J Intell Robot Syst
DOI 10.1007/s10846-012-9699-0

Micro-Helicopter for Long-Distance Missions: Description and Attitude Stabilization

Eduardo Steed Espinoza · Octavio Garcia · Guillaume Sanahuja · Alejandro Malo · Rogelio Lozano

Received: 21 May 2012 / Accepted: 10 July 2012

Abstract This paper presents the development of a micro coaxial helicopter (*MCR* UAV) whose main characteristic is that it should be carried by an air shuttle transporter and then released in a desired place far away from the launching site, to develop surveillance missions in hover flight. A real-time embedded system is built in order to validate the proposed aerodynamic prototype, and a classic control law based on a classical backstepping procedure for the dynamic system is implemented to test this vehicle in autonomous flight. Finally, simulation and practical results are presented for hover flight.

Keywords Coaxial helicopter · Wind analysis · Hover flight · Backstepping control · Real-time embedded system

E. S. Espinoza (✉) · A. Malo · R. Lozano
Laboratoire Franco-Mexicain d'Informatique et Automatique, LAFMIA UMI 3175 CNRS-CINVESTAV, Mexico, Mexico
e-mail: eespinoza@ctrl.cinvestav.mx

A. Malo
e-mail: alexmalo@ctrl.cinvestav.mx

R. Lozano
e-mail: rlozano@ctrl.cinvestav.mx

E. S. Espinoza
Universidad Politecnica de Pachuca, Hidalgo, Mexico
e-mail: steed@upp.edu.mx

O. Garcia
The Laboratory of Non-inertial Robots and Man-machine Interfaces, CINVESTAV Monterrey, Nuevo Leon, Mexico
e-mail: ogarcias@cinvestav.mx

G. Sanahuja · R. Lozano
Laboratoire Heudiasyc, Université de Technologie de Compiègne, UMR CNRS 6599, France

G. Sanahuja
e-mail: gsanahuj@hds.utc.fr

R. Lozano
e-mail: rlozano@hds.utc.fr

1 Introduction

The applications of micro UAVs (Unmanned Aerial Vehicles) are growing constantly due to the scientific-technologic challenge. Researchers from the international community to control, robotics, aerospace and control, among others tend to focus their scientific research in this area. The applications of UAVs are based on not only military actions, the civilian applications begin to take more importance. Some civilian applications can be cited as examples: the monitoring of traffic on highways, support in search and rescue, intervention in hostile environments, detection of fire in forests.These applications require an UAV that is

Published online: 10 August 2012

able to evolve and adapt to the environment in which it is operating.

Coaxial helicopters require mechanisms such as swashplates, stabilizer bars, and tilt-rotors in order to control the direction of rotor thrust vector. In fact, some theoretical and practical contributions about coaxial helicopters have been reported in the literature. A robust control for a coaxial micro helicopter was presented in [1]. Bouabdallah et al. [2] discusses the design and control of an indoor coaxial helicopter. A simplified model and backstepping control for a coaxial helicopter can be found in [3].

Our *Micro Coaxial Rocket-Helicopter* (*MCR* UAV) differs from those conventional-coaxial helicopters because it possesses control surfaces (ailerons) to control the attitude flight, and employs the air produced by the coaxial propellers (Prop-wash) over the control surfaces to maintain the vertical position. The main objective of the (*MCR* UAV) is to be catapulted and then hover at a long distance from the launching site, see Fig. 1, with the purpose of reducing the energy consumption, increasing the range, and developing surveillance missions.

On the other hand, several control techniques to stabilize UAVs have been published in the literature. In [4], the authors present the trajectory tracking control design for autonomous helicopters using a backstepping algorithm. The modeling and decoupling control of the commercial coaxial helicopter is presented in [5]. However, the control problem of our *MCR* UAV in hover flight is solved by proposing a complete backstepping procedure considering the reduced nonlinear system.

The main contribution of this paper is to present the modeling, the attitude control and the real-time embedded system of the *Micro Coaxial Rocket-Helicopter* (*MCR* UAV). This prototype is developed at LAFMIA CNRS-CINVESTAV Mexico in collaboration with the *Laboratoire Heudiasyc* (Université de Technologie de Compiègne, France). The paper is organized as follows:

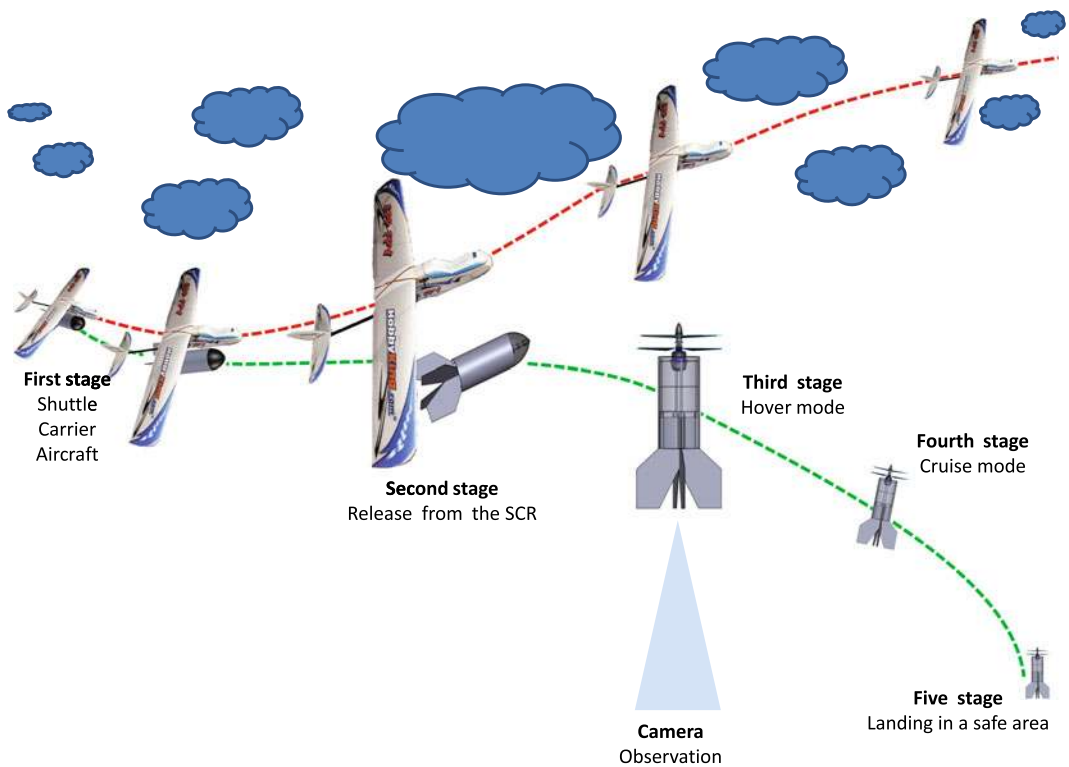

Fig. 1 Objective of the project

Section 2 introduces a functional description and the dynamic equations of the vehicle. In Section 3, the control technique based on a backstepping procedure, and the stability analysis are presented. Simulation results of the closed-loop system and experimental results in hover flight are shown in Sections 4 and 5 respectively. Finally, Section 6 gives the conclusions and future works of this project.

2 Micro Coaxial Helicopter

In this section, a complete functional description of the vehicle is given, and the dynamic model is obtained using the classical Newton–Euler equations.

2.1 Description

The micro coaxial rotorcraft-helicopter (*MCR* UAV) is based on a couple of counter rotating brushless motors, and the main characteristic of this vehicle is that it is capable of being launched through an air shuttle transporter, and transforming itself into a coaxial helicopter at a long distance from the launching site. Once the vehicle reaches the objective (a place, a building, an uninhabited area, etc.), it performs hover flight, and can inspect the environment acquiring and transmitting information through a RF camera to a ground station. Concerning the functional description, the *MCR* vehicle possesses aerodynamic control surfaces (ailerons) which are used to control the roll and pitch motion while the difference of the velocities of the two motors regulates the yaw motion.

2.2 Dynamics

Consider an inertial fixed frame and a body frame fixed attached to the center of gravity of the helicopter denoted by $\mathcal{I} = \{x_\mathcal{I}, y_\mathcal{I}, z_\mathcal{I}\}$ and $\mathcal{B} = \{x_\mathcal{B}, y_\mathcal{B}, z_\mathcal{B}\}$, respectively, see Fig. 2, [6].

Assume the generalized coordinates of the micro UAV as $\boldsymbol{q} = (x, y, z, \psi, \theta, \phi)^T \in \mathbb{R}^6$, where $\boldsymbol{\xi} = (x, y, z)^T \in \mathbb{R}^3$ represents the translation coordinates relative to the inertial frame, and $\boldsymbol{\eta} = (\psi, \theta, \phi)^T \in \mathbb{R}^3$ describes the vector of three Euler angles with rotations around z, y, x axis. These angles ψ, θ, and ϕ are called yaw, pitch and roll, respectively. Assume the translational velocity and the angular velocity in the body frame as $\boldsymbol{v} = (u, v, w)^T \in \mathbb{R}^3$ and $\boldsymbol{\Omega} = (p, q, r)^T \in \mathbb{R}^3$, respectively.

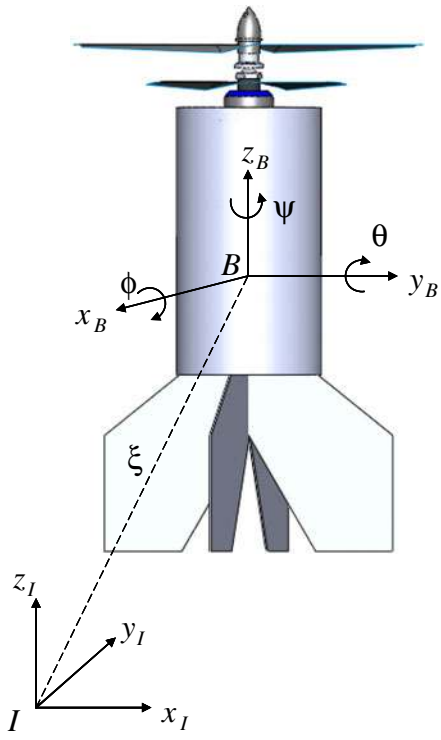

Fig. 2 *MCR* UAV

The Newton–Euler equations of motion for a rigid object provide the dynamic model for this micro coaxial helicopter. This expression is described as [7]

$$\dot{\boldsymbol{\xi}} = \boldsymbol{V} \tag{1}$$

$$m\dot{\boldsymbol{V}} = \boldsymbol{RF} \tag{2}$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\hat{\boldsymbol{\Omega}} \tag{3}$$

$$\boldsymbol{I}\dot{\boldsymbol{\Omega}} = -\boldsymbol{\Omega} \times \boldsymbol{I}\boldsymbol{\Omega} + \boldsymbol{\Gamma} \tag{4}$$

where $\boldsymbol{F} \in \mathbb{R}^3$ and $\boldsymbol{\Gamma} \in \mathbb{R}^3$ are the total force and torque acting on the vehicle, respectively. $\boldsymbol{V} = (\dot{x}, \dot{y}, \dot{z})^T \in \mathbb{R}^3$ is the translational velocity in the inertial frame, $m \in \mathbb{R}$ denotes the mass of the *MCR* UAV, $\boldsymbol{I} \in \mathbb{R}^{3\times3}$ contains the moments of inertia of the micro helicopter, and $\hat{\boldsymbol{\Omega}}$ is a skew-symmetric matrix such that $\hat{\boldsymbol{\Omega}}\boldsymbol{a} = \boldsymbol{\Omega} \times \boldsymbol{a}$. Thus, $\boldsymbol{R}$

represents the transformation matrix from the body frame to the inertial frame

$$\boldsymbol{R} = \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}$$

where the shorthand notation of $s_a = sin(a)$ and $c_a = cos(a)$ is used. For this matrix, the order of the rotations is considered as yaw, pitch and roll (ψ, θ, ϕ) [8].

2.2.1 Forces

The forces that act on the vehicle are given as follows

Propulsion Forces The thrust force is generated by two motors and is described as

$$\boldsymbol{F}_p = \begin{pmatrix} 0 \\ 0 \\ T_c \end{pmatrix}$$

where T_c is the thrust force of the two motors ($T_c = T_1 + T_2$). In this analysis, the thrust force is oriented parallel to the axis $z_{\mathcal{B}}$ of the body frame, see Fig. 3.

Aerodynamic Forces The aerodynamic forces in the body frame are written as

$$\boldsymbol{F}_a = \boldsymbol{B}^T \boldsymbol{W}^T \begin{pmatrix} -L \\ Y \\ -D \end{pmatrix}$$

where L, Y, and D are the aerodynamic forces: lift, side force, and drag, respectively. $\boldsymbol{B}$ and $\boldsymbol{W}$ are rotation matrices that represent the transformation the aerodynamic forces from the body frame to aerodynamic frame (stability and wind frames).

$$\boldsymbol{B} = \begin{pmatrix} c_\alpha & 0 & s_\alpha \\ 0 & 1 & 0 \\ -s_\alpha & 0 & c_\alpha \end{pmatrix}, \quad \boldsymbol{W} = \begin{pmatrix} c_\beta & s_\beta & 0 \\ -s_\beta & c_\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

where α is the angle of attack and β is the sideslip angle.

Gravitational Force The force due to the weight of the vehicle is described as

$$\boldsymbol{F}_w = \boldsymbol{R}^T \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix}$$

where g is the acceleration due to gravity.

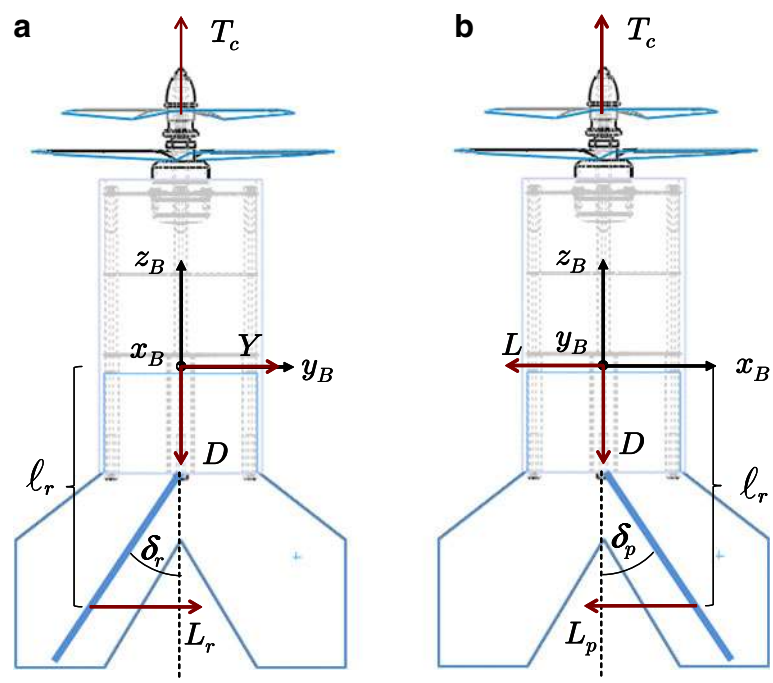

Fig. 3 Schematic of the vehicle

Therefore, the total force $\boldsymbol{F}$ is

$$\boldsymbol{F} = \begin{pmatrix} F_x \\ F_y \\ F_z \end{pmatrix} = \boldsymbol{F}_p + \boldsymbol{F}_a + \boldsymbol{F}_w$$

2.2.2 Moments

The moments acting on the micro helicopter are described as

Actuator Moments The moments due to actuators are

$$\boldsymbol{\Gamma}_{\text{act}} = \begin{pmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix}$$

where $\tau_\phi = \ell_r L_r$, $\tau_\theta = \ell_r L_p$ and $\tau_\psi = \tau_{M_1} + \tau_{M_2}$ are the control inputs with ℓ_r that represent the distance from the center of mass to the forces L_r and L_p.

Gyroscopic Moments The gyroscopic moments due to motors are described as

$$\boldsymbol{\Gamma}_{\text{gyro}} = \begin{pmatrix} q\left(I_{r_1}\omega_{r_1} - I_{r_2}\omega_{r_2}\right) \\ p\left(-I_{r_1}\omega_{r_1} + I_{r_2}\omega_{r_2}\right) \\ 0 \end{pmatrix}$$

where ω_{r_i} denotes the angular velocity of the rotor, I_{r_i} is the inertia moment of the propeller and I_{rot_i} is the moment of inertia of the rotor around its axis for $i = 1, 2$.

Aerodynamic Moments The aerodynamic moments acting on the vehicle are

$$\boldsymbol{\Gamma}_a = \begin{pmatrix} \bar{L} \\ \bar{M} \\ \bar{N} \end{pmatrix}$$

where $\bar{L}$, $\bar{M}$ and $\bar{N}$ are the aerodynamic rolling, pitching and yawing moments respectively [7, 9].

Thus, the total moment is given as

$$\boldsymbol{\Gamma} = \begin{pmatrix} \Gamma_L \\ \Gamma_M \\ \Gamma_N \end{pmatrix} = \boldsymbol{\Gamma}_{\text{act}} + \boldsymbol{\Gamma}_{\text{gyro}} + \boldsymbol{\Gamma}_a$$

2.2.3 Hover Flight Analysis

In order to obtain the behavior in hover flight, the propeller thrust and induced axial velocities in the presence of a given wind are analyzed using the Glauert's hypothesis. Figure 4 shows the vehicle perturbed with a translational wind [10, 11].

The thrust equation is described as

$$T_c = 2\rho A v_r v_i \tag{5}$$

where A represents the area of the rotor disc, ρ denotes the air density, v_o is the freestream wind velocity, v_i represents the induced wind velocity and is directed opposite to the thrust, and v_r is the resultant wind velocity.

From the Fig. 4, the resultant wind velocity v_r and the angle of attack α are determined as

$$v_r = \sqrt{(v_i - v_o \sin(\alpha))^2 + (v_o \cos(\alpha))^2} \tag{6}$$

and

$$\alpha = \sin^{-1}\left(-\frac{D_p}{T_c}\right) \tag{7}$$

where D_p is the drag force of the propeller.

Considering the case $\alpha = 0$, and $v_o = 0$, the Eqs. 5 and 6 give the induced velocity v_h of the air produced by the coaxial propellers (Prop-wash) over the control surfaces to perform the hover flight. This equation is described as

$$v_i = v_h = \sqrt{\frac{T_c}{2\rho A}} \tag{8}$$

Finally, the aerodynamic forces and moments are written by

$$\begin{aligned} D &= \frac{1}{2}\rho v_h^2 S C_D \quad & \bar{L} &= \frac{1}{2}\rho v_h^2 S b\, C_l \\ Y &= \frac{1}{2}\rho v_h^2 S C_Y \quad & \bar{M} &= \frac{1}{2}\rho v_h^2 S \bar{c} C_m \\ L &= \frac{1}{2}\rho v_h^2 S C_L \quad & \bar{N} &= \frac{1}{2}\rho v_h^2 S b\, C_n \end{aligned} \tag{9}$$

where S represents the fin-aileron area, $\bar{c}$ is the fin-aileron chord, and b is the fin-aileron span. C_D,

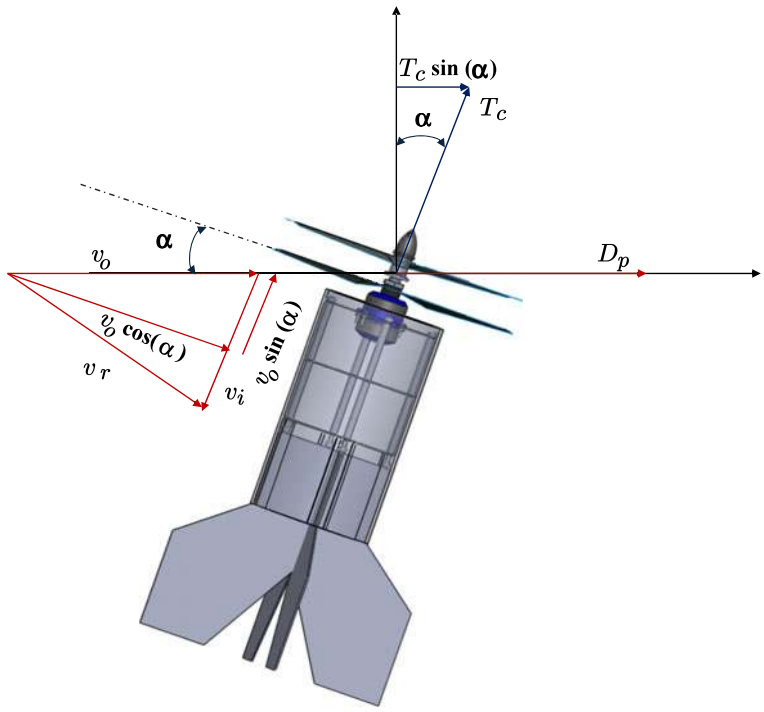

Fig. 4 Vehicle submerged in the propeller slipstream

C_Y and C_L are the aerodynamical non-dimensional coefficients of drag, sideforce and lift. C_l, C_m and C_n are aerodynamical non-dimensional coefficients of the aerodynamic rolling, pitching and yawing moments [12].

2.2.4 Equations of Motion

The nonlinear model obtained by the Newton–Euler formulation for the hover flight, i.e. $\alpha = 0$, $\beta = 0$ and $v_o = 0$, is described as

$$\begin{aligned}
\ddot{x} &= \frac{A_x}{m} c_\theta c_\psi + \frac{A_y}{m}\left(s_\phi s_\theta c_\psi - c_\phi s_\psi\right) \\
&\quad + \frac{A_z}{m}\left(c_\phi s_\theta c_\psi + s_\phi s_\psi\right) \\
\ddot{y} &= \frac{A_x}{m} c_\theta s_\psi + \frac{A_y}{m}\left(s_\phi s_\theta s_\psi + c_\phi c_\psi\right) \\
&\quad + \frac{A_z}{m}\left(c_\phi s_\theta s_\psi - s_\phi c_\psi\right) \\
\ddot{z} &= \frac{-A_x}{m} s_\theta + \frac{A_y}{m} s_\phi c_\theta + \frac{A_z}{m} c_\phi c_\theta - g \\
\ddot{\phi} &= \frac{\dot{\theta}\dot{\psi}}{c_\theta} + \frac{\dot{\theta}\dot{\phi} s_\theta}{c_\theta} + \frac{1}{I_{xx}}\left[\Gamma_L + qr\left(I_{yy} - I_{zz}\right)\right] \\
&\quad + \frac{c_\phi s_\theta}{c_\theta I_{zz}}\left[\Gamma_N + pq\left(I_{xx} - I_{yy}\right)\right] \\
&\quad + \frac{s_\phi s_\theta}{c_\theta I_{yy}}\left[\Gamma_M - pr\left(I_{xx} - I_{zz}\right)\right] \\
\ddot{\theta} &= -\dot{\phi}\dot{\psi} c_\theta + \frac{c_\phi}{I_{yy}}\left[\Gamma_M - pr\left(I_{xx} - I_{zz}\right)\right] \\
&\quad + \frac{s_\phi}{I_{zz}}\left[-\Gamma_N - pq\left(I_{xx} - I_{yy}\right)\right] \\
\ddot{\psi} &= \frac{\dot{\theta}\dot{\phi}}{c_\theta} + \frac{\dot{\theta}\dot{\psi} s_\theta}{c_\theta} + \frac{c_\phi}{c_\theta I_{zz}}\left[\Gamma_N + pq\left(I_{xx} - I_{yy}\right)\right] \\
&\quad + \frac{s_\phi}{c_\theta I_{yy}}\left[\Gamma_M - pr\left(I_{xx} - I_{zz}\right)\right]
\end{aligned} \tag{10}$$

where

$$\begin{aligned}
A_x &= -L \\
A_y &= Y \\
A_z &= T_c - D \\
\Gamma_L &= \tau_\varphi + q\left(I_{r_1}\omega_{r_1} - I_{r_2}\omega_{r_2}\right) + \bar{L} \\
\Gamma_M &= \tau_\theta + p\left(-I_{r_1}\omega_{r_1} + I_{r_2}\omega_{r_2}\right) + \bar{M} \\
\Gamma_N &= \tau_\psi + \bar{N}
\end{aligned}$$

3 Stability Analysis

In this section, the control technique for the attitude stabilization of the vehicle in hovering flight is presented using a classical backstepping procedure [13, 14]. The stability analysis demonstrates asymptotic stability about the origin of the closed-loop system. The objective of this controller is to regulate the attitude of the vehicle with different initial conditions. For simplicity, the nonlinear Eq. 10 are separated into three subsystems. One subsystem describes the longitudinal motion, and the second subsystem describes the lateral motion, and the last subsystem describes the directional motion, [15]. Since the yaw motion is mechanically stable using contra-rotating propellers, the gyroscopic moment Γ_{gyro} will essentially be zero.

Considering the condition $\phi = 0$, and $\psi = 0$, the longitudinal subsystem is written as

$$\begin{aligned}
\ddot{x} &= -\frac{L}{m}c_\theta + \frac{T_c - D}{m}s_\theta \\
\ddot{z} &= -\frac{A_x}{m}s_\theta + \frac{A_z}{m}c_\theta - g \\
\ddot{\theta} &= \frac{\tau_\theta + \bar{M}}{I_{yy}}
\end{aligned} \tag{11}$$

Defining a change of variables

$$u_\theta = \frac{\tau_\theta}{I_{yy}} + \frac{\bar{M}}{I_{yy}} \tag{12}$$

Substituting, it yields

$$\begin{aligned}
\ddot{x} &= -\frac{L}{m}c_\theta + \frac{1}{m}T_c s_\theta - \frac{1}{m}D s_\theta \\
\ddot{z} &= \frac{1}{m}T_c c_\theta - \frac{1}{m}D c_\theta + \frac{L}{m}s_\theta - g \\
\ddot{\theta} &= u_\theta
\end{aligned} \tag{13}$$

In order to stabilize the altitude of this vehicle, a nonlinear control law is proposed as

$$T_c = \frac{m}{c_\theta}\left(-k_{z_1}(z - z_d) - k_{z_2}\dot{z} + \frac{D}{m}c_\theta - \frac{L}{m}s_\theta + g\right) \tag{14}$$

where the constants $k_{z_1} > 0$ and $k_{z_2} > 0$, thus $z \to z_d$, $\dot{z} \to 0$, as $t \to \infty$. Then, the Eq. 13 can be rewritten as

$$\begin{aligned}
\ddot{x} &= -\frac{L}{m}c_\theta - \frac{L}{m}t_\theta s_\theta + g t_\theta \\
\ddot{\theta} &= u_\theta
\end{aligned} \tag{15}$$

Taking a change of variables $x_1 = x$; $x_2 = \dot{x}$; $x_3 = \theta$; $x_4 = \dot{\theta}$, the state space representation is written as follows

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -\frac{L}{m}c_{x_3} - \frac{L}{m}t_{x_3}s_{x_3} + g t_{x_3} \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= u_\theta
\end{aligned} \tag{16}$$

In order to control the previous subsystem, a nonlinear control law based on the backstepping procedure is proposed. First, let us define the error e_1 as

$$e_1 = x_1 - x_1^d \tag{17}$$

differentiating Eq. 17

$$\dot{e}_1 = \dot{x}_1 - \dot{x}_1^d$$

defining the following positive function

$$V_1 = \frac{k_1}{2}e_1^2$$

whose derivative is

$$\dot{V}_1 = k_1 e_1 \dot{e}_1 = k_1 e_1 \left(\dot{x}_1 - \dot{x}_1^d\right)$$

and using $\dot{e}_1 = \left(x_2 - \dot{x}_1^d\right)$, it yields

$$\dot{V}_1 = k_1 e_1 \left(x_2 - \dot{x}_1^d\right) \tag{18}$$

Now, let us define x_2^v as the virtual control input, such that

$$x_2^v = \dot{x}_1^d - e_1 \tag{19}$$

then, substituting Eq. 19 in Eq. 18, it implies

$$\dot{V}_1 = -k_1 e_1^2 + k_1 e_1 \left(x_2 - x_2^v\right)$$

Now, the error e_2 is defined as

$$e_2 = x_2 - x_2^v$$

it results

$$\dot{V}_1 = -k_1 e_1^2 + k_1 e_1 e_2$$

Now, the following positive function is proposed as

$$V_2 = \frac{k_2}{2} e_2^2$$

taking its derivative as

$$\dot{V}_2 = k_2 e_2 \dot{e}_2 = k_2 e_2 \left(\dot{x}_2 - \dot{x}_2^v\right)$$
$$= k_2 e_2 \left(- \left\{ \frac{L}{m} c_{x_3} + \frac{L}{m} t_{x_3} s_{x_3} - g t_{x_3} \right\} - \dot{x}_2^v \right)$$

Defining the virtual control input δ_1^v as

$$\delta_1^v = \dot{x}_2^v - \frac{k_1}{k_2} e_1 - e_2$$

then $\dot{V}_2$ yields

$$\dot{V}_2 = -k_2 e_2^2 - k_1 e_1 e_2$$
$$+ k_2 e_2 \left(\left\{ \frac{L}{m} c_{x_3} + \frac{L}{m} t_{x_3} s_{x_3} - g t_{x_3} \right\} - \delta_1^v \right)$$

Defining the error e_3 as

$$e_3 = \left\{ \frac{L}{m} c_{x_3} + \frac{L}{m} t_{x_3} s_{x_3} - g t_{x_3} \right\} - \delta_1^v$$

then, $\dot{V}_2$ yields

$$\dot{V}_2 = -k_2 e_2^2 - k_1 e_1 e_2 + k_2 e_2 e_3$$

Now, proposing a positive function V_3 as

$$V_3 = \frac{k_3}{2} e_3^2$$

whose derivative is given as

$$\dot{V}_3 = k_3 e_3 \left[\left(\frac{L}{m} t_{x_3} c_{x_3} + \frac{L}{m} s_{x_3} t_{x_3}^2 - g\left(1 + t_{x_3}^2\right) \right) x_4 - \dot{\delta}_1^v \right]$$

Let us define the virtual control input as

$$\delta_2^v = \dot{\delta}_1^v - \frac{k_2}{k_3} e_2 - e_3$$

then, $\dot{V}_3$ yields

$$\dot{V}_3 = -k_3 e_3^2 - k_2 e_2 e_3 + k_3 e_3$$
$$\times \left[\left(\frac{L}{m} t_{x_3} c_{x_3} + \frac{L}{m} s_{x_3} t_{x_3}^2 - g\left(1 + t_{x_3}^2\right) \right) x_4 - \delta_2^v \right]$$

finally, defining the error e_4 as

$$e_4 = \left(\frac{L}{m} t_{x_3} c_{x_3} + \frac{L}{m} s_{x_3} t_{x_3}^2 - g\left(1 + t_{x_3}^2\right) \right) x_4 - \delta_2^v$$

$\dot{V}_3$ yields

$$\dot{V}_3 = -k_3 e_3^2 - k_2 e_2 e_3 + k_3 e_3 e_4 \tag{20}$$

Proposing the last positive function as

$$V_4 = \frac{k_4}{2} e_4^2$$

it results

$$\dot{V}_4 = k_4 e_4 \dot{e}_4 = k_4 e_4 \left(\left(d_1 u_\theta + d_2 x_4^2 \right) - \dot{\delta}_2^v \right) \tag{21}$$

where

$$d_1 = \frac{L}{m} t_{x_3} c_{x_3} + \frac{L}{m} s_{x_3} t_{x_3}^2 - g\left(1 + t_{x_3}^2\right)$$
$$d_2 = \frac{L}{m} t_{x_3} s_{x_3} + \frac{L}{m} c_{x_3} + \frac{2L}{m} c_{x_3} t_{x_3}^2 + \frac{2L}{m} s_{x_3} t_{x_3}^2$$
$$-2g t_{x_3} - 2g t_{x_3}^3$$

From Eq. 21, a control input u_θ is proposed such that $\dot{V}_4 = -k_4 e_4^2 - k_3 e_3 e_4$, it results

$$u_\theta = \frac{1}{d_1} \left(\dot{\delta}_2^v - \frac{k_3}{k_4} e_3 - e_4 - d_2 x_4^2 \right) \tag{22}$$

now, a Lyapunov function is proposed as

$$V = V_1 + V_2 + V_3 + V_4 \tag{23}$$

and $\dot{V}$ yields

$$\dot{V} = -k_1 e_1^2 - k_2 e_2^2 - k_3 e_3^2 - k_4 e_4^2 \leq 0 \tag{24}$$

thus, the system 11 is stable in the origin [14].

In order to control the lateral subsystem, the control methodology presented to stabilize the longitudinal subsystem is employed. Considering the condition $\theta = 0$, and $\psi = 0$, the lateral subsystem is written as

$$\ddot{y} = \frac{Y}{m} c_\phi - \frac{T_c - D}{m} s_\phi$$
$$\ddot{\phi} = u_\phi \tag{25}$$

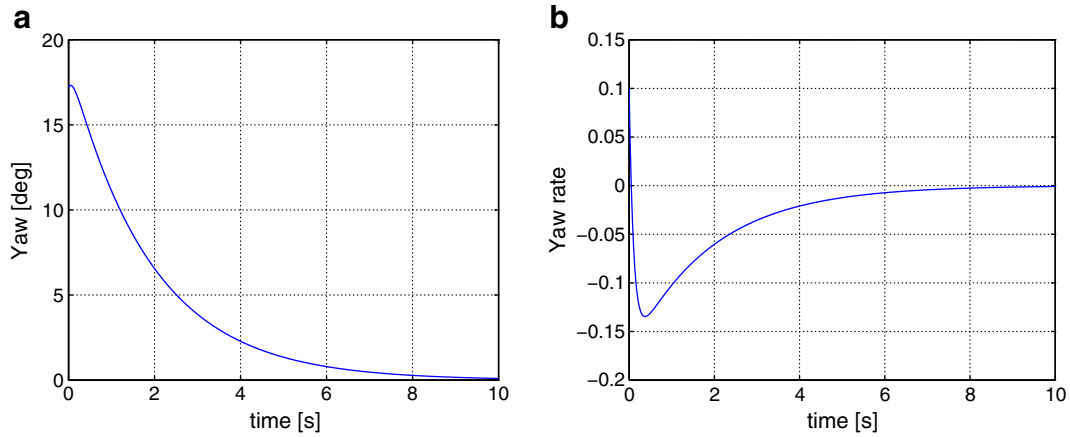

Fig. 5 Simulation results—yaw angle and angular velocity

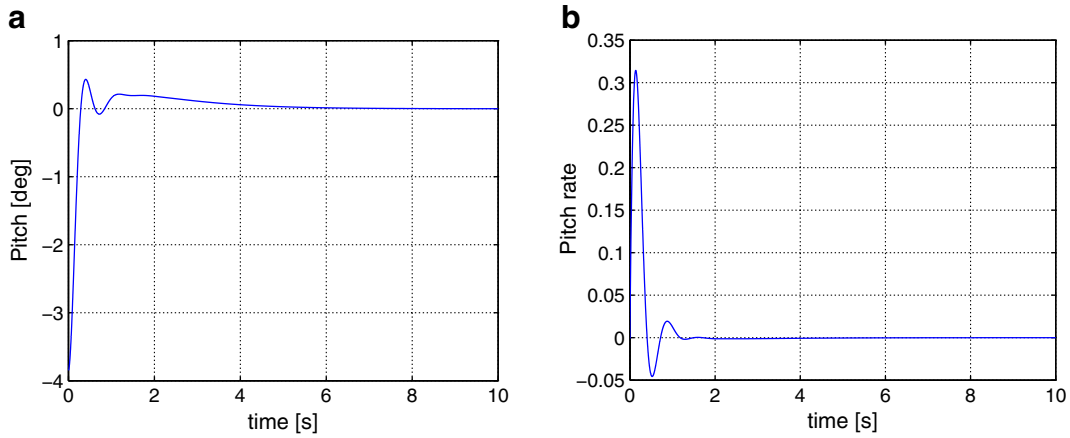

Fig. 6 Simulation results—pitch angle and angular velocity

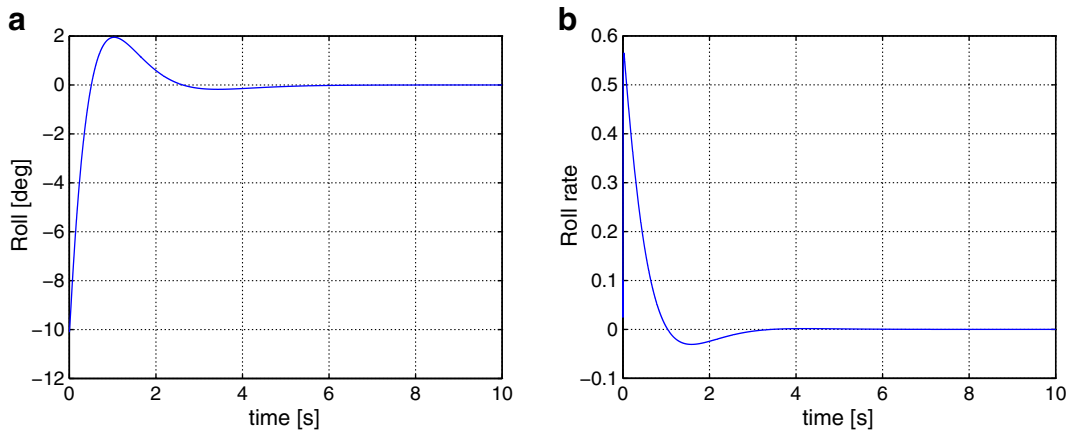

Fig. 7 Simulation results—roll angle and angular velocity

while the control law obtained considering the state variables $[x_1 = y, x_2 = \dot{y}, x_3 = \phi, x_4 = \dot{\phi}]$ is done by

$$u_\phi = \frac{1}{d_3}\left(\dot{\delta}_2^v - \frac{k_3}{k_4}e_3 - e_4 - d_4 x_4^2\right) \tag{26}$$

where

$$d_3 = -\frac{Y}{m}s_{x_3} - gc_{x_3}$$

and

$$d_4 = -\frac{Y}{m}c_{x_3} + gs_{x_3}$$

On the other hand, in order to stabilize the remaining directional subsystem, a linear control input is proposed as

$$\tau_\psi = I_{zz}(-k_{\psi_1}\psi - k_{\psi_2}\dot{\psi}) - \bar{N} \tag{27}$$

which is substituted in

$$\ddot{\psi} = \frac{\tau_\psi + \bar{N}}{I_{zz}} \tag{28}$$

Fig. 8 Gumstix and expansion board

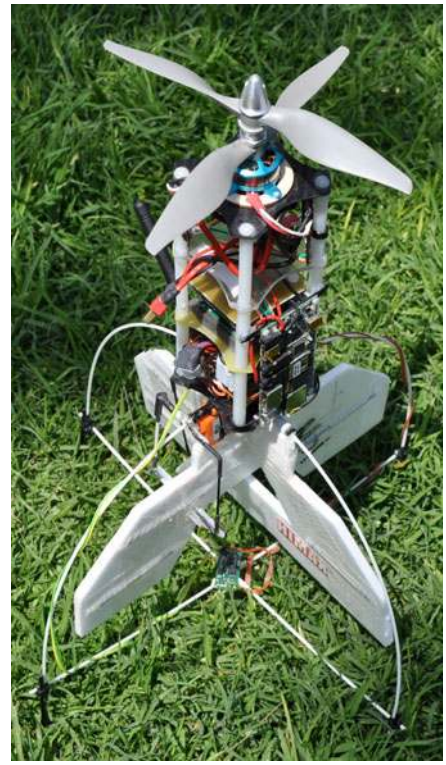

Fig. 9 Experimental platform

and it yields

$$\ddot{\psi} = -k_{\psi_1}\psi - k_{\psi_2}\dot{\psi} \tag{29}$$

Finally, the constants k_{ψ_1}, k_{ψ_2} are chosen such that the Eq. 29 is Hurwitz polynomial.

4 Numerical Simulation

In this section, the simulation results of the attitude dynamics are shown. It is observed that

Table 1 Experimental platform parameters

Parameter	Value
Fin-aileron area S	0.023 m^2
Fin-aileron chord $\bar{c}$	0.1 m
Fin-aileron span b	0.23 m
Aspect ratio	2.3
Mass vehicle (m)	0.45 Kg
Propeller	7 × 5 in
Battery LiPo	11.1v
Coaxial motor (Himax)	HC2805-1430

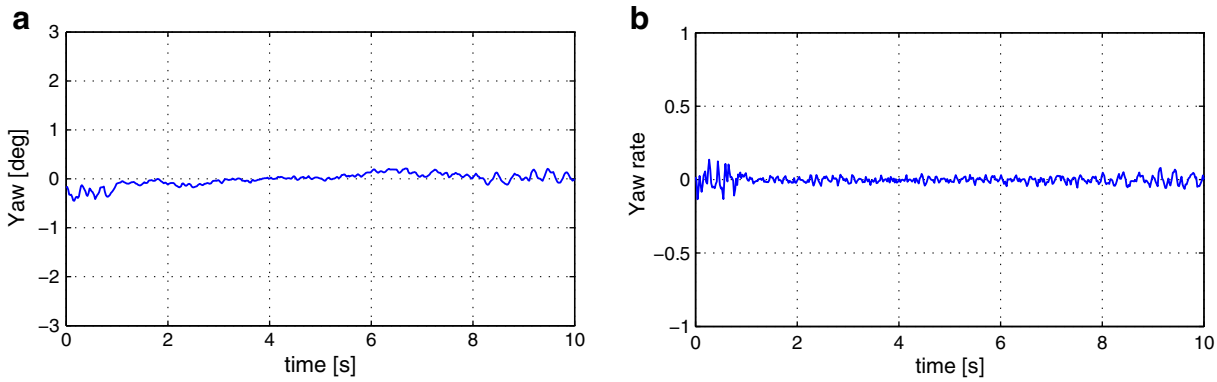

Fig. 10 Experimental results—Yaw angle and angular velocity

the controller stabilizes the attitude dynamics in a short time. The parameters and gains satisfy the tuning conditions from the stability analysis. The initial conditions used for simulation are $\psi(0) = 17.2$, $\theta(0) = -4.38$, and $\phi(0) = -10.1$

In Fig. 5a the response of the yaw motion is shown, while the yaw rate with respect to time is shown in Fig. 5b. The pitch angle response and pitch rate of the closed-loop system are plotted in Fig. 6a and b, respectively. Finally, Fig. 7a and b represent the roll motion and the roll rate. In general, observe that all states $(\phi,\dot{\phi},\theta,\dot{\theta},\psi,\dot{\psi})$ converge to the equilibrium in a short time.

5 Experimental Platform

The prototype consists of an embedded system based on the Gumstix Overo Fire Computer-on-Module (COM) [16], the Summit expansion board and the electronic board which interface all the sensors and actuators with the Gumstix COM (Fig. 8). This COM has an ARM Cortex-A8 based 720 Mhz platform which is ideal for our application due to its characteristics such as 802.11g WiFi, 512 MB RAM, microSD card slot and several communication protocols (I^2C, SCI, USB, SPI) to adquiere and control the different sensors and actuators involved in the avionics of the MAV. The most important feature of this COM is that it can run the Xenomai Real-Time framework [17], which is a real-time development framework cooperating with the Linux kernel that is achieved through a dual-kernel approach with the objective to allow deterministic response times regardless of the standard Linux implementation in order to provide hard real-time support to user-space applications [18]. The complete description of the

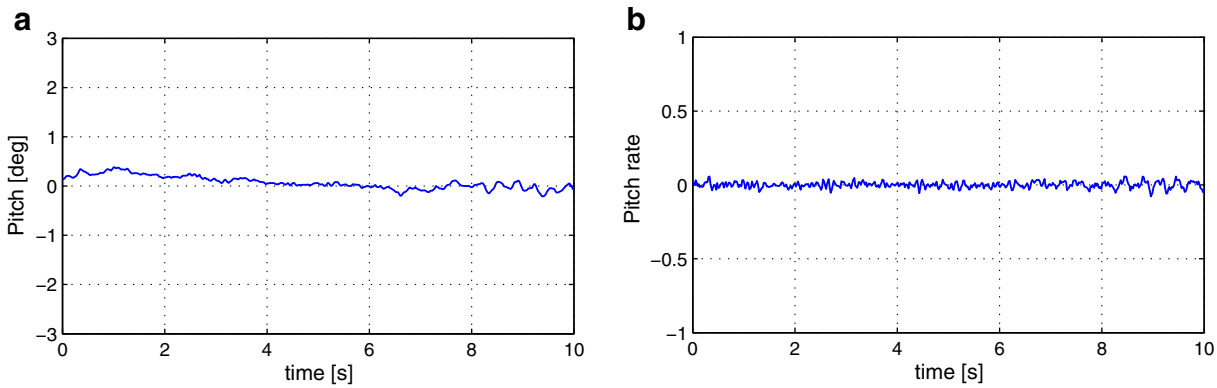

Fig. 11 Experimental results—Pitch angle and angular velocity

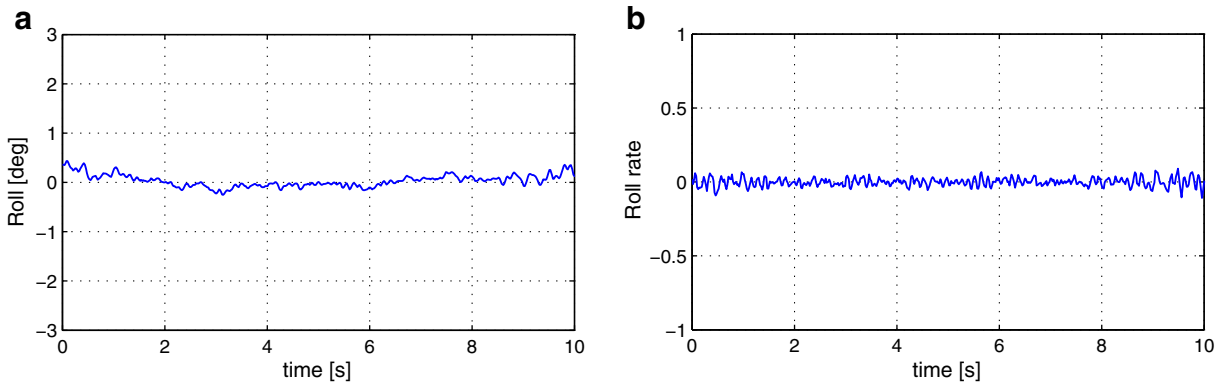

Fig. 12 Experimental results—Roll angle and angular velocity

avionics, the hardware and the software layer of this vehicle are shown in [19]. Figure 9 shows the experimental platform built to validate the simulation results and allowing us to reach the next stage of the project (forward flight). Table 1 shows the platform parameters.

Figure 10 shows the performance in yaw motion of the vehicle during the autonomous flight. The pitch angle and pitch rate are depicted in Fig. 11 while Fig. 12 illustrates the behavior of the roll angle and roll rate of this aerodynamic platform, which are the goal in this stage of the project, i.e., stability of the aerial vehicle in hover flight.

6 Conclusions

This paper addresses the description of a micro aerial vehicle launched by a semiautonomous airplane. The dynamic model considering aerodynamical forces and moments was obtained and simulated. The real-time embedded system was designed in order to satisfy all the avionics requirements, and one platform was built and tested in several experimental tests showing promising results to reach the next stage of the project (MCR vehicle launched by the air shuttle transporter). Finally, a complete backstepping controller was tuned and validated in the micro aerial vehicle for hover flight, which represents the beginning of the whole project.

Acknowledgements This work was partially supported by National Council of Science and Technology-CONACYT Mexico, PROMEP UPPACH-003, and Centre National de la Recherche Scientifique-CNRS France.

References

1. Schafroth, D., Bermes, C., Bouabdallah, S., Siegwart, R.: Modeling, System Identification and Robust Control of a Coaxial Micro Helicopter. Control Engineering Practice, Elsevier (2010)
2. Bouabdallah, S., Siegwart, R., Caprari, G.: Design and control of an indoor coaxial helicopter. In: International Conference on Intelligent Robots and Systems 2006, Beijing, China (2006)
3. Dzul, A., Hamel, T., Lozano, R.: Modeling and nonlinear control for a coaxial helicopter. In: IEEE International Conference on Systems, Man and Cybernetics 2002, Hammamet, Tunisia (2002)
4. Frazzoli, E., Dahleh, M.A., Feron, E.: Trajectory tracking control design for autonomous helicopters usign a backstepping algorithm. In: American Control Conference (ACC 2000), Chicago Illinois, USA (2000)
5. Fankhauser, P., Bouabdallah, S., Leutenegger, S., Siegwart, R.: Modeling and decoupling control of the coax micro helicopter. In: International Conference on Intelligent Robots and System (IROS 2011), San Francisco, CA, USA (2011)
6. Stengel, R.F.: Flight Dynamics. Princeton University Press, USA (2004)
7. Etkin, B., Duff Reid, L.: Dynamics of Flight: Stability and Control, 3rd edn. John Wiley and Sons (1996)
8. Goldstein, H., Poole, C.P., Safko, J.L.: Classical Mechanics, 2nd edn. Adison-Wesley, USA (1983)
9. Cook, M.V.: Flight Dynamics Principles, 2nd edn. Butterworth-Heinemann, USA (2007)
10. McCormick, B.W.: Aerodynamics of V/STOL Flight. Dover Publications, USA (1967)
11. McCormick, B.W.: Aerodynamics, Aeronautics and Flight Mechanics. John Wiley and Sons, USA (1995)

12. Phillips, W.F.: Mechanics of Flight. John Wiley and Sons, USA (2004)
13. Castillo, P., Lozano, R., Dzul, A.: Modelling and Control of Miniflying Machines. Springer-Verlag, London (2005)
14. Khalil, H.: Nonlinear Systems. Prentice Hall, New York (1995)
15. Stevens, B.L., Lewis, F.L.: Aircraft Control and Simulation. John Wiley and Sons, USA (1992)
16. Gumstix Inc. Gumstix. www.gumstix.com (2012). Accessed 16 Jan 2012
17. Xenomai, Xenomai. www.xenomai.org (2012). Accessed 20 Jan 2012
18. Choi, B.W., Shin, D.G., Park, J.H., Yi, S.Y., Gerald, S.: Real-time control architecture using Xenomai for intelligent service robots in USN environments. Intel. Serv. Robotics **2**(3), 139–151 (2009)
19. Chauffaut, C., Espinoza, E.S., Escareno, J., Lozano, R.: Towards gun- and aircraft - launched MAVs: embedded flight control system. In: IFAC Conference on Embedded Systems, Computational intelligence and Telematics in Control, Würzburg, Germany (2012)

J Intell Robot Syst
DOI 10.1007/s10846-012-9743-0

Toward Aerial Grasping and Manipulation with Multiple UAVs

Vicente Parra-Vega · Anand Sanchez · Carlos Izaguirre · Octavio Garcia · Francisco Ruiz-Sanchez

Received: 29 June 2012 / Accepted: 13 July 2012

Abstract In this paper, a multiple UAVs control scheme is developed considering the full nonlinear position/orientation model of a *j*-Quadrotor system. A novel second order sliding mode controller is presented which guarantees exponential and robust tracking of admissible time-varying pose. The harmful chattering is not involved and no dynamic model is required to implement the controller to yield fast and precise tracking. Additionally, well-posed terminal and controlled time convergence allows an enforced contact at given pre-defined stable contact points at the same time. A stiffness control is proposed for grasping objects considering virtual linkages approach. Our approach yields high performance from the control system, in contrast to other simple controllers proposed for load carrying. In this sense, our advanced nonlinear control solves the apparent limitations imposed by the available technology from the viewpoint of the precise tracking control, and control of the inherent unstable underactuated dynamics, for frictionless contact points (neither rolling nor sliding are considered). A numerical simulation study, under various conditions, shows the numerical feasibility of the proposed approach.

Keywords Underactuated dynamic model · Second order sliding mode control · Aerial grasping · Stiffness control · Cooperative Quadrotors

This work was supported in Mexico by the Conacyt grants 133346 and 133544.

V. Parra-Vega · A. Sanchez (✉) · C. Izaguirre · F. Ruiz-Sanchez
Robotics and Advanced Manufacturing Division, Research Center for Advanced Studies (Cinvestav), Saltillo Campus, Coahuila, Mexico
e-mail: anand.sanchez@cinvestav.mx

V. Parra-Vega
e-mail: vparra@cinvestav.mx

C. Izaguirre
e-mail: carlos.iza.es@gmail.com

F. Ruiz-Sanchez
e-mail: fruiz@cinvestav.mx

V. Parra-Vega · A. Sanchez · C. Izaguirre · F. Ruiz-Sanchez
The Laboratory of Non-inertial Robots and Man-machine Interfaces, Research Center for Advanced Studies (Cinvestav), Monterrey Campus, Nuevo Leon, Mexico

O. Garcia
Biomedical Engineering and Physics Division, and The Laboratory of Non-inertial Robots and Man-machine Interfaces, Research Center for Advanced Studies (Cinvestav), Monterrey Campus, Nuevo Leon, Mexico
e-mail: ogarcias@cinvestav.mx

1 Introduction

Although, it has been acknowledged the potential of rigid grasp for aerial objects with multiple

Published online: 05 September 2012

UAVs in diverse applications, there has been little, if any, formal description on the problem as such, *aerial grasp manipulation* is a concept under development. The fact is that different approaches handle loosely the terms grasping and manipulation for UAVs which suggest the need to review and clarify such a concept. An interesting particular scheme for load carrying has been widely studied and successfully implemented, for full scale [12] and multiple small helicopters [2], using only orientation dynamics. However, this paper presents an approach to the problem of grasping and manipulation of a free-flying object for j-UAVs, coined here as *aerial grasp manipulation* which is conceived in the realm of classical multi-finger rigid grasping [16]. The problem is to immobilize an object with a given contact points set to establish stable grasp of a free-flying dynamic object, motionless at time of contact, then controlling external and internal forces to move and manipulate the object [19]. Clearly, the transition from free to constrained motion is included [1], otherwise it is assumed that the object is already in a stable grasp condition.

1.1 Contribution and Organization

The problem of grasping and manipulation of a free-flying object is discussed in Section 2, then a brief review and a proposed classification are introduced in Section 3. This section also establishes a distinction on different modalities and information of what is known as aerial grasp manipulation. Section 4 shows the model of the position and orientation dynamics of a Quadrotor UAV, as well as its structural properties of a convenient open-loop error equation, including the system of j-Quadrotors. Section 5 introduces the controller and its stability analysis which guarantees the fast and robust tracking of admissible trajectories, including convergence to a desired contact time. This last stability property in particular allows Quadrotors to meet at a given point and at a given time. Section 6 presents the grasping and manipulation strategy to enforce a stable grasp map for all time. Simulations are presented and thoroughly discussed in Section 7, which lead to some remarks presented in Section 8. Finally, conclusions are given in Section 9.

2 The Problem

2.1 The Physics of Interaction with a Free-flying Object with UAVs

The concept most exploited in the literature is some sort of object or load carrying with rigid grip using one UAV or with cables using several UAVs. However, rigid grasp has not been really studied, it does not involve cables but rigid coupling, with several UAVs. Rigid grasp of an object with j-UAVs implies the ability to exert forces into the object mostly along x and y axes and to compensate for interacting forces among the UAVs. In contrast, load carrying or object grabbing mostly stands for compensation of forces in the z direction. The former maintains its center of gravity, while the latter suffers from inverted pendulum effect because its center of mass is displaced downwards. However, exertion of forces occurs mostly along x, y which requires higher pitch. This fact brings the problem that there is a physical limit imposed by the thrust required to compensate for gravitational force along z. That is, the more pitch the less thrust and viceversa. This happens because there is not independent actuation to control the displacements along x and y axes. The displacements along these directions are achieved by controlling roll and pitch angles, in other words given such desired displacements a virtual controller is built, then the corresponding roll and pitch angles are found, which finally will stand for the desired roll and pitch angles for the orientation controller, allowing the interaction into x and y axes.

It is clear that for rigid frames and objects, the interaction forces and moments are propagated all along the bodies, then fast and robust controllers must compensate or accommodate such interactions to avoid instabilities. To accommodate, distribute and share loads along x, y, z due to the object or other UAVs, some sort of passive linkage or compliant mechanism may be required, otherwise grasping will be possible only with light objects that require small pitch angles, that is, small f_x, f_y. In this paper the flight control of multiples UAVs is presented in order to establish grasp and manipulation without considering this aforementioned problem.

2.2 Problem Statement

The problem of aerial grasping only with rotary wing UAVs is considered, leaving out UAVs with fixed wings because those cannot regulate their position. Being said that, single main rotor like a typical helicopter or multiple main rotors such as a Quadrotor are considered because of its hovering characteristic to position the UAV at the given point in order to grasp the object. However, grasping a motionless object in the air requires to exploit at the limit the highly maneuvering characteristic of the underactuated vehicles. In fact, this task can be classified as an extreme flight regime; it requires high-end models, avionics, coordination and control in order to achieve grasping, and it needs top of the line precise positioning, stable hovering, sensing and time synchronization of multiple UAVs, as well as extraordinary flight control. Then, it comes to no surprise that so far there is none scheme that tackles the problem of *aerial multi-grasp manipulation*, defined as the *grasping in the air of an object with multiple UAVs and then manipulate it (controlling the pose over time) once stable grasp is achieved*.

2.3 Motivation

There are some distinct useful features of Quadrotors to solve the problem: (i) their small size, mass and inertia make them convenient for multiple UAV cooperation, their ability for underactuated maneuvering (it can reach any pose, but not through any trajectory), and (ii) their ability to positioning and stable hovering. All these characteristics are appealing for interaction tasks, but this involves a high analytical and computational cost. A Quadrotor is, in general, unstable, that is the linear approximation is of non-minimum phase similar to the helicopter case and the nonlinear model is not passive, from torque input to angular velocity output due to its underactuation. To make things worse, orientation dynamics is very fast in comparison to the position dynamics, which are related by a dynamic mapping whose solution is non-causal. This imposes to consider as much information of the system as possible, consequently, in particular there should be considered a formal approach to take into account full nonlinear models to design robust and fast nonlinear control for tracking. Solving, even partially, this problem would allow the formulation of diverse task on load carrying, deploying and recovering, and in general drone cooperation that involves physical interaction.

2.4 Hypothesis

Grasp manipulation in the air is perhaps one of the dream flight regime of a j-Quadrotor system. It would require such performance that the theoretical and technological challenges that arise seem to amount for an unsurpassed scientific problem nowadays. However, to surmount such set of challenges, some stringent hypothesis are required to, at least, work out a small part of the whole problem. As a first step in this direction, the problem based on the following technological and theoretical hypothesis is study: (a) no technological constraints are considered, such as slow sensor response with limited resolution, multisampling rate of sensors and actuators, latency requirements, limitations on mechanical design of the appendage to establish convenient contact to the object, and processing unit is able to guarantee fast and constant sampling update; (b) full state feedback is available; (c) reasonably it is assumed unknown dynamic model and unknown parameters, but full knowledge on kinematic model; (d) the dynamic object is motionless before any contact, whose invariant pose is known beforehand; and (e) there exists a motion planning that delivers online admissible desired position trajectories and desired contact force that ensures a stable grasp.

2.5 Proposed Solution

The full nonlinear model of position and orientation is assumed, that is the $\mathbb{R}^6$ model with four inputs, in contact to a rigid object. Firstly, a second order sliding mode controller is synthesized for the orientation dynamics to achieve fast and robust exponential convergence to desired smooth and bounded $\boldsymbol{\omega}_d(t) \in C^2$ trajectories. Then, a similar controller is developed for the position dynamics assuming virtual inputs. Secondly, an analytical and causal solution is found for roll and pitch angles, and for zero yaw. Henceforth,

virtual inputs are build upon controlling angular velocities of the orientation dynamics. Thirdly, a time-based generator is proposed to induce terminal stability within the invariant manifold (the sliding surface) to enforce convergence at a given desired time. This allows j Quadrotors to grasp the object at the same time in j contact points. Finally, an additional control term is designed to control the interaction forces. This term is based on virtual linkages approach, using a stiffness control approach, to guarantee a stable grasp, and it is proved the stability of the whole closed-loop system. This builds a solution to the problem of grasping and manipulation, with multiple UAVs, of a motionless free-flying dynamic object located off ground. Then, exertion of a given set of internal forces is required to guarantee stable grasp and be able to commute to controlling external forces to move the object.

3 Brief Background

3.1 Rough Classification

The described problem is virtually non-existent in the literature and, in this paper, it is confined to the discussion and review of related problems such as load carrying, load gripping and grabbing as well as rigid formation of UAVs. Briefly, aerial grasp has been loosely used indistinctly to refer to grabbing of an object with an UAV. The following classification arises, without much rigor, but depending on how the object is carried or grasped:

- *Load Carrying Rigidly*. One UAV is carrying an object rigidly attached to its mainframe. In this case, the UAV is likely to be off-centered or imbalanced, diminishing the flight capability.
- *Load Carrying with Cables*. It considers at least three UAVs with cables, whose ends are attached to the object at a point where static equilibrium arises. Flight is difficult because the object attracts the UAVs, then this configuration is useful only for large objects, *w.r.t.* to the size of the UAVs [7].
- *Aerial Gripping*. An object is already rigidly attached to the UAV all the time [8].
- *Ground Grabbing*. The object is at the ground, then the UAV encounters the aerodynamical problem of *ground effect* from hovering near ground which looses thrust, then grabs with an active device the object.
- *Aerial Grabbing*. A static aerial object is grabbed by magnetic or gripper-type mechanism. Then, the object is rigidly attached to the mainframe of the UAV. When two or more UAVs are involved, a new UAV arises because the two original mainframes are now rigidily connected through the object.
- *Aerial Manipulation*. The pose of an object is passively controlled in the air by two or more UAVs.
- *Aerial Grasp and Manipulation*. A motionless dynamic object is grasped without any ground effect, then its pose is controlled independently of the UAV pose.

The last category is studied in this paper.

3.2 Relevant Works

Aerial grabbing is proposed in [15] with a cooperative scheme for load transportation with a simplified model and control, and planning of this tasks is further proposed in [7]. Gripping is presented in [14] for magnetic coupling. Interestingly, with a claimed submilimiter precision, aerial gripping is proposed in [8], and a simplified control scheme under rigid constraints is discussed in [9]. There are other references in particular impressive videos can be found in a popular video web channel, however there does not exist formal publications on the algorithmic foundations.

4 The Dynamic Model of the Quadrotor System

4.1 The Dynamic Model of a Quadrotor

The dynamic model of a Quadrotor is basically obtained representing the aerial vehicle as a rigid body evolving in 3D and subject to one force and three moments [6, 10]. Let us consider earth fixed frame $\mathcal{I} = \{e_x, e_y, e_z\}$ and body fixed frame $\mathcal{A} = \{e_x^b, e_y^b, e_z^b\}$, as seen in Fig. 1. The center of mass and the body fixed frame origin are assumed to

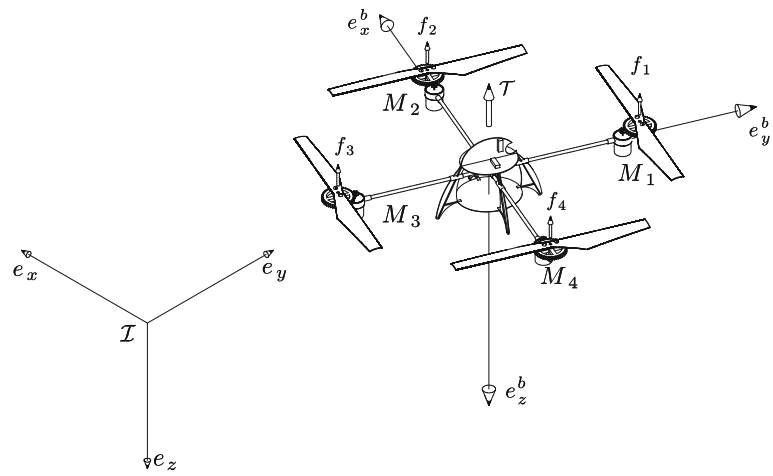

Fig. 1 The UAV system. f_i represents the thrust of motor M_i and $\mathcal{T}$ is the main thrust

coincide. The orientation of the rigid body is given by a rotation $\mathcal{R} : \mathcal{A} \longrightarrow \mathcal{I}$, where $\mathcal{R} \in SO(3)$ is an orthogonal rotation matrix, parameterized by the Euler angles ψ, θ, ϕ (yaw, pitch, roll). Newton–Euler equations of motion state the dynamics of the quad-rotor as follows:

$$m\ddot{\xi} = -\mathcal{T}\mathcal{R}e_z + F(t) \tag{1}$$

$$\dot{\mathcal{R}} = \mathcal{R}\omega^{\times} \tag{2}$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} + \tau + d(t) \tag{3}$$

where $\xi = (x, y, z)^T$ denotes the position of the center of mass of the airframe in the frame $\mathcal{I}$ relative to a fixed origin, $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3)^T \in \mathcal{A}$ denotes the angular velocity of the airframe expressed in the body fixed frame. m denotes the mass of the rigid object and $\mathbf{J} \in \mathbb{R}^{3\times3}$ denotes the constant inertia matrix around the center of mass (expressed in the body fixed frame $\mathcal{A}$). $\omega^{\times}$ denotes the skew-symmetric matrix of the vector $\boldsymbol{\omega}$, which is given by

$$\omega^{\times} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}$$

$\mathcal{T} \in \mathbb{R}_+$ represents the magnitude of the principal non-conservative forces applied to the object. $F(t)$ represents the external forces applied to the aerial vehicle, such that in the absence of forces exerted by the environment (aerodynamic reaction forces, etc.) $F(t) = mge_z$. $\tau \in \mathcal{A}$ is the control torque, and $d(t) \in \mathbb{R}^3$ represents the external torque disturbances induced by $F(t)$ which is assumed to be smooth and bounded.

4.1.1 The Open-Loop Error Equation

In order to design a control law, it is useful to define a dynamic equation parameterized by the error which is called open loop error equation.

Let us define a parametrization Y_r in terms of a nominal reference $\boldsymbol{\omega}_r$, to be defined, and its derivative $\dot{\boldsymbol{\omega}}_r$, as follows

$$Y_r = \mathbf{J}\dot{\boldsymbol{\omega}}_r + \omega^{\times}\mathbf{J}\boldsymbol{\omega}_r + \omega_r^{\times}\mathbf{J}\boldsymbol{\omega} - \omega_r^{\times}\mathbf{J}\boldsymbol{\omega}_r \tag{4}$$

Introducing Eq. 4 into Eq. 3 yields

$$\mathbf{J}\dot{S}_r + S_r^{\times}\mathbf{J}S_r = \tau + d(t) - Y_r \tag{5}$$

where the error coordinates S_r are defined by

$$S_r = \boldsymbol{\omega} - \boldsymbol{\omega}_r \tag{6}$$

At this point, the control objective is to design a τ such that S_r is stable despite the presence of bounded disturbances.

4.2 The Dynamic Model of a Set of *j*-Quadrotors

Based on the dynamic model of a Quadrotor presented in Section 4.1, we now present the dynamic model of N Quadrotors. The complete model is such that the $j-th$ element corresponds to the j-Quadrotor, whose dynamic model is given by

$$\begin{aligned} m_j\ddot{\xi}_j &= -\mathcal{T}_j\mathcal{R}_j e_z + F_j(t) \\ \dot{\mathcal{R}}_j &= \mathcal{R}_j\omega_j^{\times} \\ \mathbf{J}_j\dot{\boldsymbol{\omega}}_j &= -\omega_j^{\times}\mathbf{J}_j\boldsymbol{\omega}_j + \tau_j + d_j(t) \end{aligned} \tag{7}$$

where j goes from 1 to N. Then, a compact form of the dynamic model of N Quadrotors is given by

$$\begin{aligned}\bar{m}\ddot{\Xi} &= -\bar{\mathcal{T}}\bar{\mathcal{R}}e_z + \bar{F}\\ \dot{\bar{\mathcal{R}}} &= [\bar{\mathcal{R}}]\,\hat{\Omega}\\ \bar{\mathbf{J}}\dot{\Omega} &= -\left[\hat{\Omega}\right]\bar{\mathbf{J}}\Omega + \bar{\tau} + \bar{d}\end{aligned} \tag{8}$$

where

$\Xi = (\xi_1^T, \ldots, \xi_N^T)^T$, $\Omega = (\boldsymbol{\omega}_1^T, \ldots, \boldsymbol{\omega}_N^T)^T$,
$\bar{\mathcal{R}} = (\mathcal{R}_1^T, \ldots, \mathcal{R}_N^T)^T$, $\hat{\Omega} = ((\omega_1^\times)^T, \ldots, (\omega_N^\times)^T)^T$,
$\bar{m} = \text{blockdiag}(m_1 I, \ldots, m_N I)$, $\bar{\tau} = (\tau_1^T, \ldots, \tau_N^T)^T$,
$\bar{\mathcal{T}} = \text{blockdiag}(\mathcal{T}_1 I, \ldots, \mathcal{T}_N I)$, $\bar{d} = (d_1^T, \ldots, d_N^T)^T$,
$\bar{\mathbf{J}} = \text{blockdiag}(\mathbf{J}_1, \ldots, \mathbf{J}_N)$, $\bar{F} = (F_1^T, \ldots, F_N^T)^T$,

I represents the 3×3 identity matrix and $[X]$ denotes the block diagonal matrix whose diagonal block elements are the elements of X.

5 Control Design and Stability Analysis

In this section a Second Order Sliding Mode (SOSM) controller is provided for the system 1–3 [5, 18].

5.1 Attitude Control Design

We employ the unit quaternion as the attitude representation. Using this representation the attitude control design does not suffer from singularities. The unit quaternion is defined as

$$\mathbf{q} = \begin{pmatrix} q_0 \\ q \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\mu}{2}\right) \\ e\sin\left(\frac{\mu}{2}\right) \end{pmatrix} \tag{9}$$

where e is the Euler axis and μ is the Euler angle. The unit quaternion satisfies the following constraint

$$\mathbf{q}^T\mathbf{q} = q_0^2 + q^T q = 1 \tag{10}$$

and it is related to the angular velocity $\boldsymbol{\omega}$ by the following differential equations

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{q}_0 \\ \dot{q} \end{pmatrix} = \begin{pmatrix} -\frac{1}{2}q^T\boldsymbol{\omega} \\ \frac{1}{2}\left(q_0 I + q^\times\right)\boldsymbol{\omega} \end{pmatrix} \tag{11}$$

Let us define the angular velocity error $\boldsymbol{\omega}_e$ as follows

$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\omega}_d \tag{12}$$

where $\boldsymbol{\omega}_d$ is the desired angular velocity expressed in the body fixed frame.

Let us consider the following nominal reference

$$\boldsymbol{\omega}_r = \boldsymbol{\omega}_d - \alpha q_e + S_d - \gamma\sigma \tag{13}$$

where $\dot{\sigma} = \text{sgn}(S_q)$, feedback gains, $\alpha > 0$ and γ is diagonal positive definite matrix; the function $\text{sgn}(X) = (\text{sgn}(x_1), \text{sgn}(x_2), \text{sgn}(x_3))^T$ stands for the input wise discontinuous function of X, and

$$S_q = S - S_d \tag{14}$$

$$S = \boldsymbol{\omega}_e + \alpha q_e \tag{15}$$

$$S_d = S(t_0)\exp(-k(t - t_0)) \tag{16}$$

with $k > 0$ and $S(t_0)$ stands for $S(t)$ at $t = t_0$. $\mathbf{q}_e = (q_{0e}, q_e^T)^T$ is the relative attitude error defined as

$$\mathbf{q}_e = \mathbf{q} \otimes \mathbf{q}_d^* \tag{17}$$

where $\otimes$ denotes the operator for quaternion multiplication, $\mathbf{q}_d = (q_{0d}, q_d^T)^T$ is the desired attitude, such that $q_{0d}(t), q_d(t)$ are one time differentiable functions and $\mathbf{q}_d^*$ is the inverse of $\mathbf{q}_d$. The vectorial part q_e is given by

$$q_e = -q_0 q_d + q_{0d} q - q^\times q_d \tag{18}$$

Notice that $\dot{\boldsymbol{\omega}}_r$ is discontinuous, because of $\dot{\sigma} = \text{sgn}(S_q)$, and $S_q(t_0) = 0$ for any initial condition. From Eqs. 6, 12, 13 and 15 the dynamic error coordinates S_r are given by

$$S_r = S_q + \gamma\sigma \tag{19}$$

5.1.1 Structural Properties of the Open-Loop Error Equation

There exist positive scalars β_i for $i = 0, \ldots, 4$, such that

$$\begin{aligned} 0 < \beta_0 < \lambda_{\min}(\mathbf{J}) \le \|\mathbf{J}\| &\le \lambda_{\max}(\mathbf{J}) < \beta_1 < \infty\\ \|q_e\| &< 1\\ \|\boldsymbol{\omega}_r\| &\le \beta_2 + \|\gamma\|\|\sigma\|\\ \|\dot{\boldsymbol{\omega}}_r\| &\le \beta_3 + \beta_4\|\boldsymbol{\omega}_e\| \end{aligned} \tag{20}$$

where $\lambda_{\min}(\mathbf{J}), \lambda_{\max}(\mathbf{J})$ stand for the minimum and maximum eigenvalues of matrix $\mathbf{J} \in \mathbb{R}^3$, $\|\mathbf{J}\| = \sqrt{\lambda_{\max}(\mathbf{J}^T\mathbf{J})}$ and $\|\cdot\|$ stands for the vector Euclidean norm.

From Eqs. 3, 4 and using Eq. 20, $d(t) - Y_r$ can be bounded as

$$\begin{aligned}\|d(t) - Y_r\| &\leq \|d(t)\| + \|\mathbf{J}\|\|\dot{\boldsymbol{\omega}}_r\| + 2\|\boldsymbol{\omega}\|\|\mathbf{J}\|\|\boldsymbol{\omega}_r\| \\ &\quad + \|\mathbf{J}\|\|\boldsymbol{\omega}_r\|^2 \\ &\leq \|d(t)\| + \beta_1\beta_3\|\boldsymbol{\omega}_e\| + 2\beta_1\beta_2\|\sigma\|\|\boldsymbol{\omega}\| \\ &\quad + 2\beta_1\beta_2\|\gamma\|\|\sigma\| + \beta_1\|\gamma\|^2\|\sigma\|^2 + \beta_5 \\ &\leq \eta(t) \end{aligned} \tag{21}$$

where $\beta_5 = \beta_1\beta_4 + \beta_1\beta_2^2$, and $\eta(t)$ is a state-dependent function. Notice that, $\eta(t)$ considers all the external torques including state-dependence of $d(t)$.

Consider the following control law

$$\tau = -K_d S_r \tag{22}$$

where K_d is a diagonal positive definite matrix. We now have the following result.

Theorem 1 *Consider the attitude dynamics 3 in closed loop with the controller 22. Then, semi-global exponential tracking is assured, provided that γ in Eq. 19 and K_d are large enough, for small initial errors.*

The proof of this theorem is given in the Appendix.

5.2 Control Design for Position

Consider the translational dynamics 1. Now, let us define the following virtual control

$$u = \mathcal{T}\mathcal{R}e_z \tag{23}$$

Then, the system 1 can be rewritten as follows

$$m\ddot{\xi} = -u + F(t) \tag{24}$$

Following a similar procedure from the Section 5.1, the control design becomes straightforward. The parametrization can be written in terms of a nominal reference ξ_r as follows

$$m\ddot{\xi}_r = \bar{Y}_r \tag{25}$$

Introducing Eq. 25 into Eq. 24 yields

$$m\dot{\bar{S}}_r = -u + F(t) - \bar{Y}_r \tag{26}$$

where $\bar{S}_r = \dot{\xi} - \dot{\xi}_r$.

Consider the following nominal reference $\dot{\xi}_r$

$$\dot{\xi}_r = \dot{\xi}_d - \bar{\alpha}\xi_e + \bar{S}_d - \bar{\gamma}\bar{\sigma} \tag{27}$$

$$\dot{\bar{\sigma}} = \operatorname{sgn}\left(\bar{S}_q\right) \tag{28}$$

where the tracking error $\xi_e = \xi - \xi_d$, reference trajectory $\xi_d(t) \in C^2$, feedback gains $\bar{\alpha}, \bar{\gamma}$ are diagonal positive definite matrices, and

$$\bar{S}_q = \bar{S} - \bar{S}_d \tag{29}$$

$$\bar{S} = \dot{\xi}_e + \alpha\xi_e \tag{30}$$

$$\bar{S}_d = \bar{S}(t_0)\exp(-\bar{k}(t - t_0)) \tag{31}$$

for $\bar{k} > 0$ and $\bar{S}_r$ rewritten as

$$\bar{S}_r = \bar{S}_q + \bar{\gamma}\bar{\sigma} \tag{32}$$

As before, $F(t) - \bar{Y}_r\bar{\Theta}$ can be bounded as

$$F(t) - \bar{Y}_r\bar{\Theta} \leq \|F(t)\| + m\|\ddot{\xi}_r\| \tag{33}$$

$$\leq \|F(t)\| + m(\bar{\beta} + \bar{\alpha}\|\dot{\xi}_e\|) \tag{34}$$

$$\leq \bar{\eta}(t) \tag{35}$$

where $\bar{\beta}$ is a positive constant, and $\bar{\eta}(t)$ is a state-dependent function. Notice that, $\bar{\eta}$ not only includes all the external forces affecting the aerial vehicle (buoyancy forces, aerodynamic forces, gravity, etc.) but also a general state-dependence of $F(t)$.

Then, a control law that assures semiglobal exponential tracking, in closed-loop with system 1, is given by

$$u = \bar{K}_d\bar{S}_r \tag{36}$$

where $\bar{K}_d$ is a diagonal positive definite matrix.

In a similar way to the proof of the Theorem 1, it follows that $\bar{S}_r$ and $\dot{\bar{S}}_r$ are upper bounded, and the sliding mode in $\bar{S}_q(t) = 0$ is enforced for all time. In this way, tracking errors are constrained to evolve on a manifold that has exponential solution toward the desired trajectory $\xi_d(t)$ for

designer parameters $\bar{k}$ and $\bar{\alpha}$. This establishes the exponential convergence of tracking errors

$$\xi(t) \to \xi_d(t) \quad \dot{\xi}(t) \to \dot{\xi}_d(t) \tag{37}$$

regardless of uncertainty of system parameters.

Let us now present the deduction of the desired attitude trajectories to satisfy Eq. 23.

In order to compute the desired Euler angles and angular velocity, let $\mathcal{T}_d$ be defined as the magnitude of u and $\mathcal{R}_d e_z$ as a unit vector, representing the direction, as follows

$$\mathcal{T}_d = \|u(t)\|, \ \mathcal{R}_d e_z = u(t)/\mathcal{T}_d$$

Then, solving the above relations for yaw $\psi_d = 0$, we obtain the following desired Euler angles

$$\theta_d = \arctan(u_1/u_3) \tag{38}$$

$$\phi_d = -\arcsin(u_2/\mathcal{T}_d) \tag{39}$$

where u_1, u_2, u_3 are the components of the control input u. The desired angular velocity $\boldsymbol{\omega}_d$ is deduced from the relationship between the Euler angles (and its derivatives) and the angular velocity $\boldsymbol{\omega}$, as follows

$$\omega_{1_d} = -(\dot{u}_2\mathcal{T}_d - \dot{\mathcal{T}}_d u_2)/\mathcal{T}_d\sqrt{u_1^2 + u_3^2} \tag{40}$$

$$\omega_{2_d} = \sqrt{1 - u_2^2/\mathcal{T}_d^2}\left(\frac{u_3\dot{u}_1 - \dot{u}_3 u_1}{u_1^2 + u_3^2}\right) \tag{41}$$

$$\omega_{3_d} = u_2(u_3\dot{u}_1 - \dot{u}_3 u_1)/\mathcal{T}_d(u_1^2 + u_3^2) \tag{42}$$

where $\dot{\mathcal{T}}_d = \frac{u_1\dot{u}_1 + u_2\dot{u}_2 + u_3\dot{u}_3}{\mathcal{T}_d}$ and the time derivative of the control $\dot{u} = (\dot{u}_1, \dot{u}_2, \dot{u}_3)$ is established from the fact that $\bar{S}_q = 0$ implies $\dot{\bar{S}}_q = 0$. Then, from Eqs. 32 and 36 $\dot{u} = \bar{K}_d\bar{\gamma}\text{sgn}(\bar{S}_q)$. $\mathbf{q}_d$ is obtained, from the desired Euler angles, by using the conversion between quaternion and Euler angles.

Notice that, Eqs. 38–42 are well-posed, since the third component of the control u_3 is counteracting the gravity, and therefore $u_3 > 0$ for all time.

5.3 Control Design for Terminal Stability

In order to implement the control law 36, the vectorial term $\mathcal{T}\mathcal{R}e_z$ must converge to u in finite-time. This is done by assuring finite-time convergence of attitude tracking errors through a desired rotation matrix $\mathcal{R}_d$ and thrust $\mathcal{T}_d$, computed from the control u. However, the result presented in Theorem 1 only provides exponential convergence of the attitude tracking errors, and does not guarantee finite-time convergence.

We now propose a new sliding surface, parameterized by a time base generator (TBG), based on [17], which moves and rotates continuously the nominal sliding surface through a known, state-independent, vanishing vector to achieve finite time convergence of tracking errors, with an arbitrary convergence time. This methodology can also be applied to the position control as will be shown in simulation results.

Consider the following first order differential equation

$$\dot{z} = -\rho(t)z \tag{43}$$

where

$$\rho(t) = \rho_0\frac{\dot{\chi}}{(1-\chi) + \delta} \tag{44}$$

with $\rho_0 = 1 + \varepsilon$, $0 < \varepsilon \ll 1$, and $0 < \delta \ll 1$. The time base generator $\chi(t) \in C^2$ must be provided by the user so as to χ goes smoothly from 0 to 1 in finite time $t = t_b > 0$, and $\dot{\chi}(t)$ is a bell-shaped derivative of χ such that $\dot{\chi}(t_0) = \dot{\chi}(t_b) \equiv 0$. Under these conditions, a solution of Eq. 43 is given by

$$z(t) = z(t_0)\left[(1-\chi) + \delta\right]^{1+\varepsilon} \tag{45}$$

with $\rho(t_b) > 0$. Note that t_b is independent of any initial conditions and hence

$$\rho(t_b) = 1 \Rightarrow z(t_b) = z(t_0)\delta^{1+\varepsilon} > 0$$

can be made arbitrarily small in arbitrarily finite time t_b. Thus, the key idea is to bring the solution of the attitude tracking errors to an equation similar to Eq. 45, defining $\rho_0 = \frac{1+\varepsilon}{q_{0e}}$. Notice that $q_{0e}(t)$ is positive and converges to 1, so ρ_0 is well possed.

Consider that the sliding mode is induced on $S_q(t) = 0$, for all time, and $S_d(t) \approx 0$ as has been proved in Appendix. Then, from Eq. 70, and replacing the gain α by 2ρ we obtain

$$\frac{d}{d\chi}q_e = -\rho_0 q_{0e}\frac{q_e}{(1-\chi) + \delta}$$

which attains the following solution

$$\begin{aligned} q_e(t) &= q_e(t_0)\left[(1-\chi) + \delta\right]^{1+\varepsilon} \\ &= q_e(t_0)\delta^{1+\varepsilon} \ \text{ at time } t = t_b \end{aligned}$$

since by assumption $\chi(t_b) = 1$ and by considering $q_{0e} \approx 1$. Considering that δ and ε are very small, then at $t = t_b$, tracking errors belong to a very small vicinity ε of the origin, which in practice may stand for the required precision or zero error. Note that at $t > t_b$, the time varying feedback gain $\rho(t)$ must be reset to the desired constant $\alpha > 0$. Thus, convergence of the attitude tracking errors are guaranteed in finite-time.

6 Control Design of Force for Stable Grasp

Based on [20], which establishes the model and computation for internal forces at a static equilibrium, [22] proposes a multi-grasp manipulation scheme for robot arms that account for a stiffness control, that is, force model is computed according to the restitution force of the Joule model. In this way, force arises because of the compression ratio of virtual springs that connect each finger with the object and the object with the target point. This is interesting for our problem, because it indicates that it is possible to design a grasping matrix to enforce stable grasp using a simple stiffness control as the gradient of the potential energy of the virtual springs. To accommodate this new torque control term, the position-orientation controller proposed in the previous section will compensate for as if this torque stands for a torque disturbance.

6.1 System Model and Assumptions

Consider without loss of generality that $j = 4$, that is modeling 4 Quadrotors to grasp a rigid object. Then, the following Euler-Lagrange model can be derived for the multi-body system,

$$M_r(x_r)\ddot{x}_r + C_r(x_r, \dot{x}_r)\dot{x}_r + g_r(x_r) = w_{\text{ext}} + w_f$$

$$M_q(x_q)\ddot{x}_q + C_q(x_q, \dot{x}_q)\dot{x}_q + g_q(x_q) = \tau - \tau_f + \tau_{\text{ext}}$$

where $x_q = (x_{f_1}^T, ..., x_{f_N}^T)^T \in \mathbb{R}^M$ is the vector of generalized positions for N Quadrotors. The vector $\tau \in \mathbb{R}^{M-2N}$ contains the corresponding generalized control inputs, that is three for orientation and the thrust in z axis for each Quadrotor. The vector $x_r \in \mathbb{R}^6$ is the local representation of the object frame $H_r \in SE(3)$. Consider that indexes q and r stand for Quadrotor and object, respectively, then $M_q(x_q) \in \mathbb{R}^{M\times M}$ and $M_r(x_r) \in \mathbb{R}^{6\times 6}$ stand for the symmetric and positive definite inertial matrices, $C_q(x_q, \dot{x}_q) \in \mathbb{R}^M$ and $C_r(x_r, \dot{x}_r) \in \mathbb{R}^6$ contain the centripetal and centrifugal forces in terms of the Coriolis components, and $g_q(x_q) \in \mathbb{R}^M$ and $g_r(x_r) \in \mathbb{R}^6$ are the vectors of generalized gravity forces. Finally, $w_{\text{ext}} \in \mathbb{R}^6$ contains the external generalized wrench acting on the object, and then $w_f \in \mathbb{R}^6$ models the wrench applied to the object at the contact points by the Quadrotors. Vector τ_f represents the generalized torque inputs due to the contact forces in the Quadrotors. In order to focus on the presentation of the multi-Quadrotor part of the dynamics, other physical effects like link and joint flexibility as well as joint friction are neglected reasonably because it is assumed frictionless contact and rigid frames.

The following definitions are used to facilitate the notation. A frame $H_{x,y} = [R_{X,Y}, p_{X,Y}] \in SE(3)$, consisting of a rotation matrix $R_{X,Y} \in SO(3)$ and a translation $p_{X,Y} \in \mathbb{R}^3$, transforms a given position in the coordinate system Y into the coordinate system X. If only one index is used, the coordinate system X is considered as an inertial coordinate system, that is, $H_y = H_{I,Y}$. A frame H_X can be described by a local parametrization $x_X \in \mathbb{R}^6$. Vector $w_X = (f_X^T, m_X^T)^T$ contains the generalized forces and moments acting at the origin of the coordinate system H_x, with $f_x, m_x \in \mathbb{R}^3$ represented in the body frame [16].

In the following, fine manipulation with multiple Quadrotors is treated. Assume $N > 2$ and a manipulable grasp map [16], for a $3D$ object; furthermore, consider the following assumptions

1. Internal forces are sufficiently large such that the friction constraints are fulfilled for all contact points, that is no sliding arises at any contact point.
2. The contact between the object and the Quadrotors is restricted at one point.
3. The relative contact points between the fingertips and the object do not change (neglecting rolling effects). This assumption introduces a relatively small kinematic error to maintain consistently a holonomic constraint.

4. The magnitude of internal forces ensures a stable grasp for any reachable pose within the available Quadrotors' thrust.
5. The corresponding internal forces that guarantee object manipulation are inside the subspace of admissible Quadrotor pose during the contact with the object.
6. There is a point contact with friction, that is, only forces and no torques can be transmitted at the contact point. There are only constraints in position, but not in orientation, it means that the Quadrotor exerts only forces to the object, but not moments and the Quadrotor can change its orientation without directly affecting the orientation of the object.

Now consider forces f_{fi}, the velocities $\dot{p}_{fi}$, and the variations of position $\delta p_{fi}(x_{fi})$ of the Quadrotors in stacked notation, and that the i-th Cartesian Quadrotor position $p_{fi}(x_{fi}) \in \mathbb{R}^3$ and its orientation relative to the inertial frame $R_{fi}(x_{fi})$ can be calculated as a function of the generalized positions x_{fi}. To indicate stacked notation the index is removed [e.g. $f_f = (f_{f,1}^T, ..., f_{f,N}^T)^T$]. Note that stacked variables are expressed in the inertial frame, then the Quadrotors' stacked jacobian $J_q(x_q) = \frac{\partial p_f(x_q)}{\partial x_q}$ maps the Quadrotor velocities to the inertial frame. Assumptions (4) and (5) ensure that $J_q^{-1}(x_q)$ exists and is well-posed. In this condition, the grasp map $G_r \in \mathbb{R}^{6\times 3N}$ is used to determine the effect of the stacked contact forces at the Quadrotors' contact points $f_f \in \mathbb{R}^{3N}$ on the object wrench w_f, with

$$\begin{aligned} G_r &= [Ad^T_{H^{-1}_{r,f_1}} B, ..., Ad^T_{H^{-1}_{r,f_N}} B] R_f^T, \\ R_f &= \text{blockdiag}(R_{f_1}, ..., R_{f_N}), \\ B &= [I_{3\times 3} 0_{3\times 3}]^T, \end{aligned} \tag{46}$$

where $B \in \mathbb{R}^{6\times 3}$ is the wrench basis, and $Ad^T_{H_{x,y}}$ represents the adjoint transformation associated with $H_{x,y}$ [16], given by

$$Ad_{H_{x,y}} = \begin{bmatrix} R_{x,y} & \hat{p}_{x,y} R_{x,y} \\ 0 & R_{x,y} \end{bmatrix}^T, \tag{47}$$

with the skew-symmetric operator $\hat{w} = w^\times : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$, and H_{r,f_i} is the configuration of the i-th contact frame relative to the object frame. Rotation $R_{f_i}^T$ transforms the forces at the fingertips f_f represented in the inertial system into the i-th contact frame. The grasp map relates not only forces, but also velocities and variations at the Cartesian fingertip level, with the ones at the object level as follows

$$w_f = G_r f_f, \qquad G_r^T \dot{x}_r = \dot{p}_f, \qquad G_r^T \delta x_r = \delta p_f$$

Now, the well-known grasp constraint can be formulated as follows

$$J_q(x_q)\dot{x}_q = G_r^T(x_q, x_r)\dot{x}_r \tag{48}$$

6.2 Implementation of Impedance Behaviors

The impedance behaviors are based on the following control law defined as

$$\tau_o = -\mathcal{TR}e_z - \left(\frac{\partial V(x_q)}{\partial x_q}\right)^T \tag{49}$$

where $\mathcal{TR}e_z$ represents the virtual control on the Quadrotor position for hovering at a desired altitude z_d. The overall potential function $V(x_q)$ models the environmental potential energy from the virtual springs connecting the Quadrotors with the object at each contact point. Clearly, $V(x_q)$ stands for the impedance behaviors according to [22], consisting of the sum of the individual potential functions of all the springs components. In our case, since the controller τ_q guarantees an invariant manifold $S_q = 0$ in closed-loop for tracking of admissible position and orientation, the control input τ_o creates the restitution forces of such springs.

6.3 Inversion of the Grip Map

Most grasp controllers presented in the literature are based on the inversion of the grip map Q [13]. To obtain it, consider the grasp map $G(p_f(x_q), x_r)$ stacked together with its orthogonal complement $E(p_f(x_q))$, which spans its null space. The matrix $E(p_f(x_q))$ can be modeled using the virtual linkage [20]. To this end, consider task states x as follows

$$x = \begin{pmatrix} x_r \\ x_\eta \end{pmatrix}, \ \Delta x = x - x_d, \ K_x = \begin{bmatrix} K_r & 0 \\ 0 & K_\eta \end{bmatrix} \tag{50}$$

where x_r stands for the coordinates of the object and $x_\eta \in \mathbb{R}^{(N-1)N/2}$ is the coordinates of the virtual

linkage, with the i-th distance between fingertips $x_{\eta,i} = \| p_{f,j} - p_{f,k} \|$. The coefficients j, k cover all possible fingertip connections, where $x_d = (x_{r,d}^T, x_{\eta,d}^T)^T$ models its corresponding desired values where full rank of $G(p_f(x_q), x_r)$ occurs. As suggested by [22], such a grip map-based controller can be obtained as follows. Consider a potential function that describes the energy of an object-level spring, corresponding to internal forces, clearly a direct choice is

$$V(\Delta x) = \frac{1}{2} \Delta x^T K_x \Delta x \tag{51}$$

although more interesting and physical meaningful restitution forces can be modeled with nonlinear saturated springs. Then, the compliance control law 49 is nothing but

$$\begin{aligned} \tau_o &= \left(\frac{\partial V(\Delta x)}{\partial \Delta x} \frac{\partial \Delta x}{\partial p_f} \frac{\partial p_f}{\partial x_q} \right)^T \\ &= J_q^T \left(\frac{\partial \Delta x}{\partial p_f} \right)^T K_x \Delta x, \end{aligned} \tag{52}$$

using the local transformation

$$\frac{\partial p_f}{\partial x} = \left[\frac{\partial p_f}{\partial x_r} \frac{\partial p_f}{\partial x_\eta} \right] \equiv Q^T. \tag{53}$$

Herein, using the local properties, $G_r^T(p_f(x_q), x_r) = \frac{\partial p_f}{\partial x_r}$ can be recognized as the grasp map, and the inversion map $E(p_f) := \frac{\partial x_\eta}{\partial p_f}$ can be determined by the virtual linkage [22].

For a four-fingered hand, each entry of $E(p_f)$ can be computed using the unit vector pointing from fingertip k to j, that is, $e_{j,k} = (p_{f,j} - p_{f,k}) / \| p_{f,j} - p_{f,k} \|$, then $E(p_f)$ becomes

$$E = \begin{bmatrix} e_{12} & e_{13} & e_{14} & 0 & 0 & 0 \\ -e_{12} & 0 & 0 & e_{23} & e_{24} & 0 \\ 0 & -e_{13} & 0 & -e_{23} & 0 & e_{34} \\ 0 & 0 & -e_{14} & 0 & -e_{24} & -e_{34} \end{bmatrix}, \tag{54}$$

where the equivalence between the virtual linkage E as proposed in the literature and our potential-function-based derived mapping is discussed in Appendix. In this way, the virtual linkage E is orthogonal to the grasp map, that is, $G_r E = 0$.

Consequently, the choice of coordinates for the virtual linkage combined with the grasp is such that the matrix Q is square with full rank if the grasp is stable and the virtual linkage establishes a non-degenerate map. In this case, the inverse of the grip map arises as follows

$$Q^{-1} = \left(\frac{\partial \Delta x}{\partial p_f} \right)^T = [G_r^+ E]. \tag{55}$$

In [20], the Moore-Penrose pseudoinverse of the grasp map G_r is used, furthermore, [4] shows that a weighted pseudoinverse, the object wrench is in the range space of the grasp map $K_r x_r \subset R(G_r)$ and the controlled fingertip forces $f_f = J_q^{-T} \tau_o$ have consistent physical units, providing a way to introduce kernel control schemes, if necessary.

Since $\dot{x}_r$ cannot be measured directly, the inverse of the transposed grasp map is used to derive it based on the velocities of the fingertips

$$\dot{x}_r = G_r^{T+} \dot{p}_f \tag{56}$$

Furthermore, according to [21], integrating Eq. 56 yields x_r, respectively $H_r(x_r)$, instead of observing the real-object dynamics, certainly a viable option in real implementations, as long as initial conditions are available for the integration process.

To obtain the stiffness properties, it is assumed that the controller compensates for object gravitational forces, and the object Coriolis forces are neglected; object dynamics can be analyzed in virtual coordinates, that is

$$M_x \ddot{x} = f_x + f_{x,\text{ext}} \tag{57}$$

with $f_x = Q f_f$, the generalized external force $f_{x,\text{ext}} = (w_{\text{ext}}^T, f_{\eta,\text{ext}}^T)^T$, and $f_{\eta,\text{ext}}$ the external force related to the coordinates x_η. The inertia matrix $M_x = blockdiag(M_r, 0)$ since the object does not have any inertia *w.r.t.* to internal motions. Now since $t_f = J_q^T f_f$, dynamics can be written as

$$M_q \ddot{x}_q = -J_q^T f_f + \tau \tag{58}$$

with $\dot{x} = Q^{-1} J_q \dot{x}_q$, and the acceleration constraint $\ddot{x} = Q^{-T} J_q \ddot{x}_q + \frac{d}{dt}(Q^{-T} J_q) \dot{x}_q$, in which the second term can be interpreted as a Coriolis term, neglected in the damping design. Pre-multiplying Eq. 58 by $Q^{-T} J_q M_q^{-1}$, and using $f_f = Q^{-1} f_x$ and $\ddot{x}$, we obtain

$$\ddot{x} = -Q^{-T} J_q M_q^{-1} J_q^T Q^{-1} f_x + Q^{-T} J_q M_q^{-1} \tau. \tag{59}$$

Now, multiplying Eq. 59 with $M_q^x = (Q^{-T} J_q M_q^{-1} J_q^T Q^{-1})^{-1}$, the Quadrotors inertia

matrix represented in generalized coordinates, it arises

$$M_q^x \ddot{x} = -f_x + M_q^x Q^{-T} J_q M_q^{-1} \tau. \tag{60}$$

Solving for f_x, and using the above equation

$$(M_x + M_q^x)\ddot{x} = M_q^x Q^{-T} J_q M_q^{-1} \tau + f_{x,\text{ext}}. \tag{61}$$

where $\tau = \tau_q + \tau_o$ is the total control law. Finally, the simplified closed-loop dynamics can be written as

$$(M_x + M_q^x)\ddot{x} + K_x x = -M_q^x Q^{-T} J_q M_q^{-1} K_d S_r + f_{x,\text{ext}} + K_x x_d \tag{62}$$

6.4 The Full Controller

The control $\tau = \tau_q + \tau_o$ for stable grasping yields $K_x x_d$ as an additional endogenous controlled force.

7 Simulation Study

7.1 Simulator Setup

A modular simulator is programmed using Matlab® and Simulink® V10.2, on a Windows 7 personal computer equipped with Intel Core I7 first generation, with 4 Gb of DRAM. It can be distinguished six modules, four for the 4 Quadrotors dynamics, one for the object interaction, and another for the grasp matrix calculations. Module means an embedded function block of Simulink.

Two cases are considered, the first one is grasping, that is, immobilize the object from different initial conditions of each Quadrotor. The second simulation considers grasp and then manipulation, that is controlling the pose of the object. In both case, terminal stability is considered to reach and touch the object exactly at the same time by all Quadrotors.

Table 1 Quadrotors' initial conditions

Quadrotor 1	Quadrotor 2	Quadrotor 3	Quadrotor 4
$x_0 = 0$	$x_0 = 2.5$	$x_0 = 2.5$	$x_0 = 0$
$y_0 = 0$	$y_0 = 0$	$y_0 = 2.5$	$y_0 = 2.5$
$z_0 = 0$	$z_0 = 0$	$z_0 = 0$	$z_0 = 0$

Table 2 Desired trajectories

Quadrotor 1	Quadrotor 2	Quadrotor 3	Quadrotor 4
$x_{1d} = \varrho + \frac{1}{2}$	$x_{2d} = -\varrho + 2$	$x_{3d} = -\varrho + 2$	$x_{4d} = \varrho + \frac{1}{2}$
$y_{1d} = \varrho + \frac{1}{2}$	$y_{2d} = \varrho + \frac{1}{2}$	$y_{3d} = -\varrho + 2$	$y_{4d} = -\varrho + 2$
$z_{1d} = \varrho + \frac{1}{2}$	$z_{2d} = \varrho + \frac{1}{2}$	$z_{3d} = \varrho + \frac{1}{2}$	$z_{4d} = \varrho + \frac{1}{2}$

with $\varrho = 0.5 \tanh(4t - 3.4)$

7.2 Conditions, Desired Trajectories, Parameters and Feedback Gains

7.2.1 Conditions

In the first case, initial conditions for each 4 Quadrotor starts at different locations, see Table 1; then, the controller τ_q drives them in finite time to grab a cylindrical plate and immobilizing it. The object has a plate shape of a diameter $d_o = 0.7$ m, a height $h_o = 0.05$ m, a mass $m_o = 0.4$ Kg and is placed motionless at $z = 1$ m of altitude.

7.2.2 Desired Trajectories

Desired references are shown in Table 2.

7.2.3 Control Gains

Feedback gains for all Quadrotors are equally tuned independent of different reference signals and different initial conditions. Desired time to grasp the object is at 1.88 s, see Table 3.

7.3 Simulations Results

7.3.1 Case 1: Grasping Only

Convergence of position trajectories stands for the synchronization of all Quadrotors, according to the task, as it can be seen in Figs. 2 and 3.

Table 3 Control gains

Attitude control gains	Position control gains
$k_d = \text{diag}(10, 10, 10)$	$\bar{k}_d = \text{diag}(26, 26, 23)$
$\gamma = \text{diag}(1, 1, 1)$	$\bar{\gamma} = \text{diag}(0.25, 0.25, 0.8)$
$\alpha = \text{diag}(25, 25, 25)$	$\bar{\alpha} = \text{diag}(11, 11, 11)$
$s(t_0) = \text{diag}(0, 0, 0)$	$s(t_0) = -0.0089 I_{3\times 3}$
$k = 5$	$k = 5$
$t_b = 0.5$	$t_b = 1.88$

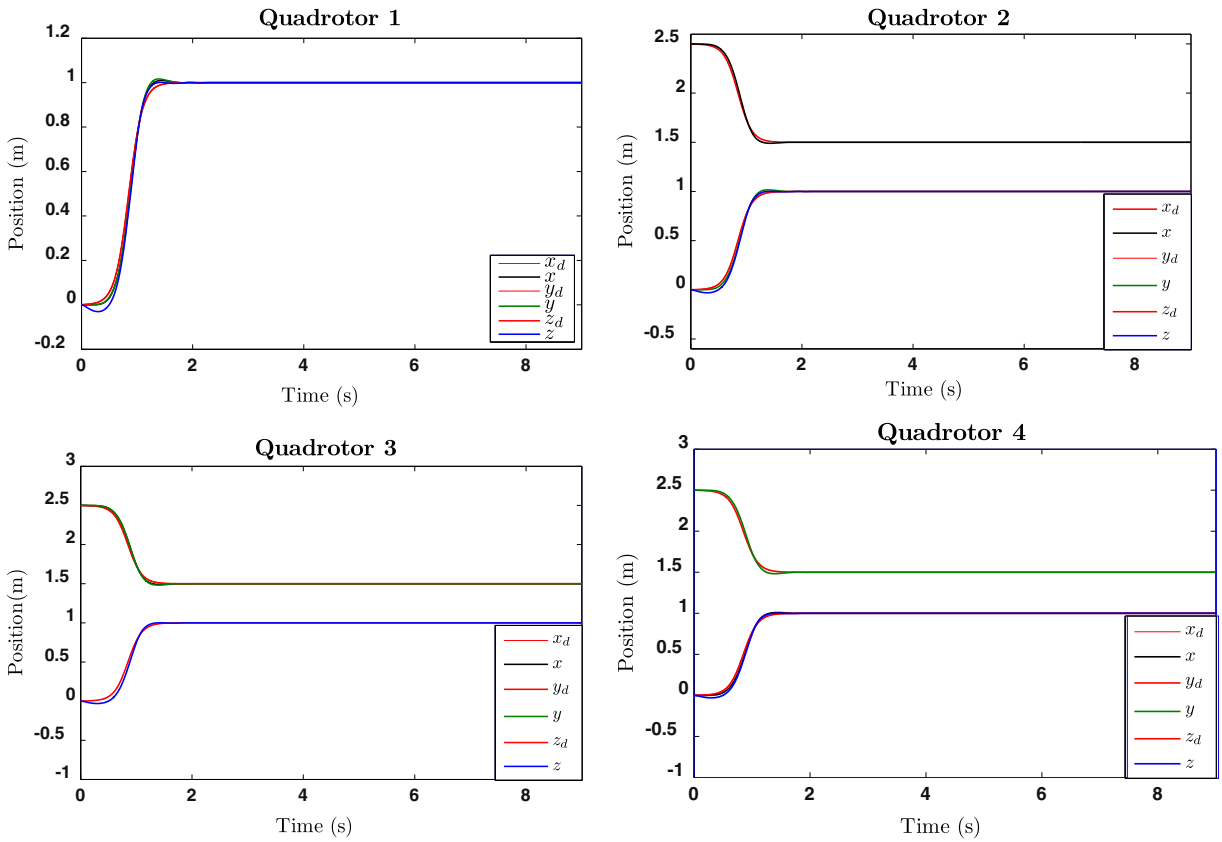

Fig. 2 Exponential position tracking

Figure 4 shows the force tracking that ensures exerting of a planned profile that complies to a stable grasp of all the Quadrotors starting at time $t = t_b$.

Three dimensional contact forces allows the exertion of predefined and necessary forces to establish a grasp map that enforces stable grasp.

Figure 5 shows how the grip map does not lose rank at any moment once the Quadrotors have grabbed the object, meaning that stable grasp is always obtained.

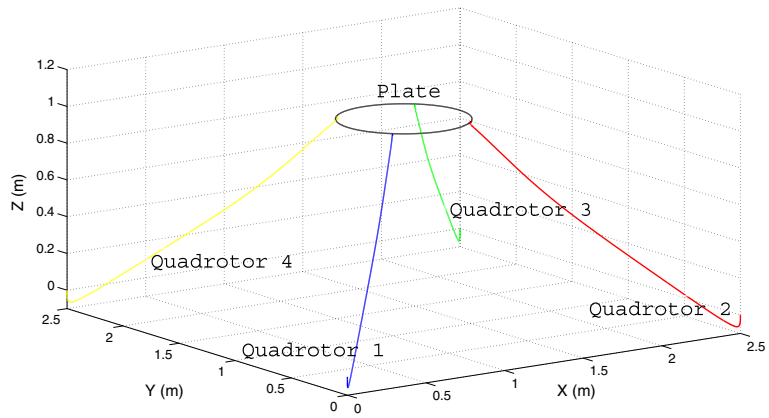

Fig. 3 Representation in 3D of the 4 Quadrotors and the object

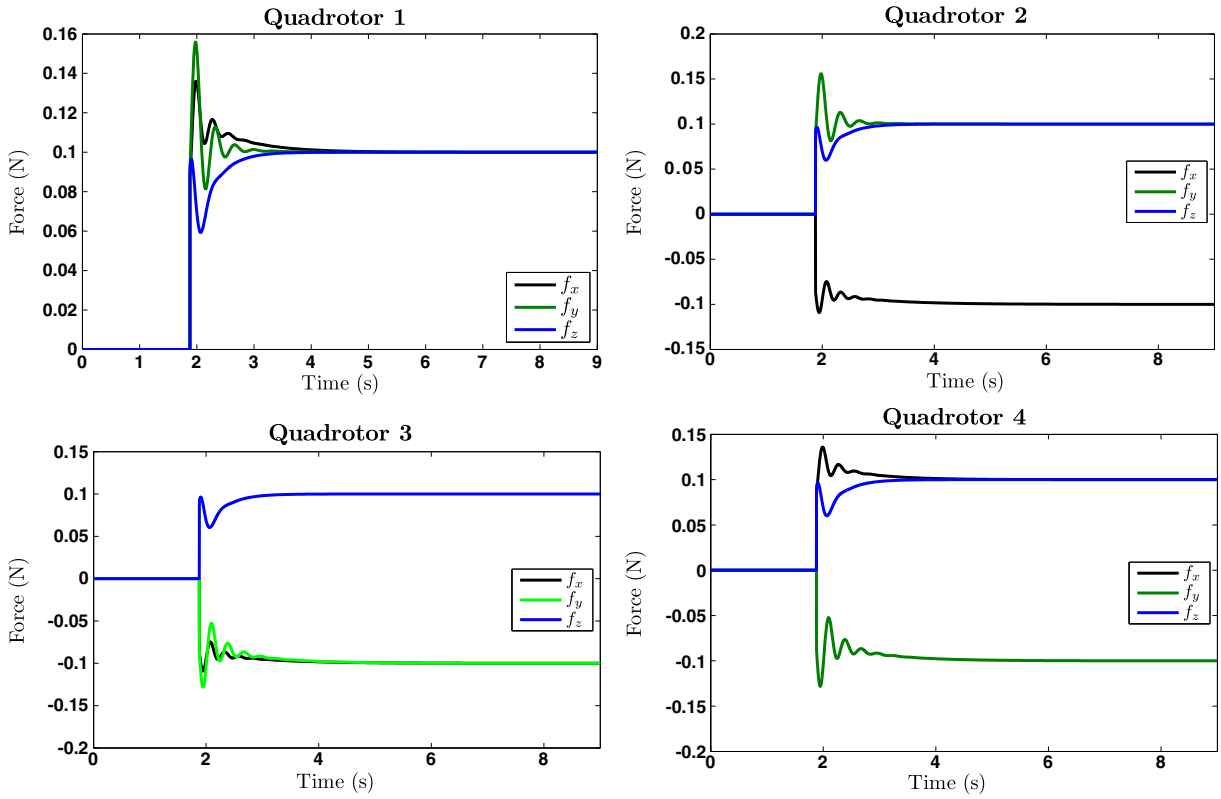

Fig. 4 Force applied to the object

7.3.2 Case 2: Grasping and Manipulation

This involves the whole case 1, complemented with controlling the pose of the object, that is, manipulation using slight different feedback gains, see Table 3, with same initial conditions. Once stable grasp, the object is taken up and down over and over, and desired trajectories are achieved. For the second case of study, the reference signals were changed after the grabbing of the plate and the objective is to move the plate up and down, see Fig. 6. In consequence, the reference signals are divided and chosen for a certain time; the reference signals until $t = 4$ s are shown in Table 4.

After $t = 4s$, manipulation starts with the trajectories defined in Table 5.

Results indicate that Quadrotors lift up and down the object, and stable grasp has been achieved, see Fig. 6. Evidence of this is the ranks of matrices Q, G_r and E which maintain rank during the whole manipulation regime, see Fig. 7.

8 Remarks

8.1 On the Control System

The closed-loop system shows fast and robust tracking without any knowledge of dynamic model of any Quadrotor nor the object. The controller is quite simple to implement it, while the theoretical proof is not that easy. It involves advanced arguments of Lyapunov stability, terminal stability, variable structure control and high order sliding modes, as well as structural properties of the Quadrotors.

So, this makes more interesting the control structure since the computational cost to implement this controller is very low, demanding low

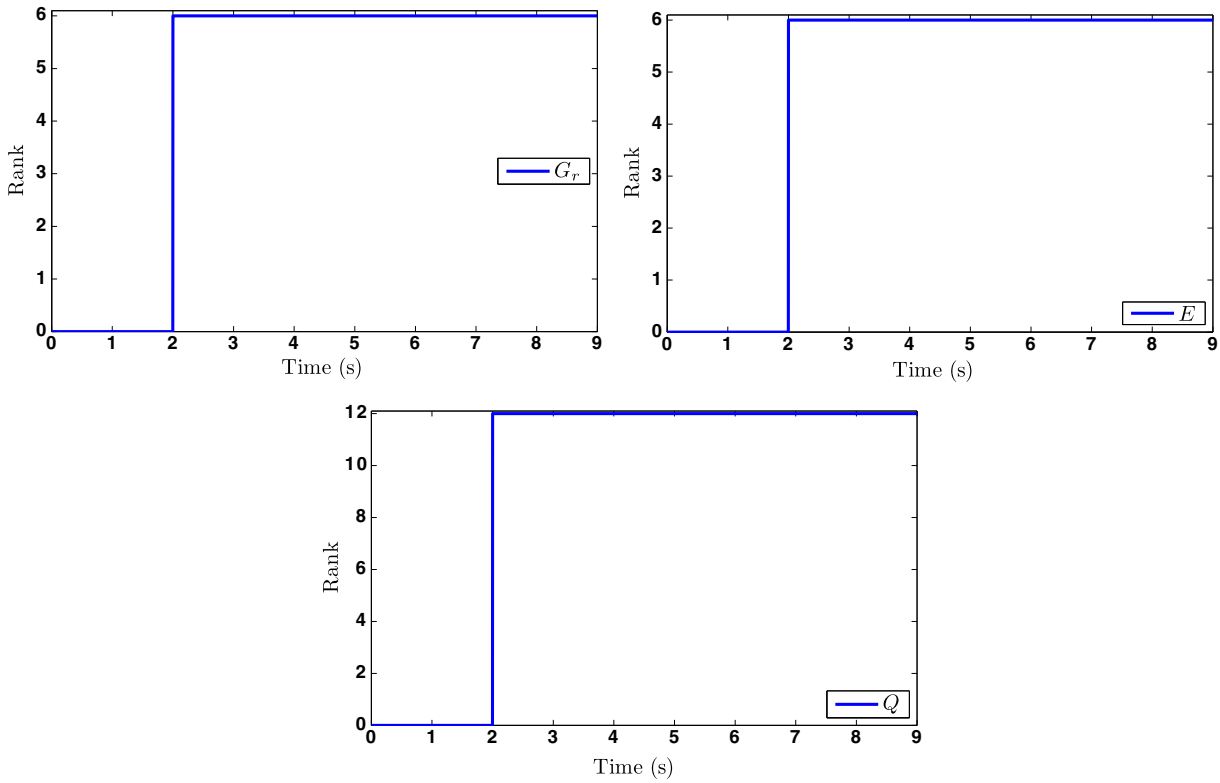

Fig. 5 Computation of rank(G_r), rank(E), rank(Q), showing a stable grasp at time of grabbing and during manipulation

computational resources, although full state feedback is required. No acceleration is needed, and no approximation of any sort has been assumed on the controller of system dynamics.

Our approach requires full state feedback, which limits its scope for practical applications, as any scheme that assumes it. The challenging and relevant problem of partial feedback stabilization or tracking with velocity estimation is not studied in this paper. For instance, an alternative approach for the regulation case is proposed in [3]. However, the Quadrotor possesses a typical

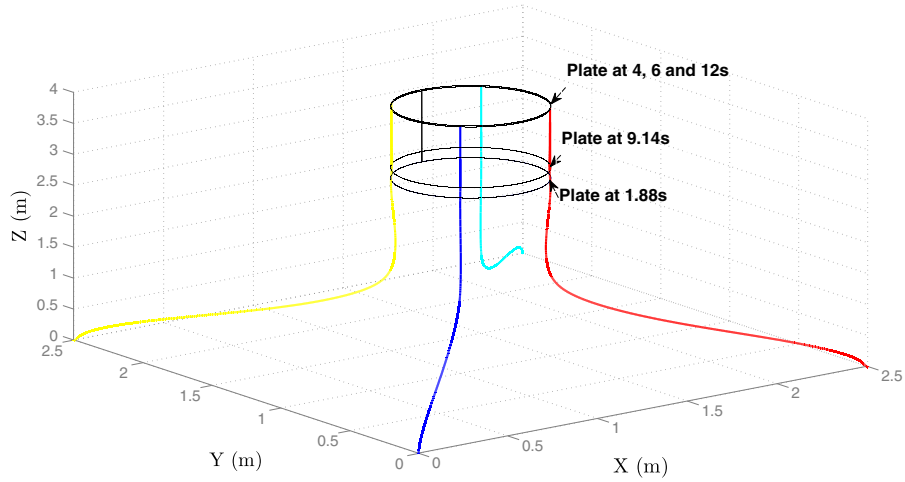

Fig. 6 Manipulation of 3D object with 4 Quadrotors

Table 4 Reference signals

Quadrotor 1	Quadrotor 2	Quadrotor 3	Quadrotor 4
$x_{1d} = \varrho + \frac{1}{2}$	$x_{2d} = -\varrho + 2$	$x_{3d} = -\varrho + 2$	$x_{4d} = \varrho + \frac{1}{2}$
$y_{1d} = \varrho + \frac{1}{2}$	$y_{2d} = \varrho + \frac{1}{2}$	$y_{3d} = -\varrho + 2$	$y_{4d} = -\varrho + 2$
$z_{1d} = \varsigma + 2$	$z_{2d} = \varsigma + 2$	$z_{3d} = \varsigma + 2$	$z_{4d} = \varsigma + 2$

with $\varrho = 0.5\tanh(4t - 3.4)$ and $\varsigma = 2\tanh(1.3t - 2)$

structure of an Euler Lagrange system (with virtual inputs) which makes plausible to develop a deterministic estimator for position and orientation dynamics. We are studying this problem based on our previous work [11].

8.2 On the Interaction Process

Interaction forces couple the whole system since object and UAVs are considered composed by rigid frames. However, our approach considers only one type of interaction forces, the contact forces, and leave out forces arising from rigid interaction of UAV i to UAV m. A passive impedance attaching device [23], may be an effective option to accommodate smoothly such interaction with a second order filter, then such force is treated as a smooth external bounded force. Another option to compensate actively for the interaction forces, similar to [24], is to model the interaction as a constrained *velocity*, that is the velocity projected into each force subspace $J_\varphi(q)$ spanned by the gradient $\frac{\partial \varphi}{\partial q}$ of the kinematic model of the object, written in terms of an implicit equation $\varphi(q) = 0$. This would stand for an active compensation of interaction forces, a more promising scheme that would couple models and controllers only in terms of this $v_i = \sum_i^j J^T_{\varphi_i(q_i)} \dot{q}_j$. In this case, clearly for rigid interaction it had that $\sum_i^j v_i = 0$, then $\sum_i^j \Delta v_i = 0$, for $\Delta v_i = v_i - v_{di}$, and $v_{di} = J^T_{\varphi_i(q_{di})} \dot{q}_j$. This last approach has been proved successful in multirobot arms for cooperative manipulation [24].

Table 5 Desired trajectories after $t = 4$ s

Quadrotor 1	Quadrotor 2	Quadrotor 3	Quadrotor 4
$x_{1d} = \varrho + \frac{1}{2}$	$x_{2d} = -\varrho + 2$	$x_{3d} = -\varrho + 2$	$x_{4d} = \varrho + \frac{1}{2}$
$y_{1d} = \varrho + \frac{1}{2}$	$y_{2d} = \varrho + \frac{1}{2}$	$y_{3d} = -\varrho + 2$	$y_{4d} = -\varrho + 2$
$z_{1d} = \rho + \frac{7}{2}$	$z_{2d} = \rho + \frac{7}{2}$	$z_{3d} = \rho + \frac{7}{2}$	$z_{4d} = \rho + \frac{7}{2}$

with $\varrho = 0.5\tanh(4t - 3.4)$ and $\rho = 0.5\cos(t - 6)$

8.3 What We Learned on the Simulation Study

Exponential tracking is achieved in position coordinates to synchronize all Quadrotors, so as to they meet at a given peripheral point onto the object at a given precise time. Grasp occurs and then manipulation happens. Simulations are carried out in rather ideal conditions since neither gust wind nor cross (from one Quadrotor to the other) wind are considered. Anyway, the controller considers such bounded external disturbances, but clearly as long as no aerodynamical cross gust happens.

Nevertheless, if a disturbance strong enough were to happen, the object will be thrown by the Quadrotors due to the lack of a cooperative interaction control design with each other since our controller is not cooperative. For cooperation, an additional control variable must be included to convey a variable that codes a *cooperative behavior*, for instance see [24].

8.4 The Virtual Linkage Approach

In some cases, the virtual linkage can degenerate, however for the Quadrotors case we are interested in those configurations corresponding to not feasible poses where flight control cannot happen. For instance, when all contact points are coplanar or more than two points lie on a common line. Excluding such defunct configurations with assumed good motion planning, E does not lose row rank according to [21]. Thus, it poses no problem to the definition of the stiffness properties of the virtual linkage.

8.5 On the Device for Contact

Active and passive devices have been proposed for grabbing and gripping an aerial object, as well as simple devices for load carrying. They argue that such devices allow the poor position precision observed in UAV with simple controllers. Although the passive controlled and variable impedance device proposed in [23] may accom-

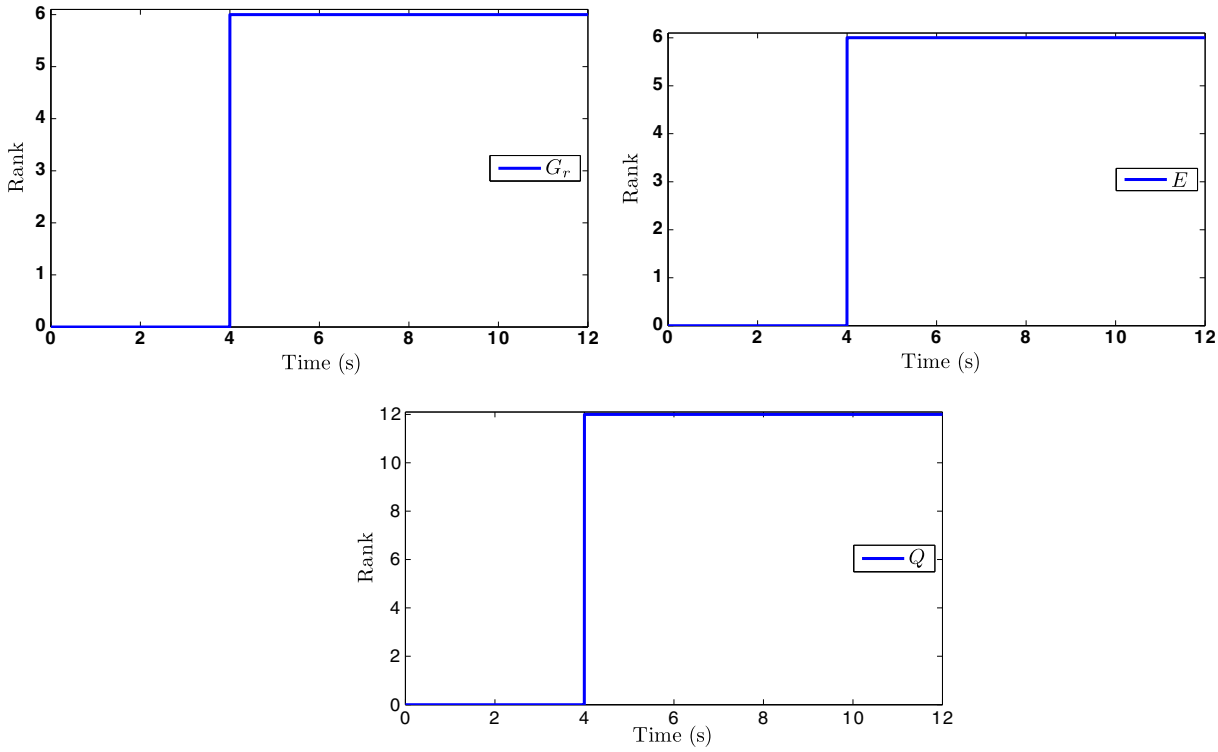

Fig. 7 Rank of matrices G_r, E and Q indicating stable grasp and manipulation of the object

modate poor position tracking of the UAV, it is true that sub-millimeter position tracking can be achieved when a more complete dynamic model and advanced controller are used [8]. In our approach, it is clear that since we assume that the object is grasped at the same time at given contact points by j-UAVs, it is required the best possible position tracking attainable. The device proposed in [23] may accommodate passively the force sensor within the object, so as to measure contact force $f_c \in \mathbb{R}^3$ at each j-contact points with a passive 3D ball joint. In this way, the object should be instrumented with j wireless 3D force sensors.

9 Conclusions

A control scheme for *aerial grasp manipulation* with a j-Quadrotor system is proposed. Position and orientation dynamics are considered for the design of the controller, which enforces a second order sliding mode for fast and robust tracking of admissible trajectories. Firstly, each Quadrotor's dynamics, with possible different initial conditions and different parameters, reaches a given spatial point onto the object at the very same time to grasp the object. Then, when stable grasp is verified, the virtual linkages approach ensures rigid grasping and the manipulation can be carried out, with an object-level stiffness control term. It is assumed stable contact points and admissible object trajectories. Clearly, implementation of this approach suggest a passive gripper for attaching and detaching, for instance, distance sensors within the object would determine that all UAVs are ready for contact to launch the attaching. The numerical simulation study allows the detection of a number of issues to be analyzed. It suggests that a new dynamic model is required such that the j-Quadrotor system is modeled as a unique super-system, such that the whole system behaves as a super-Quadrotor system, with the object in common.

Appendix

Proof of Theorem 1 Substituting Eq. 22 into Eq. 26 yields

$$\mathbf{J}\dot{S}_r = -\left(K_d S_r + S_r^{\times}\mathbf{J}S_r\right) + d(t) - Y_r \tag{63}$$

Let us consider the following Lyapunov function

$$V = \frac{1}{2}S_r^T\mathbf{J}S_r \tag{64}$$

The total derivative of Eq. 64 along its solution 63 gives rise to

$$\begin{aligned}\dot{V} &= -S_r^T K_d S_r + S_r^T\left(d(t) - Y_r\right)\\ &\leq -\|S_r\|\left(\lambda_{\min}(K_d)\|S_r\| - \eta(t)\right)\end{aligned} \tag{65}$$

Let $c = \sup_{t\geq 0}\eta(t)$. Note that if $\|S_r\| > (c/\lambda_{\min}(K_d))$ then $\dot{V} < 0$. This implies that exists a time t_1 such that

$$\|S_r\| \leq \frac{c}{\lambda_{\min}(K_d)} \quad \forall t > t_1 \tag{66}$$

In this way S_r is upper bounded by $c/\lambda_{\min}(K_d)$. Boundedness of S_r implies the boundedness of the state which includes σ. Therefore, we can conclude that $\dot{S}_r \leq \bar{c}$ for some real $\bar{c} > 0$.

Now, we will show that for a given γ, sliding mode is induced on $S_q = 0$. Consider the following dynamical system defined by Eq. 19

$$\dot{S}_q = -\gamma\,\mathrm{sgn}(S_q) + \dot{S}_r \tag{67}$$

with the following positive definite function

$$V_q = \frac{1}{2}S_q^T S_q \tag{68}$$

The total derivative of Eq. 68, along its solution 67 give rise to

$$\begin{aligned}\dot{V}_q &= -S_q^T\gamma\,\mathrm{sgn}(S_q) + S_q^T\dot{S}_r\\ &\leq -\left(\lambda_{\min}(\gamma) - \sqrt{3}\bar{c}\right)|S_q|\end{aligned}$$

Thus, in order to prove that $S_q \to 0$ in finite time, we can always choose $\nu = \lambda_{\min}(\gamma) - \sqrt{3}\bar{c} > 0$ which guarantees the existence of a sliding mode condition. This implies that a sliding mode is established at time $t_s \leq \left(|S_q(t_0)|/\nu\right)$, and since $S_q(t_0) = 0$ for any initial condition, then the sliding mode in $S_q(t) = 0$ is enforced for all time.

Considering that $S_d \approx 0$, then

$$\boldsymbol{\omega}_e = -\alpha q_e \tag{69}$$

Introducing Eq. 69 into the time derivative of $\mathbf{q}_e$ yields

$$\dot{\mathbf{q}}_e = \begin{pmatrix}\frac{\alpha}{2}q_e^T q_e\\ -\frac{\alpha}{2}q_{0e}q_e\end{pmatrix} = \begin{pmatrix}f_1(t)\\ -f_2(t)q_e\end{pmatrix} \tag{70}$$

where $f_1(t)$ is a positive-definite function and $f_2(t)$ is a positive-definite function, provided that $q_{0e}(t_0) > 0$, a simple and yet practical constraint easy to meet for small errors. Then the solution of Eq. 70 is given by

$$q_{0_e}(t) = q_{0_e}(t_1) + \int_{t_1}^{t} f_1(t) \tag{71}$$

$$q_e(t) = q_e(t_1)e^{-f_2(t)(t-t_1)} \tag{72}$$

Then, by using the constraint of a unit quaternion 10, exponential convergence of q_e and q_{0e} is achieved. This completes the proof of Theorem 1.

References

1. Akella, P., Parra-Vega, V., Arimoto, S., Tanie, K.: Discontinuous model-based adaptive control for robots executing free and constrained tasks. In: Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 3000–3007 (1994)
2. Bernard, M., Kondak, K.: Generic slung load transportation system using small size helicopters generic slung load transportation system using small size helicopters. In: 2009 IEEE International Conference on Robotics and Automation, pp. 3258–3264 (2009)
3. Bertrand, S., Guenard, N., Hamel, T., Piet-Lahanier, H., Eck, L.: A hierarchical controller for miniature VTOL UAVs: design and stability analysis using singular perturbation theory. Control Eng. Pract. **19**, 1099–1108 (2011)
4. Bonitz, R.G., Hsia, T.C.: Force decomposition in cooperating manipulators using the theory of metric spaces and generalized inverses. In: Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 1521–1527 (1994)
5. Erdong, J., Zhaowei, S.: Robust controllers design with finite time convergence for rigid spacecraft attitude tracking control. Aerosp. Sci. Technol. **12**, 324–330 (2008)
6. Etkin, B., Reid, L.D.: Dynamics of Flight. Wiley, New York. ISBN 0471034185 (1959)

7. Fink, J., Michael, N., Kim, S., Kumar, V.: Planning and control for cooperative manipulation and transportation with aerial robots. Int. J. Rob. Res. **30**(3), 324–334 (2011)
8. Ghadiok, V., Goldin, J., Ren, W.: Autonomous indoor aerial gripping using a Quadrotor. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4645–4651 (2011)
9. Ghadiok, V., Goldin, J., Ren, W.: Synthesis of feedback controllers for multiple aerial robots with geometric constraints. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4646–4651 (2011)
10. Goldstein, H.: Classical Mechanics. Addison-Wesley. ISBN 0201029693 (1980)
11. Gudiño-Lau, J., Arteaga, M.A., Muñoz, L.A., Parra-Vega, V.: On the control of cooperative robots without velocity measurements. IEEE Trans. Control Syst. Technol. **12**(4), 600–608 (2004)
12. Stuckey, R.A.: Mathematical Modelling of Helicopter Slung-Load Systems, DSTO-TR-1257. Air Operations Division Aeronautical and Maritime Research Laboratory, Department of Defense, Australia (2002)
13. Mason, M.T., Salisbury, J.K.: Robot Hands and the Mechanics of Manipulation. MIT Press, Cambridge (1985)
14. Mellinger, D., Shomin, M., Michael, N., Kumar, V.: Cooperative grasping and transport using multiple Quadrotors. In: Proceedings of the International Symposium on Distributed Autonomous Robotic Systems (2010)
15. Michael, N., Fink, J., and Kumar, V.: Cooperative manipulation and transportation with aerial robots. Auton. Robots **30**(1), 73–86 (2011)
16. Murray, R.M., Li, Z., Sastry, S.S.: A Mathematical Introduction to Robotic Manipulation. CRC Press (1994)
17. Parra-Vega, V.: Second order sliding mode control for robot arms with time base generators for finite-time tracking. Dynamics and Control, Springer **11**(2), 175–186 (2001)
18. Parra-Vega, V., Arimoto, S., Yun-Hui, L., Hirzinger, G., Akella, P.: Dynamic sliding PID control for tracking of robot manipulators: theory and experiments. IEEE Trans. Robot. Autom. **19**(6), 967–976 (2003)
19. Parra-Vega, V., Rodriguez-Angeles, A., Arimoto, S., Hirzinger, G.: High precision constrained grasping with cooperative adaptive handcontrol. J. Intell. Robot. Syst. **32**(3), 235–254 (2001)
20. Williams, D., Khatib, O.: The virtual linkage: a model for internal forces in multi-grasp manipulation. In: Proceedings of the 1993 IEEE International Conference on Robotics and Automation, pp. 1025–1030 (1993)
21. Wimbock, T., Ott, C., Albu-Schäffer, A., Hirzinger, G.: Comparison of object-level grasp controllers for dynamic dexterous manipulation. Int. J. Rob. Res. **31**(1), 3–23 (2012)
22. Wimbock, T., Ott, C., Hirzinger, G.: Analysis and experimental evaluation of the Intrinsically Passive Controller (IPC) for multifingered hands. IEEE International Conference on ICRA 2008, pp. 278–284 (2008)
23. Wimbock, T., Ott, C., Hirzinger, G.: Impedance behaviors for two-handed manipulation: design and experiments. 2007 IEEE International Conference on Robotics and Automation, pp. 4182–4189 (2007)
24. Yun-Hui, L., Arimoto, S., Parra-Vega, V., Kitagaki, K.: Decentralized adaptive control of multiple manipulators in co-operations. Int. J. Control **67**(5), 649–674 (1997)