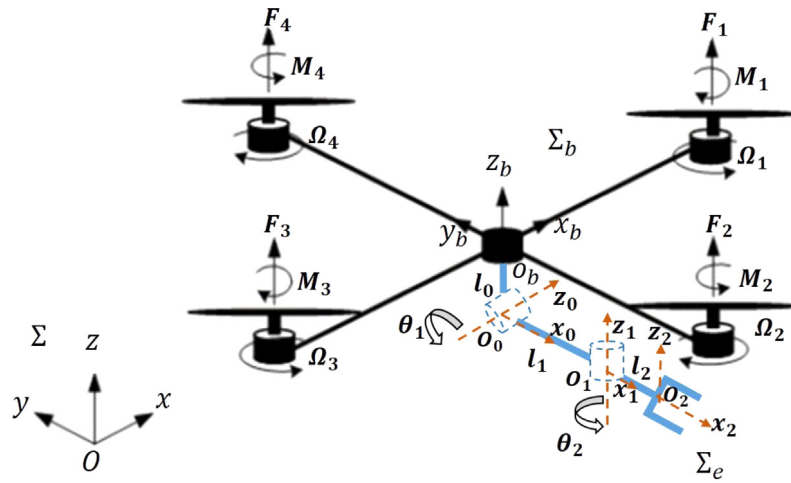


Fig. 2. Geometrical frames of the system.

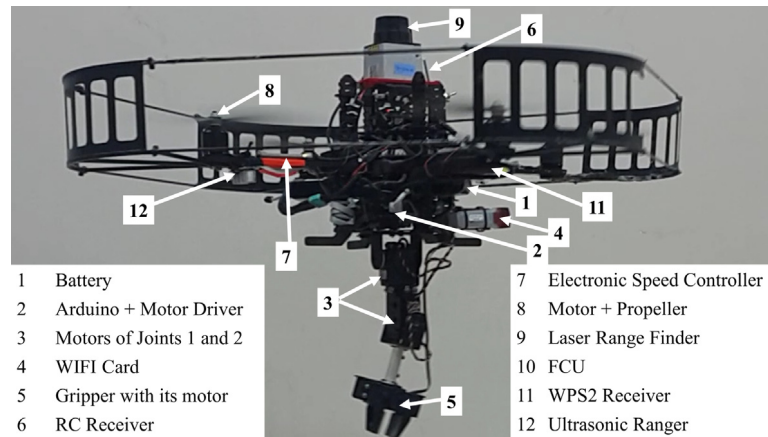


Fig. 3. System prototype.

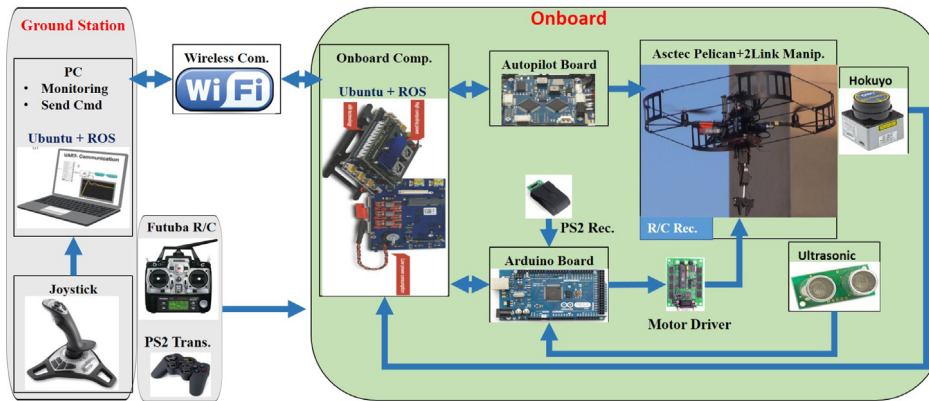


Fig. 4. Functional block diagram of the experimental setup.

onboard computer via WiFi ROS network with the ground computer, and Futaba R/C to send low level command and emergency recovery.

2.2. Software

The companion and ground computers are both have Ubuntu operating system and Robot Operating System (ROS) [35] installed. The computers are connected by WiFi and ROS networks for monitoring and control purposes.

The reason for using ROS is to manage the communication between the multiple different computing systems (ground and Onboard computers, autopilot board particularly the HLP, and Arduino board). Moreover, the ROS has the essential drivers and software for processing the data from the LIDAR sensor.

The autopilot manufacturer provides a Software Development Kit (SDK) used to develop the required algorithms on the HLP that are used to implement the data fusion and high-level position control techniques. We develop a C++-based software in the from

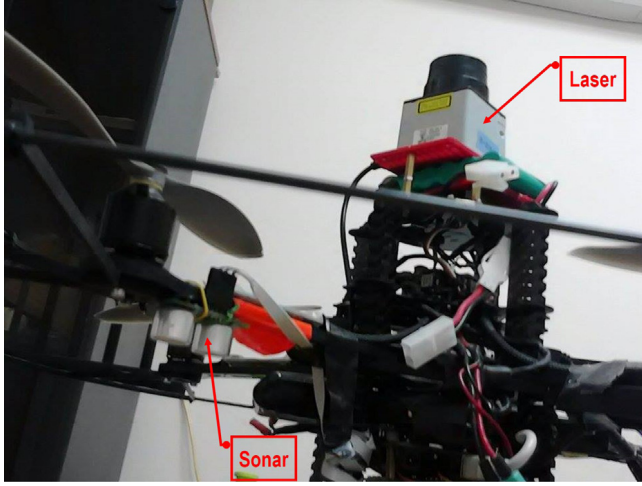


Fig. 5. Sensors setup to get 3D position.

of ROS packages and nodes to oversee and process data between the computing systems.

3. Mathematical modeling

Kinematic and dynamic models of the proposed system have been presented in [21]. In this section, they are reviewed.

3.1. Kinematics

Let us define the vehicle body-fixed reference frame as Σ_b , O_b - x_b y_b z_b with its origin at the center of mass of the body as shown in Fig. 2. Its position with respect to the world-fixed inertial reference frame, Σ , O - x y z , is given by $p_b = [x \ y \ z]^T$, while its orientation is represented by the rotation matrix R_b :

$$R_b = \begin{bmatrix} C_\psi C_\theta & S_\psi C_\theta C_\phi - S_\psi C_\phi & S_\psi S_\phi + C_\psi S_\theta C_\phi \\ S_\psi C_\theta & C_\psi C_\theta C_\phi + S_\psi S_\theta S_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}, \quad (1)$$

where $\Phi_b = [\psi \ \theta \ \phi]^T$ is the yaw-pitch-roll angles. Note that C and S are short notations for $\cos(\cdot)$ and $\sin(\cdot)$ respectively. The end-effector pose is represented by the frame Σ_e , O_e - x_e y_e z_e . Therefore, the position of Σ_e with respect to Σ is given by

$$p_e = p_b + R_b p_{eb}^b, \quad (2)$$

where the vector p_{eb}^b gives the position of Σ_e with respect to and expressed in Σ_b , and it is calculated from fourth column of the total transformation matrix between Σ_e and Σ_b . The orientation of Σ_e can be given by

$$R_e = R_b R_e^b, \quad (3)$$

3.2. Dynamics

Each rotor-assembly j has angular velocity Ω_j and it generates thrust force F_j and drag moment M_j that are given as

$$F_j = K_{f_j} \Omega_j^2, \quad (4)$$

$$M_j = K_{m_j} \Omega_j^2, \quad (5)$$

where K_{f_j} and K_{m_j} are the thrust and drag coefficients.

The dynamic model of the proposed aerial manipulator is given as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau; \quad \tau = Bu, \quad (6)$$

where $q = [x \ y \ z \ \psi \ \theta \ \phi \ \theta_1 \ \theta_2]^T$ is (8×1) vector. M , C , and G represent the symmetric and positive definite inertia matrix,

Table 1

Identified system parameters.

Parameter	Value	Unit	Parameter	Value	Unit
m	1	kg	l_2	$85e-3$	m
d_q	$223.5e-3$	m	m_0	$30e-3$	kg
I_x	$13.215e-3$	N m s ²	m_1	$55e-3$	kg
I_y	$12.522e-3$	N m s ²	m_2	$112e-3$	kg
I_z	$23.527e-3$	N m s ²	I_r	$33.216e-6$	N m s ²
l_0	$30e-3$	m	l_1	$70e-3$	m
K_{F_1}	1.667×10^{-5}	kg m rad ⁻²	K_{F_2}	1.285×10^{-5}	kg m rad ⁻²
K_{F_3}	1.711×10^{-5}	kg m rad ⁻²	K_{F_4}	1.556×10^{-5}	kg m rad ⁻²
K_{M_1}	3.965×10^{-7}	kg m ² rad ⁻²	K_{M_2}	2.847×10^{-7}	kg m ² rad ⁻²
K_{M_3}	4.404×10^{-7}	kg m ² rad ⁻²	K_{M_4}	3.170×10^{-7}	kg m ² rad ⁻²

the Coriolis and centrifugal terms matrix, and the gravity vector of the combined system respectively, τ is (8×1) the input forces/torques, $u = [F_1, F_2, F_3, F_4, T_{m_1}, T_{m_2}]^T$ represents the actuator inputs, $B = HN$ is the input matrix to get the body generalized forces from the actuator inputs. N is given by:

$$N = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ \gamma_1 & -\gamma_2 & \gamma_3 & -\gamma_4 & 0 & 0 \\ -d & 0 & d & 0 & 0 & 0 \\ 0 & -d & 0 & d & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

where $\gamma_j = K_{m_j}/K_{f_j}$, d is the quadrotor center of mass to the rotor rotational axis distance. H is (8×8) matrix used to express the body input forces in Σ and it is given by

$$H = \begin{bmatrix} R_b & O_3 & O_2 \\ O_3 & T_b^T R_b & O_2 \\ O_{2 \times 3} & O_{2 \times 3} & I_2 \end{bmatrix}, \quad (8)$$

and T_b is

$$T_b(\Phi_b) = \begin{bmatrix} 0 & -S_\psi & C_\psi C_\theta \\ 0 & C_\psi & S_\psi C_\theta \\ 1 & 0 & -S_\theta \end{bmatrix}. \quad (9)$$

3.3. System identification

The 3D CAD model of the system is developed using SOLIDWORKS from which we calculate the structural parameters including mass, mass moment of inertia, center of mass, and geometrical parameters.

The next step is to estimate the coefficients of rotor assembly, i.e., K_{f_j} and K_{m_j} , by using the proposed experimental setup illustrated in Fig. 6.

In this experiment, the rotor assembly is mounted on a six-DOF force/torque sensing device which sends data to a computer via Data Acquisition Card from NI (NIDAC). A SIMULINK program is developed as an interface to read and process data acquired by NIDAC. We increase the rotor velocity gradually, and for each value, the resulted thrust and drag moment are measured. The collected input (rotor velocity)/output (thrust or drag) data are fed to a curve fitting algorithm developed by using MATLAB Curve Fitting toolbox. The acquired data of thrust and moment are fitted to be in the form of (4) and (5) so that we can get the thrust and drag coefficients. The relation between thrust force/drag moment and the squared rotor speed is given in Figs. 7a/7b. The resulted curve fitting Root Mean Squared Error (RMSE) is calculated to quantify the fitting efficiency and we find it equal to 0.3378 and 0.003742 for the thrust and drag relations respectively which is a satisfactory results. The identified parameters are collected and given in Table 1.

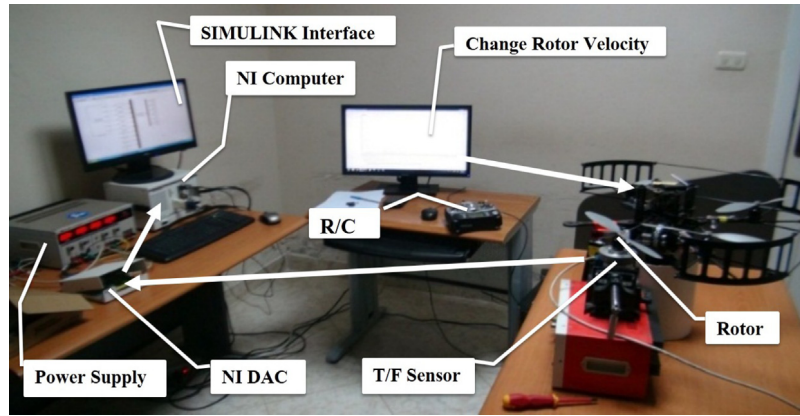
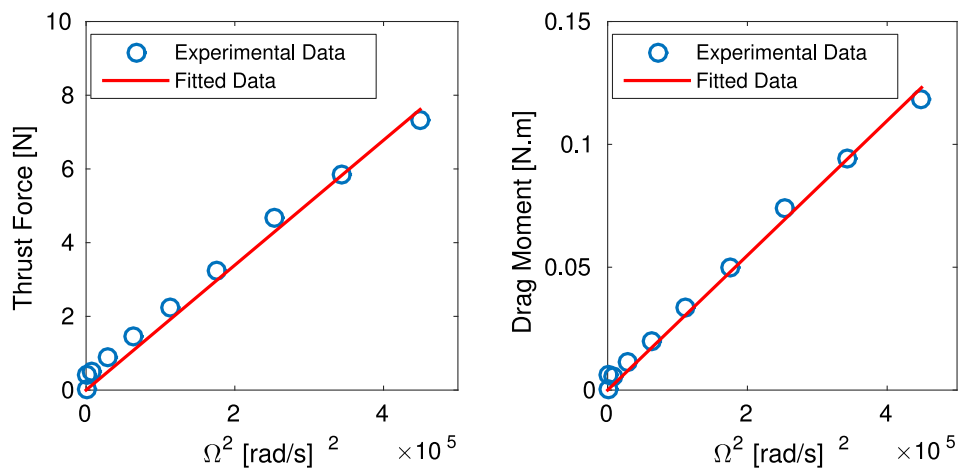


Fig. 6. The proposed identification setup to get the rotor assembly coefficients.



(a) Thrust force V.S. Rotor speed squared (b) Drag moment V.S. Rotor speed squared

Fig. 7. Relation between the generated thrust force/drag moment and rotor speed squared of the quadrotor.

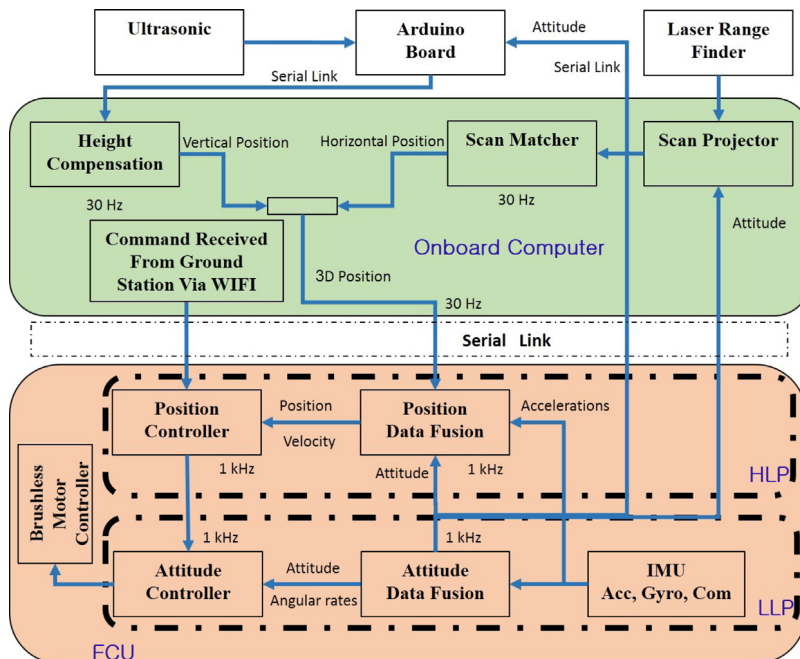


Fig. 8. Block diagram representing the state estimation scheme.

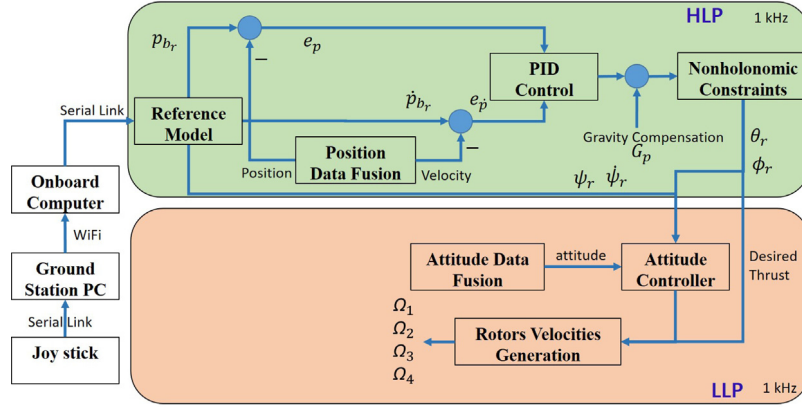


Fig. 9. Block diagram of the control structure.

4. Linear position and velocity estimation

Achieving an accurate position holding, which requires precise system states measurement/estimation, is a prerequisite for aerial manipulation. In this section, we present the proposed state measurement/estimation scheme see Fig. 8.

The quadrotor orientation and angular rates (i.e., body angular position and velocity) can be obtained from the LLP at 1 kHz rate directly. The range sensors gives the 3D linear position measurements but at rate of about 30 Hz. On the other hand, the quadrotor motion dynamics have high bandwidth, it can reach to 1 kHz. Therefore, this linear position information is fused with the body accelerations \ddot{p}_b^b given by the IMU at 1 kHz. This fusion algorithm is implemented in the HLP and its output (system states) is sent to a position controller. Both the fusion filter and the position controller are implemented on the HLP at 1 kHz.

The horizontal position can be calculated by using the technique presented in [36] from LIDAR. By using the ultrasonic sensor, we can get the distance between the sensor position and the ground, s_z . The body vertical height, z , is calculated by compensating against the quadrotor roll and pitch by

$$z = s_z \cos(\phi) \cos(\theta). \quad (10)$$

The rest of this section is dedicated to design the fusion algorithm/filter. The fusing filter is designed and decoupled for three axes x, y, z in the world frame. The body-fixed accelerations, \ddot{p}_b^b , are rotated in the global frame by the rotational matrix, R_b as

$$\ddot{p}_b = R_b \ddot{p}_b^b - [0 \ 0 \ g_r]^T, \quad (11)$$

where g_r is the gravity acceleration.

We use a Luenberger observer [37] to design this filter. For simplicity and readability, the filter for motion in x -axis is presented and similarly we can design for the other axes (i.e., y and z). Let us define the filter states as the position, $p_{b,x}$, speed, $\dot{p}_{b,x}$, and the acceleration sensor bias, b_x , the filter input as acceleration expressed in the world frame, the measurement as the position sensors readings, $p_{s,x}$, and the output as linear position and velocity.

If one represent the filter state variable as $X_{obs} = [p_{b,x} \ \dot{p}_{b,x} \ b_x]^T$, $U_{obs} = \ddot{p}_x$, and $Y_{obs} = p_{s,x}$. Then, the observer state equations are given by

$$\begin{aligned} \dot{\hat{X}}_{obs} &= A_{obs} \hat{X}_{obs} + L_{obs}(Y - \hat{Y}) + B_{obs} U_{obs}, \\ Y &= C_{obs} \hat{X}_{obs}, \end{aligned} \quad (12)$$

where $A_{obs} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$, $B_{obs} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$, $C_{obs} = [1 \ 0 \ 0]$, and $L_{obs} = \begin{bmatrix} L_{obs1} \\ L_{obs2} \\ L_{obs3} \end{bmatrix}$.

5. Control design

The proposed real time control system is illustrated in Fig. 9 shows. The Joystick is used to send the desired pose for the quadrotor–manipulator system via a ROS-based network between ground station and onboard computer.

As under-actuated system, our aerial vehicle, quadrotor, has 4 control inputs/actuators but it moves six-DOF space, so 6 variables have to be controlled. As a result, the linear position, p_b , and yaw angle are used as the controlled variables. The vertical position and yaw rotation is controlled directly while the horizontal position controller has a cascade structure where the pitching and rolling that are treated as in-between control inputs. The LLP implements an inner loop to control vehicle attitude. The rolling and pitch references are provided from the outer loop position controller. A nonlinear PID position controller based on gravity compensation is used as the outer loop that it is carried out in the HLP.

$$\begin{aligned} \tau_p &= [\tau_x \ \tau_y \ \tau_z]^T = K_p(p_{br} - p_b) + K_d(\dot{p}_{br} - \dot{p}_b) + \\ &K_i \int_0^t (p_{br} - p_b) dt + G_p, \end{aligned} \quad (13)$$

where $G_p = G(1 : 3)$ is the translational part of gravity term given in the dynamic equation (6). K_p , K_d , and K_i are the controller parameters. p_{br} and \dot{p}_{br} are the desired position and velocity respectively.

The sent command from joystick, p_{bc} , is a step input so it does not have a second-time derivative. Therefore, to generate smooth reference trajectories, p_{br} , we propose the following technique

$$\dot{p}_{br} = \omega_n^2 (p_{bc} - p_{br}) - 2\zeta \omega_n \dot{p}_{br}. \quad (14)$$

The characteristics of the generated trajectories is tuned by the natural frequency ω_n and damping ζ . We choose a damping of 1 and the natural frequency of 2.5 to get a critically damped response and satisfactory settling time.

The reference values of pitching, θ_r , and rolling, ϕ_r , are calculated from the position controller output by the relation

$$\begin{bmatrix} \theta_r \\ \phi_r \end{bmatrix} = \frac{1}{\tau_p(3)} \begin{bmatrix} C_\psi & S_\psi \\ S_\psi & -C_\psi \end{bmatrix} \begin{bmatrix} \tau_p(1) \\ \tau_p(2) \end{bmatrix}, \quad (15)$$

which is conducted from (1) considering small roll and pitch angles approximation.

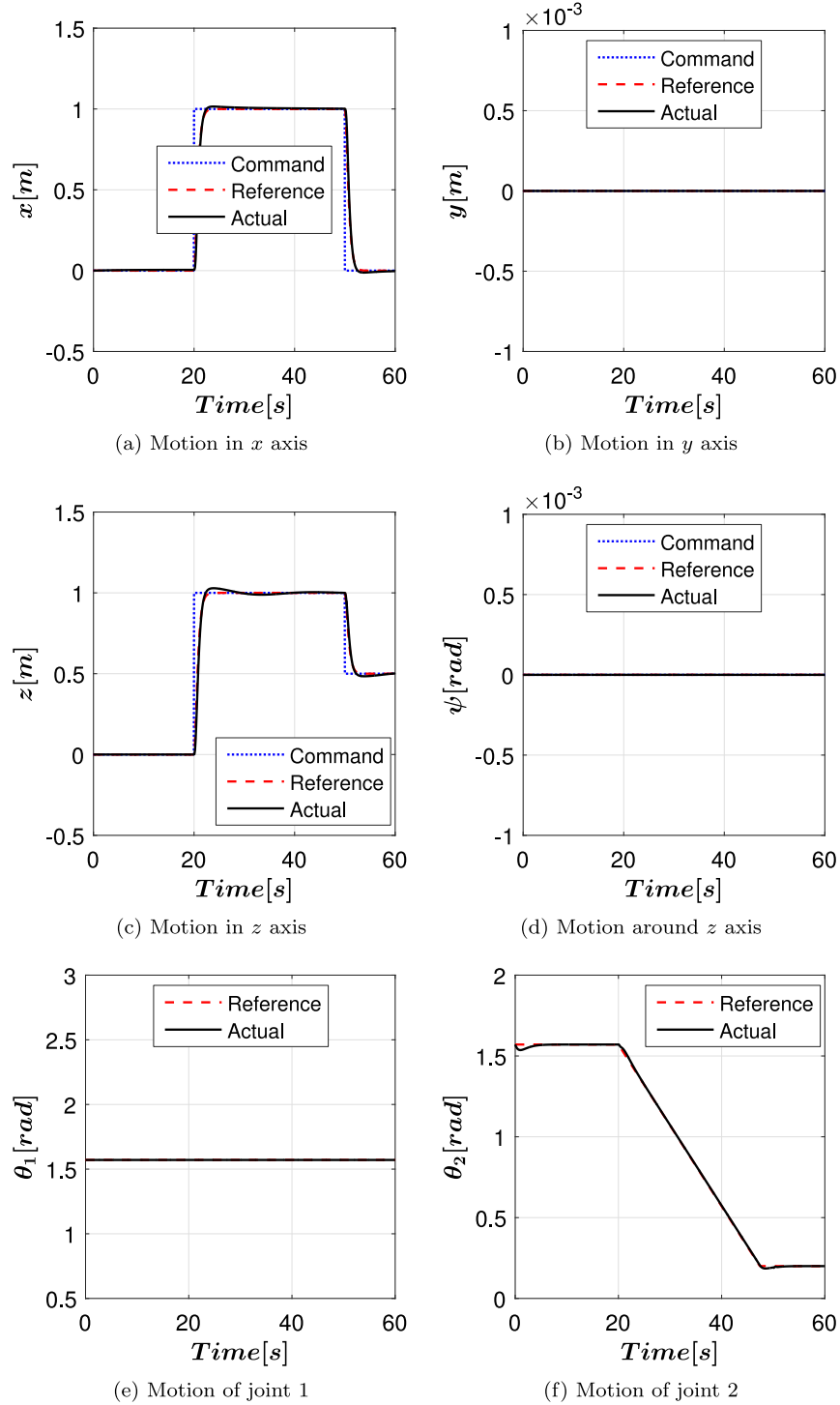


Fig. 10. Simulation study: actual and reference trajectories of the system pose.

The robotic arm joint space position control is achieved by the built-in a PID servo controller. This is feasible due to the simplicity of the manipulator dynamics as a result of the low-speed and lightweight arm parts.

6. Simulation results

The system kinematic dynamic equations of motion, as well as the proposed control algorithm, are carried out in MATLAB/SIMULINK. For quite realistic simulation, we add to the system states a normally distributed measurement noise of 10^{-3} mean

and 5×10^{-3} . Besides, we use the parameter estimated by the experimental identification. The controller parameters, given in Table 2, are fine-tuned to the required performance. Fig. 10, show the simulation results of the system position, orientation, and joints angles. The desired destination point is considered to be in the x -axis direction, and a 50 g object is placed in the end-effector gripper. First, to make a vertical take-off, we send a positive position command in the z -axis til the quadrotor reaches at a required height as shown in Fig. 10c. To move towards the desired position, to mount the object, a command is sent in the x -axis, see Fig. 10a. At the same time, to prepare the arm for the object releasing, an

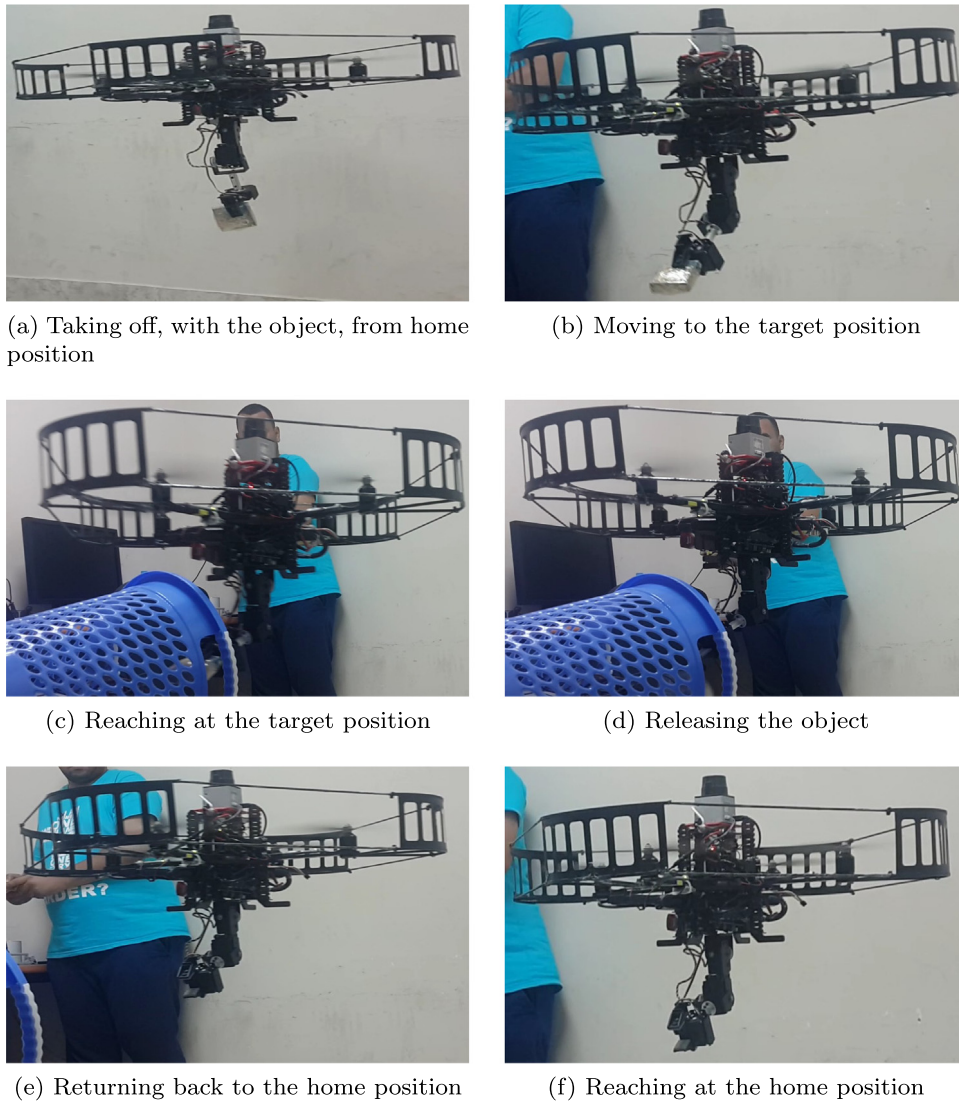


Fig. 11. Real-time demonstration.

Table 2
Simulation study: controller parameters.

Parameter	Value	Parameter	Value
$K_{p_{x,y}}$	2	$K_{d_{x,y}}$	7
$K_{i_{x,y}}$	0.5	K_{p_z}	10
K_{d_z}	10	K_{i_z}	5

Table 3
Experimental study: controller parameters.

Par.	Value	Par.	Value
$K_{p_{x,y}}$	5	$K_{d_{x,y}}$	12
$K_{i_{x,y}}$	2	K_{p_z}	15
K_{d_z}	15	K_{i_z}	4
$L_{obs_{x,y}}$	$[18.01 \ 45.18 \ 0.45]^T$	L_{obs_z}	$[18.01 \ 45.18 \ 0.45]^T$

increasing ramp reference is sent to the motor of joint 2, θ_2 , see Fig. 10f. When the end-effector is at the desired position, we send a z command to make the gripper near to the destination point, next transmit an order to the gripper to release the object. After object placing, an order is sent in the x-axis to return to the initial point again. These simulation results enlighten the feasibility and efficacy of the proposed aerial manipulator. In the next section, we verify these results experimentally.

7. Experimental results

The experimental setup, state measurement and estimation algorithm, and control technique introduced in the previous sections are tested in real-time as the following scenario. Our objective is to transfer a target object from an initial point to another destination point in a teleoperation mode. Before starting the

test, a 50 g object is attached to the gripper. Table 3 gives the control parameters of this experimental test. The experiment is implemented over six phases as illustrated in Fig. 11: taking off, motion towards the destination, reaching the target position, position holding to release the object, motion to the home position, and finally, reaching the home point.

The Joystick is used to send the desired pose for the quadrotor–manipulator system via a ROS-based network between the ground station and onboard computer. The estimated sensor bias is presented in Fig. 13. Unfortunately, accurate ground truth is not available to test the estimation accuracy. Therefore, the plots are evaluated qualitatively. It is noticed that the filter converges after approximately 1 s.

Fig. 12 presents the real-time results. The wireless joystick is used to send desired angles to the motors driver. As shown in Fig.

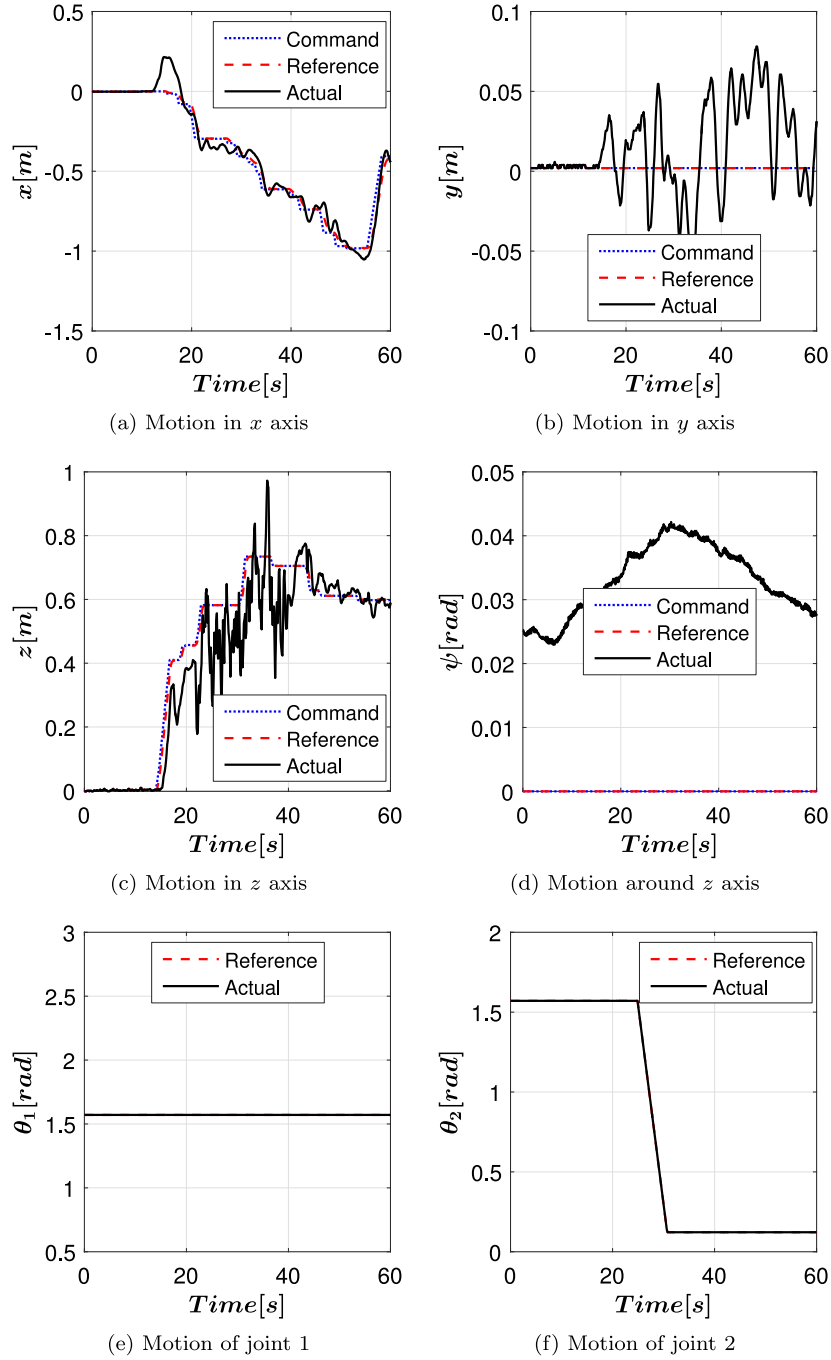


Fig. 12. Experimental study: system position, orientation, and joints angles.

12c, to start the task, a z position reference is sent for taking off during which a small drift appears in the x , y , z , and ψ directions that are recovered soon due to the robustness of position holding. Then, the vehicle is commanded to move in x direction towards the destination place, see Fig. 12a. As soon as we reach the target position, we activate the position holding mode. At this point, an order is sent to the grasping mechanism to open for releasing the object. After that, the vehicle goes back to the initial point. Finally, a z command is sent for landing on the base station.

As the task requires, there is no command for motion ψ direction, see Fig. 12d. Since, the required task is simple, only a ramp reference is sent to θ_2 motor for facilitating the object

mounting task, and we send command such that we keep θ_1 at its initial value $\pi/2$, see Figs. 12e and 12f.

To quantify the controller performance, we calculate the Root Mean Squared (RMS) error of the variables as given in Table 4. We notice that the RMS error in the vertical direction, z , is higher than that in the x direction. This is due to the sonar's lower resolution and its sensitivity to system vibrations.

These results show the reliability and a satisfactory proficiency of the proposed aerial manipulator, the state estimation scheme, and the control algorithm. However, they show that the position control algorithm requires improvements in future work. Besides, the sonar has to be replaced by a more accurate and reliable position sensor.

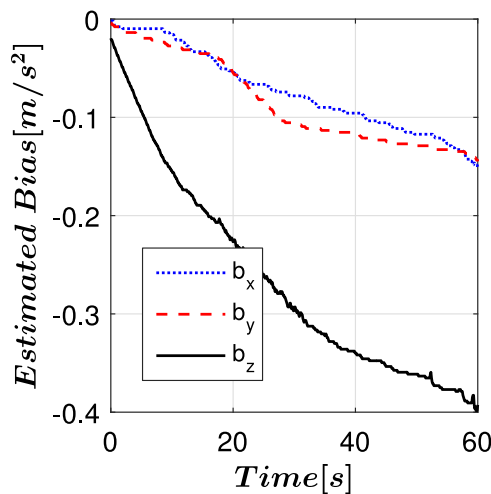


Fig. 13. Experimental study: sensor bias.

Table 4
Experimental study: RMS error.

Type	RMS error	Maximum command
x [m]	0.0595	1
y [m]	0.0297	0
z [m]	0.0944	1
ψ [rad]	0.0335	0

8. Conclusion

In this research, a new quadrotor-based aerial manipulator is successfully designed and implemented. It has superior features over the currently developed systems. Its construction, modeling, and experimental setup are introduced. Experimental parameter identification is conducted. A scheme for system states measurement and estimation is proposed, which is satisfactory efficient, using onboard sensors and low computational cost algorithms. A light nonlinear PID position control algorithm is designed and implemented. The reliability and efficiency of the proposed system are demonstrated by simulation and experimental studies with object pick/place scenario. As future work, higher reliability and accuracy vertical position sensor and more robust position controller have to be tested.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] Wang H, Li Z, Xiong H, Nian X. Robust h_∞ attitude tracking control of a quadrotor uav on so (3) via variation-based linearization and interval matrix approach. *ISA Trans* 2019;87:10–6.
- [2] Ruggiero F, Lippiello V, Ollero A. Aerial manipulation: A literature review. *IEEE Robot Autom Lett* 2018;3(3):1957–64.
- [3] Mellinger D, Lindsey Q, Shomin M, Kumar V. Design, modeling, estimation and control for aerial grasping and manipulation. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE; 2011, p. 2668–73.
- [4] Braithwaite A, Alhinai T, Haas-Heger M, McFarlane E, Kovač M. Tensile web construction and perching with nano aerial vehicles. In: *Robotics research*. Springer; 2018, p. 71–88.
- [5] Lindsey Q, Mellinger D, Kumar V. Construction with quadrotor teams. *Auton Robots* 2012;33(3):323–36.

- [6] Willmann J, Augugliaro F, Cadalbert T, D'Andrea R, Gramazio F, Kohler M. Aerial robotic construction towards a new field of architectural research. *Int J Archit Comput* 2012;10(3):439–60.
- [7] Goodarzi FA, Lee D, Lee T. Geometric control of a quadrotor uav transporting a payload connected via flexible cable. *Int J Control Autom Syst* 2015;13(6):1486–98.
- [8] Guerrero-Sánchez ME, Mercado-Ravell DA, Lozano R, García-Beltrán CD. Swing-attenuation for a quadrotor transporting a cable-suspended payload. *ISA Trans* 2017;68:433–49.
- [9] Sarkisov YS, Kim MJ, Bicego D, Tsetserukou D, Ott C, Franchi A, Kondak K. Development of sam: cable-suspended aerial manipulator. 2019, arXiv preprint arXiv:1903.02426.
- [10] Goodarzi FA, Lee T. Dynamics and control of quadrotor uavs transporting a rigid body connected via flexible cables. In: *2015 American control conference (ACC)*. IEEE; 2015, p. 4677–82.
- [11] Kim S, Choi S, Kim HJ. Aerial manipulation using a quadrotor with a two dof robotic arm. In: *Intelligent robots and systems (IROS), 2013 IEEE/RSJ international conference on*. IEEE; 2013, p. 4990–5.
- [12] Abaunza H, Castillo P, Victorino A, Lozano R. Dual quaternion modeling and control of a quad-rotor aerial manipulator. *J Intell Robot Syst* 2017;88(2–4):267–83.
- [13] Yilmaz E, Zaki H, Unel M. Nonlinear adaptive control of an aerial manipulation system. In: *2019 18th European control conference (ECC)*. 2019, p. 3916–21. <http://dx.doi.org/10.23919/ECC.2019.8795709>.
- [14] Korpela C, Orsag M, Oh P. Towards valve turning using a dual-arm aerial manipulator. In: *Intelligent robots and systems (IROS 2014), 2014 IEEE/RSJ international conference on*. IEEE; 2014, p. 3411–6.
- [15] Yüksel B, Staub N, Franchi A. Aerial robots with rigid/elastic-joint arms: Single-joint controllability study and preliminary experiments. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE; 2016, p. 1667–72.
- [16] Ollero A, Heredia G, Franchi A, Antonelli G, Kondak K, Sanfeliu A, Viguria A, Martinez-de Dios JR, Pierri F, Cortés J, et al. The aeroarms project: Aerial robots with advanced manipulation capabilities for inspection and maintenance. *IEEE Robot Autom Mag* 2018;25(4):12–23.
- [17] Lippiello V, Ruggiero F. Cartesian impedance control of a uav with a robotic arm. In: *10th international IFAC symposium on robot control*. 2012, p. 704–9.
- [18] Lippiello V, Ruggiero F. Exploiting redundancy in cartesian impedance control of uavs equipped with a robotic arm. In: *IEEE/RSJ international conference on intelligent robots and systems*. IEEE; 2012, p. 3768–73.
- [19] Orsag M, Korpela C, Oh P. Modeling and control of mm-uav: Mobile manipulating unmanned aerial vehicle. *J Intell Robot Syst* 2013;69(1–4):227–40.
- [20] Heredia G, Jimenez-Cano A, Sanchez I, Llorente D, Vega V, Braga J, Acosta J, Ollero A. Control of a multirotor outdoor aerial manipulator. In: *IEEE/RSJ international conference on intelligent robots and systems*. IEEE; 2014, p. 3417–22.
- [21] Khalifa A, Fanni M, Ramadan A, Abo-Ismael A. Modeling and control of a new quadrotor manipulation system. In: *2012 IEEE/RAS international conference on innovative engineering systems*. IEEE; 2012, p. 109–14.
- [22] Fanni M, Khalifa A. A new 6-dof quadrotor manipulation system: Design, kinematics, dynamics, and control. *IEEE/ASME Trans Mechatronics* 2017;22(3):1315–26.
- [23] Khalifa A, Fanni M. A new quadrotor manipulation system: Modeling and point-to-point task space control. *Int J Control Autom Syst* 2017;15(3):1434–46.
- [24] Khalifa A, Fanni M, Ramadan A, Abo-Ismael A. Controller design of a new quadrotor manipulation system based on robust internal-loop compensator. In: *2015 IEEE international conference on autonomous robot systems and competitions*. IEEE; 2015, p. 97–102.
- [25] Choi Y-C, Ahn H-S. Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests. *IEEE/ASME Trans Mechatronics* 2014;20(3):1179–92.
- [26] Ruggiero F, Trujillo MA, Cano R, Ascorbe H, Viguria A, Pérez C, Lippiello V, Ollero A, Siciliano B. A multilayer control for multirotor uavs equipped with a servo robot arm. In: *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE; 2015, p. 4014–20.
- [27] Baizid K, Giglio G, Pierri F, Trujillo MA, Antonelli G, Caccavale F, Viguria A, Chiaverini S, Ollero A. Experiments on behavioral coordinated control of an unmanned aerial vehicle manipulator system. In: *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE; 2015, p. 4680–5.
- [28] Wang N, Deng Q, Xie G, Pan X. Hybrid finite-time trajectory tracking control of a quadrotor. *ISA Trans* 2019.
- [29] Cai W, She J, Wu M, Ohyama Y. Disturbance suppression for quadrotor uav using sliding-mode-observer-based equivalent-input-disturbance approach. *ISA Trans* 2019.
- [30] Miranda-Colorado R, Aguilar LT. Robust pid control of quadrotors with power reduction analysis. *ISA Trans* 2019.
- [31] Asctec Pelican Quadrotor. 2014, Available at <http://www.asctec.de/en/uav-uas-drone-products/asctec-pelican/>.

- [32] Hokuyo URG-04LX laser range finder. 2014, Available at https://www.hokuyo-aut.jp/02sensor/07scanner/urg_04lx.html/.
- [33] Devantech SRF04 ultrasonic range finder. 2014, Available at <https://acroname.com/products/DEVANTECH-SRF05-SONAR-RANGING-MODULE>.
- [34] LYNXMOTION. 2014, Available at <http://www.lynxmotion.com/default.aspx>.
- [35] Robot Operation System. 2014, Available at <http://wiki.ros.org/>.
- [36] Dryanovski I, Valenti RG, Xiao J. An open-source navigation system for micro aerial vehicles. *Auton Robots* 2013;34(3):177–88.
- [37] Achtelik M, Achtelik M, Weiss S, Siegwart R. Onboard imu and monocular vision based control for mavs in unknown in-and outdoor environments. In: *IEEE international conference on robotics and automation (ICRA)*. IEEE; 2011, p. 3056–63.