Combined algorithms for analytical inverse kinematics solving and control of the Q-PRR aerial manipulator*

Kamel Bouzgou, Laredj Benchikh, Lydie Nouveliere, Zoubir Ahmed-Foitih, Yasmina Bestaoui

Combined Algorithms for Analytical Inverse Kinematics Solving and Control of the Q-PRR Aerial Manipulator*

Kamel Bouzgou[1,2,*], Laredj Benchikh[2], Lydie Nouveliere[2], Zoubir Ahmed-Foitih[1], Yasmina Bestaoui[2]

Abstract

This paper presents the design and modeling of a new aerial manipulator system, called Q-PRR, composed of three joints with a fixed base in the center of mass of the multirotor considered as a whole system. This structure has a prismatic joint as a first joint which allows to keep the center of gravity of the Q-PRR as close as possible to the center of gravity of the multirotor. This will also allow to reduce the influence of arm motion on the multirotor roll thus to ensure the stability of the system on trajectory tracking with dynamic changes in the multirotor's center of gravity. Furthermore, the configuration of the manipulator arm for the desired position of the end-effector given by the inverse kinematics model is kept without any change in the position and attitude of the multirotor. This paper develops both forward and inverse kinematics models for a nonlinear underactuated system using the Denavit-Hartenberg notation. When a new algorithm is presented for the inverse kinematics based on Levenberg-Marquardt algorithm. Then, the dynamic model in the joint spaces is developed with the Lagrangian formalism. The Q-PRR is controlled using a model-free control with a comparison of two states, a free fly and disturbance forces applied to the whole system with manipulator arm movement.

Keywords: Aerial Manipulation, UAVs, Dynamic Modeling, Inverse Kinematic modeling, Manipulator Arm, MFC Control.

*Project developed in LEPESA and IBISC Laboratories.

*Corresponding author: Kamel Bouzgou (*bouzgou_kamel@hotmail.fr*).
[1]Université des sciences et de la technologie USTO-MB, LEPESA Laboratory, Algeria
[2]Université Paris Saclay. Univ-Evry Val d'Essonne,IBISC Laboratory, France.

1. Introduction

Unmanned Aerial Vehicles (UAVs) have seen a growing interest due to numerous applications such as search and rescue, film industry, civil engineering, and so on. New applications such as aerial transportation or aerial manipulation have led to the combination of an aerial vehicle with a robotic arm, that can accomplish manipulation tasks that the human could not in the most dangerous situations and area. Quadcopters are the most popular aircraft platforms in the aerial manipulation applications, nearly all systems consider a quad-rotor with a revolute joints manipulator. The system support is placed on the first joint while the other joints generate the workspace of the robot arm during the UAV hovering. Many tools are attached to the UAV for grasping and object manipulation, by using gripper, multi-fingered hand (flying hand FH) also multi-link robotic arms. In this case, the object can be manipulated during the flight and manipulators are usually placed as close as possible to the UAV center of mass. These design and positioning choices are intended to avoid an important change in the UAV's inertia property and to reduce the effect of the manipulator dynamics and generated torques on UAV. Several projects have launched in Europe with aerial manipulation systems, namely remote aerial inspection (AIRobot Marconi et al. (2012); Airobots), cooperative robot system assembly and structural construction (ARCAS arcas), collaborative aerial robotic workers (Aeroworks Aeroworks), multi-arms aerial robotic system integration and advanced inspection and maintenance manipulation capabilities (AEROARMSAeroarms). Aerial manipulators system can be classified by different properties, dimensions, number of rotors, mechanical configuration, robot arm type, actuators, also with a number of multirotor. For the gripper tools, many applications are presented for different Multirotors, low-complexity grippers are designed for quadrotors to grasp beams and pick up of objects and its presented in Mellinger et al. (2011), and for the carrying and transporting payload with the helicopter is studying in Backus et al. (2014). By dimension

there are small craft aerial manipulation when a very light robot arm is attached to the quadrotor, helicopter platform and a fully actuated seven Degrees of Freedom redundant industrial robotic arm for a heavyweight manipulator in Huber et al. (2013). Several dimensions of UAVs and multi-rotor can be used to carry a robot arm in manipulation tasks, in Ibrahim et al. (2017), the authors present a hexacopter with a 3-Dof manipulator arm modeled with a Newton-Euler method.

It is possible to distinguish various types and configurations of manipulators mounted on the multi-rotors, serial robots as well as parallel robots. The Delta manipulator fixed on one side of an UAV is presented in Fumagalli et al. (2014), while in Cho and Shim (2017) and Danko et al. (2015), an aerial parallel manipulator is designed. For the interaction of a manipulator with an object, forces and torques are applied by using a dual 4-Dof arm on UAV Korpela et al. (2013). In Yang and Lee (2015) a team of quad-rotors equipped with a 2-Dof robot arm for cooperative tasks is designed. In Staub et al. (2017), the system composed by a ground manipulator and one aerial robot collaborate with each other for many ground-aerial tasks. A team of quad-rotors connected with an airborne base where the robot arm is fixed in the system center is also considered Kim et al. (2015).

For the aerial manipulation with a serial robotic arm, different configurations varying from 1-Dof Thomas et al. (2013); Yang et al. (2014), an avian-inspired 1-Dof arm was applied to accomplish bird-like grasping under a multirotor, 2-Dof Kim et al. (2013); Khalifa et al. (2015) to Pick up a small woodblock and put it in a box, 3-Dof in Mersha et al. (2014), or even to seven-Dof in Heredia et al. (2014). A hyper-redundant aerial manipulation arm has been designed with a 9-Dof in Danko and Oh (2014). Moreover, a dual four degrees-of-freedom manipulators attached to the gantry system perform grasping and manipulating tasks is developed in Korpela et al. (2012).

Some works suggested compensating the position changing of the center of mass CoM by putting a counterweight mass on the UAV when usually the center

of the multirotor fixed frame is chosen to coincide with the center of gravity (CoG). Since, it is profoundly affected by the movement of a manipulating tool. With a parallel structure, authors in Keemink et al. (2012) use a Delta mechanism, a design with 2-Dof passive rotational joints with a defined balance for overcoming gravity and defining a stable position. The revolute joint is the most widely used for the robot arm in the aerial manipulation field. However, there is little work in aerial robotics where a prismatic joint is used, in Forte et al. (2012), a manipulator arm composed of a prismatic joint is mounted on the aircraft. In Mersha et al. (2014), the 1-Dof manipulator with a prismatic joint is mounted on the quadrotor. In Backus and Dollar (2017) the prismatic-revolute-revolute joint finger is used to grasp an object and fixed on the multirotor, the function of that prismatic joint is to adapt the grasping device to the dimension of the object.

A lightweight prototype 3-arms manipulator is used in Orsag et al. (2013) to build an efficient system considered as legs of multirotor during the landing and handling operations.

The literature lists many control strategies for aerial manipulation. For instance, a Hierarchical Control of Aerial Manipulation Vehicle using PID controller in operational space is presented in Kannan et al. (2017), when in Orsag et al. (2013) a model reference adaptive control is proposed for a light-weight prototype three-arm manipulator, each arm with 2 Dofs. While an adaptive hierarchical control to compensate uncertain modelling is presented in Pierri et al. (2018). To overcome contact force effects and external disturbances, the Cartesian impedance control is designed in Lippiello and Ruggiero (2012a). In Korpela et al. (2013), the proportional-integral-derivative (PID)-based controller estimates the inertia moments and mass of two robotic arms and their motion is considered to also be disturbances of the multirotor model. In Garimella and Kobilarov (2015), a non-linear model predictive control (NMPC) approach is proposed to achieve optimized performance for aerial pick-and-place task. In Orsag et al. (2014), hybrid adaptive control is provided to achieve dynamic stability through a combination of Gain scheduling and Lyapunov based model

adaptive control (MRAC). A proportional-derivative with sliding mode controller (PD-SMC) based decoupled aerial manipulation is presented in Bouzgou et al. (2020).

Model-free has recently attracted considerable attention from researchers, as it does not require a precise system model to be controlled. As one of the free-control strategies, (MFC) model free control, authors in Younes et al. (2016) use a basic controller along with an ultra-local model to compensate for system uncertainties and disturbances. Another technique is the artificial neural networks, in Imanberdiyev and Kayacan (2019), for the outer adaptive position controller of the aerial manipulator, the fusion of the artificial neural networks and type-2 fuzzy logic controllers is presented. A variable parameter integral backstepping algorithm is used for compensating the motion of a manipulator Acosta et al. (2020).

The principle of stabilizing the global system with the control of the center of gravity using mechanical devices is presented in some works. This device is passive, its movement is provided just for aerial manipulator stabilization. A structure with a moving battery to maintain the Center of Gravity (CoG) of the whole system in a position as close as possible to the vertical axis and to counterweight the effect of the 6-Dof robotic arm is presented in Ruggiero et al. (2015). However, the battery movement is very limited when the end-effector tried to reach a desired position and cannot ensure the alignment of both UAV and robot arm center of gravity.

The principle of stabilizing the global system with the control of the center of gravity using mechanical devices is presented in some works. This device is passive, its movement is provided just for aerial manipulator stabilization. A structure with a moving battery to maintain the Center of Gravity (CoG) of the whole system in a position as close as possible to the vertical axis and to counterweight the effect of the 6-Dof robotic arm is presented in Ruggiero et al. (2015). However, the battery movement is very limited when the end-effector tried to reach a desired position and cannot ensure the alignment of both UAV and robot arm center of gravity.

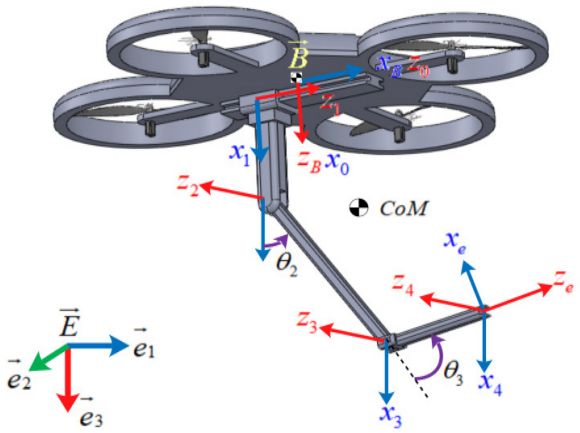


Figure 1: The structure of Q-PRR with principal frames

In this paper, the design of an aerial 3-Dof manipulator arm is developed Bouzgou et al. (2019), offering several features such as: 1) This structure can be mounted on any UAV such as multi-rotor or helicopter. 2) It ensures a wide workspace and stability of the flying system. 3) It offers many configurations of robot arm, the choice of the position and joints value for a specific task are provided by using the equation of the center of gravity as a constraint on the inverse kinematics model. 4) The first prismatic joint can compensate a small revolute joint displacement of the second and third joints due to the UAV oscillation. 5) The control of the system is ensured by a combination of CoGs in the kinematics model solutions. 6) The interaction between multirotor and moving manipulator. 7) The interaction between whole system and the object environment. There are multiple functions of the prismatic joint for the desired end-effector orientation, the first is to keep the center of gravity of the overall system as close as possible to the vertical z-axis, the second is to ensure a small displacement of the system along (x, y) axes when the target is reached and to maintain the same configuration of the manipulator.

This article is organized as follows. The next section describes the modelling part, while in the first subsection, the forward and inverse kinematic modelling

is presented using the D-H convention. In the next subsection, a new algorithm to obtain the inverse kinematic model is developed, when the dynamic model for the coupled system is introduced using a Lagrange formalism in the last subsection. In section **3**, the control approach of the Q-PRR is presented. Simulations and scenarios are shown with discussions of results in section **4**. Finally, conclusions and research prospects are presented in the last section.

2. System Modeling

Modeling and control are the important challenges of aerial manipulation. There are two approaches to address modeling and control problems. The first independent approach divides the system into two separate parts and models are designed of each component Pereira et al. (2016). This approach considers the motion and dynamics of the manipulator as external disturbances that disturb the multirotor stability, therefore, it is easier to implement it. The second is a global modeling approach Xilun et al. (2019); Lippiello and Ruggiero (2012b), by considering the multirotor and attached manipulator arm as an overall system. The challenge is that the center of mass (CoM) is constantly changing and the complex dynamics of the aerial manipulator present an important modification of the system center of mass and inertia parameters when the robot arm is moving to the desired position. Therefore, the robot arm interaction with the UAV base generates forces and torques that disturb the system.

Consider a system composed by a Multirotor vehicle equipped with a $n-$Dof robotic arm attached to the bottom, depicted in Fig (1), Let $\overrightarrow{E} : \{\overrightarrow{e_1}, \overrightarrow{e_2}, \overrightarrow{e_3}\}$ be the inertial reference frame, let $\overrightarrow{B} : \left\{\overrightarrow{b_1}, \overrightarrow{b_2}, \overrightarrow{b_3}\right\}$ be the mobile frame placed at the vehicle center of mass, and $\vec{O}_i$ the body frame of i-th link, where i=1..n denotes the link number. All body-fixed coordinate frames are located at the center of mass of their corresponding rigid body.

The position of the frame $\overrightarrow{B}$ with respect to the inertial frame $\overrightarrow{E}$, is given by the (3×1) vector denoted by p_b, while its orientation matrix denoted by R_b is described by the sequence of rotations (XYZ) around axes of the fixed frame. It

can be computed via premultiplication of three elementary rotation R_ϕ around x, R_θ around y and R_ψ around z. Hence, the attitude of multirotor can be expressed by the vector $\varphi_b = \begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^T$ of roll-pitch-yaw Euler angles. R_b can be written as follows

$$R_b = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} \tag{1}$$

where $s_* = sin(*)$, $c_* = cos(*)$, R_ψ, R_θ, R_ϕ and R_b are matrices defined in the special orthogonal group $SO(3)$, which has the following property

$$SO(3) = \left\{R \in \mathbb{R}^{3\times 3} | \, R^T R = I, det(R) = 1\right\}$$

The translation coordinates of the multirotor center of gravity relative to inertial frame $\overrightarrow{E}$ is given by the vector $p_b = \begin{bmatrix}x_b & y_b & z_b\end{bmatrix}^T \in \mathbb{R}^{(3\times 1)}$, where the pair $(p_b, R_b) \in SE(3)$. Let R_e^b be the orientation matrix of the frame attached to the end-effector and $p_e^b = \begin{bmatrix}x_{eb} & y_{eb} & z_{eb}\end{bmatrix}^T$ be the position vector of origin of such a frame with respect to $\overrightarrow{B}$. The absolute position vector and orientation matrix of the end-effector with respect to $\overrightarrow{E}$ is given by $p_e = \begin{bmatrix}x_e & y_e & z_e\end{bmatrix}^T$ and R_e respectively, the coordinate frame assignment is depicted in Figure (1).

2.1. kinematic Modeling

The Forward kinematic Model (FKM) gives the end-effector position that corresponds to a given joint configuration of both the Q-PRR and the robot arm. It is written as follows:

$$\begin{aligned} \boldsymbol{k} : \mathcal{N}(\mathbb{R}^{9\times 1}) &\longrightarrow \mathcal{M}(\mathbb{R}^{6\times 1}) \\ \xi_i &\longmapsto \chi_i = \boldsymbol{k}(\xi_i) \end{aligned} \tag{2}$$

where $\chi_i = \begin{bmatrix}x_e & y_e & z_e & \phi_e & \theta_e & \psi_e\end{bmatrix}$ is the operational joint of the end-effector, and $\xi_i = \begin{bmatrix}x_b & y_b & z_b & \phi & \theta & \psi & q_1 ... q_n\end{bmatrix}$ is the generalized joints vector of $n_\xi = 6 + n$ vector of the joint coordinates of the Q-PRR system. The Forward kinematic model (FKM) can be decomposed in two sub-FKMs. The

first is to determine the generalized end-effector coordinates in the base frame $\overrightarrow{B}$ as a function of the joint coordinates $q_{eb} = \begin{bmatrix} r_0 & \theta_2 & \theta_3 \end{bmatrix}$ of the robot arm. Hence, the FKM of the 3 Dof robot arm based on Denvit-Hartenberg convention is determined by considering the multirotor as the base for the manipulator. Thus, it will be assumed that the first joint is static and fixed to the ground. The second is to define the kinematics model of the overall system, where the generalized end-effector coordinates (χ_i) are expressed in the reference frame $\overrightarrow{E}$ as a function of the (ξ_i) joint coordinates using algebra and vector calculus. The algorithm summarizing steps to obtain the Forward kinematics model is described as follows

Algorithm 1 FKM Algorithm

1: Identify joint coordinates.
2: Identify geometric parameters that define the system.
3: Associate a reference frame to each joint.
4: Determine the orientation matrix R, position vector p of each attached frame relative to the previous one.
5: Generate table Denavit-Hartenberg.
6: Formulate matrices of homogeneous transformations from the D-H table.
7: Compute the **FGM** of the Q-PRR system.

The FGM of the robot arm in the $\overrightarrow{B}$ frame needs some measures to establish the D-H parameters. Such parameters must be obtained by putting the arm in its zero position, i.e. when the values of the joints are zero; otherwise, for each joint, a reference frame is allocated according to its type of joint, see Figure (2).

Substantiated assumptions on the modified parameters of Denavit-Hartenber in Klug et al. (2019), should be taken into consideration when establishing the D-H parameters, subsequently, are used to form homogeneous transformation matrices that mathematically define the relative position and orientation of the reference frames corresponding to the end-effector.

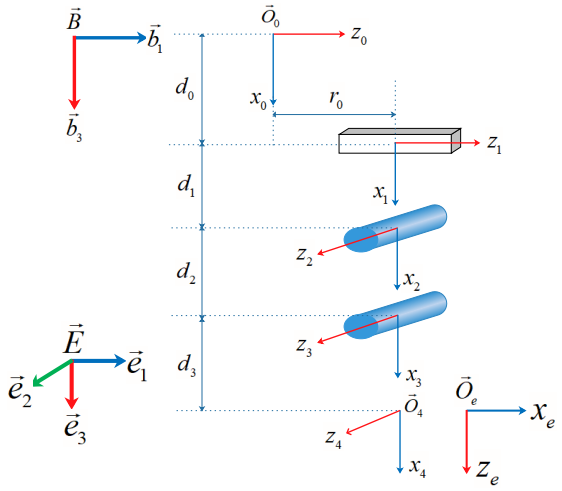


Figure 2: Frame assignments according to the Denavit-Hartenberg (D-H) notations

Hypothesis 1. *The transition between the reference frame, which the origin is in the center of mass of the multirotor to the manipulator arm frame, is made by two transformation matrices. The first is the rotation around the y-axis by* $\frac{\pi}{2}$ *angle and the second is by the rotation around the z-axis with a* π *angle.*

Hypothesis 2. *The passage from the* ${}^3T_4$ *frame to the attached end-effector frame expressed in the* $\overrightarrow{B}$ *frame by* ${}^4T_{eb}$ *matrix is done by two transformation matrices,* $\frac{\pi}{2}$ *is the rotation around the y-axis, then,* $-\frac{\pi}{2}$ *around the z-axis.*

The following table summarizes the values of the D-H parameters for the PRR robot.

Table 1: Denavit–Hartenberg parameters for the PRR Manipulator

Link number (i)	σ_i	α_i [rad]	d_i [mm]	θ_i [rad]	r_i [mm]
0	1	0	d_0	0	r_0
1	0	π	d_1	θ_2	0
2	0	0	d_2	θ_3	0
3	0	0	d_3	0	0

Since the D-H parameters are already identified in the table 1. The transformation matrices from joint i-1 to joint i can be computed using (3) as follows:

$$ {}^{i}T_{i-1} = \begin{bmatrix} cos(\theta_i) & sin(\theta_i) & 0 & a_{i-1} \\ cos(\alpha_{i-1})sin(\theta_i) & cos(\alpha_{i-1})cos(\theta_i) & -sin(\alpha_{i-1}) & -r_i sin(\alpha_{i-1}) \\ sin(\alpha_{i-1})sin(\theta_i) & sin(\alpha_{i-1})cos(\theta_i) & cos(\alpha_{i-1}) & r_i cos(\alpha_{i-1}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3} $$

The manipulator transformation matrix ${}^{b}T_{eb}$ is obtained by multiplying all the transformation matrices as follows: ${}^{b}T_{eb} = {}^{b}T_0\,{}^{0}T_1\,{}^{1}T_2\,{}^{2}T_3\,{}^{3}T_4\,{}^{4}T_{eb}$, and the matrix that presents position and orientation of the PRR robot arm can be written as follows

$$ {}^{b}T_{eb} = \begin{bmatrix} c_{23} & 0 & s_{23} & r_0 + d_2 s_2 + d_3 s_{23} \\ 0 & 1 & 0 & 0 \\ -s_{23} & 0 & c_{23} & d_0 + d_1 + d_2 c_2 + d_3 c_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4} $$

The following equivalence can be found for the global forward kinematic model

$$ {}^{b}T_{eb} = \begin{bmatrix} R_{eb} & p_{eb} \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{5} $$

By multiplying the transformation matrix of the multirotor center of mass with the matrix in (5). The position and orientation of the end-effector with respect to the inertial frame $\overrightarrow{E}$, can be defined with the following equations

$$ \begin{cases} p_e = p_b + R_b\, p^b_{eb} & \text{(6a)} \\ R_e = R_b . R^b_{eb} & \text{(6b)} \end{cases} $$

The position vector given by (6a), can be written as :

$$ p_e = \begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{bmatrix} x_b + z_{eb}(s_\phi s_\psi + c_\phi c_\psi s_\theta) + c_\psi c_\theta x_{eb} \\ y_b - z_{eb}(c_\psi s_\phi - c_\phi s_\psi s_\theta) + c_\theta c_\psi x_{eb} \\ z_b - s_\theta x_{eb} + c_\phi c_\theta z_{eb} \end{bmatrix} $$

The rotation matrix in (6b) of the end-effector to respect to the inertial frame $\overrightarrow{E}$ is written as a function of φ_b vector, and θ_2, θ_3 rotation joint angles of the robot arm, R_e can be expressed as

$$R_e = \begin{bmatrix} c_{23}c_\psi c_\theta - s_{23}(s_\phi s_\psi + c_\phi c_\psi s_\theta) & c_\psi s_\phi s_\theta - c_\phi s & c_{23}(s_\phi s_\psi + c_\phi c_\psi s_\theta) + s_{23}c_\psi c_\theta \\ s_{23}(c_\psi s_\phi - c_\phi s_\psi s_\theta) + c_{23}c_\theta s & c_\phi c_\psi + s_\phi s_\psi s_\theta & s_{23}c_\theta s_\psi - c_{23}(c_\psi s_\phi - c_\phi s_\psi s_\theta) \\ -c_{23}s_\theta - s_{23}c_\phi c_\theta & c_\theta s_\phi & c_{23}c_\phi c_\theta - s_{23}s_\theta \end{bmatrix} \quad (7)$$

Given the orientation of the end-effector as a function of Euler angles (ϕ_e, θ_e, ψ_e), the equation (7) can be rewritten in matrix elements form as

$$R_e = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

It is useful to solve the inverse problem, i.e. Determine Euler angles corresponding to the rotation matrix of the end-effector.

Assuming that $r_{13} \neq 0$ and $r_{33} \neq 0$ the following equations can be written

$$\phi_e = atan2(r_{23}, r_{13})$$
$$\theta_e = atan2(\sqrt{r_{13}^2 + r_{23}^2}, r_{33})$$
$$\psi_e = atan2(r_{32}, -r_{31})$$

Finally, equations of the Forward kinematics model of the whole system can be given as follows

$$\begin{cases} x_e = x_b + c_\psi c_\theta(r_0 + d_2 s_2 + d_3 s_{23}) + (d_0 + d_1 + d_2 c_2 + d_3 c_{23})(s_\phi s_\psi + c_\phi c_\psi s_\theta) \\ y_e = y_b + c_\theta s_\psi(r_0 + d_2 s_2 + d_3 s_{23}) - (d_0 + d_1 + d_2 c_2 + d_3 c_{23})(c_\psi s_\phi - c_\phi s_\psi s_\theta) \\ z_e = z_b - s_\theta(r_0 + d_2 s_2 + d_3 s_{23}) + c_\phi c_\theta(d_0 + d_1 + d_2 c_2 + d_3 c_{23}) \\ \phi_e = atan2(r_{23}, r_{13}) \\ \theta_e = atan2(\sqrt{r_{13}^2 + r_{23}^2}, r_{33}) \\ \psi_e = atan2(r_{32}, -r_{31}) \end{cases} \tag{8}$$

2.2. Inverse Kinematics modeling

The inverse kinematics (IK) problem is to compute joint variables that correspond to the desired position and orientation of the end-effector. In other words, find q_e for desired position p_e and orientation matrix R_e of the end-effector. The inverse of 2 can be defined as

$$\begin{aligned} \boldsymbol{k}^{-1} : \mathcal{M}(\mathbb{R}^{6\times 1}) &\longrightarrow \mathcal{N}(\mathbb{R}^{9\times 1}) \\ \chi_i &\longmapsto \xi_i = \boldsymbol{k}^{-1}(\chi_i) \end{aligned} \tag{9}$$

In the literature, several techniques for computing inverse kinematic equations are used to find closed-form solutions: Iterative methods defined as a closed-loop inverse kinematic algorithm (CLIKA) is described inLippiello and Ruggiero (2012b), numerical methods presented in Bouzgou and Ahmed-Foitih (2014). In this paper a new numerical algorithm for the inverse kinematics model is presented. It is based on the principle that multirotor should be in a stable attitude for manipulation tasks and it will be when roll and pitch angles are close zero. For the manipulation task, the ideal is that the multirotor is in an equilibrium position, thus, motion along the x and y axis should be close to zero, and since the θ and ϕ angles are the ones that generate the displacement, they must also be either zero.Therefore, a following hypothesis can be considered

Hypothesis 3.

In the hovering flight:

1) $\forall\, p_e \in \mathbb{R}^3$, *2)* $\phi = \theta = 0$, *3)* $z_b > z_e$ *And* $z_b > \sum_{i=0}^{i=3} d_i$.

For desired end-effector position and orientation $\chi_i \in \mathbb{R}^{6\times 1}$, a vector of generalized coordinates $\xi \in \mathbb{R}^{9\times 1}$ will be found with an inhomogeneous set of equations and under that assumption 3, a novel algorithm called (NAIK) which combines the numerical method, the concept of the prismatic joint to achieve the ideal position of the Center of mass of the manipulator and the Levenberg-Marquardt algorithm is presented as

Algorithm 2 NAIK Algorithm

1: Desired position and orientation p_e^d, φ_e^d.
2: Let $\theta = \phi = 0$.
3: Simplifying equation (8) with new value of θ and ϕ (10).
4: Computing ψ and $(\theta_2 + \theta 3)$ (11)(12)
5: Substitution of variables (13)
6: Apply the (LM) non-linear optimization algorithm.
7: Computing z_b.
8: Compute r_0 from the center of the mass equation (18) using θ_2,$\theta_3$values.
9: Compute x_b and y_b.

The simplified equation (8) in the hovering flight of the Q-PRR is written as follows

$$\begin{cases} x_e = x_b + c_\psi(r_0 + d_2 s_2 + d_3 s_{23}) \\ y_e = y_b + s_\psi(r_0 + d_2 s_2 + d_3 s_{23}) \\ z_e = z_b + d_0 + d_1 + d_2 c_2 + d_3 c_{23} \\ \phi_e = atan2(s_\psi c_{23}, c_\psi s_{23}) \\ \theta_e = atan2(s_{23}, c_{23}) \\ \psi_e = -\frac{1}{2}(\pi(sgn(s_{23})) - 1) \end{cases} \tag{10}$$

ψ is calculated from the 4^{th} line of (10).

$$\begin{cases} \psi = \phi_e \\ or \\ \psi = \phi_e - \pi \end{cases} \tag{11}$$

In the same way, $(\theta_2 + \theta_3)$ is computed from the 5^{th} line of (10).
$\tan(\theta_e) = \tan(\theta_2 + \theta_3)$. Then

$$\begin{cases} \theta_2 + \theta_3 = \theta_e \\ or \\ \theta_2 + \theta_3 = \theta_e - \pi \end{cases} \tag{12}$$

Substituting the (11) and (12) results in the equation system (10) and by removing the r_0 variable. the new Q-PRR position equation is given:

$$\begin{cases} x'_b = x_b + c_\psi r_0 \\ y'_b = y_b + s_\psi r_0 \end{cases} \tag{13}$$

Therefore, from (11), (12) and (13), system of equations will be defined from $\mathbb{R}^9$ to $\mathbb{R}^4$ with only three equations

$$\begin{cases} x_e = x'_b + c_{\phi_e}(r_0 + d_2 s_2 + d_3 s_{\theta_e}) \\ y_e = y'_b + s_{\phi_e}(r_0 + d_2 s_2 + d_3 s_{\theta_e}) \\ z_e = z_b + d_0 + d_1 + d_2 c_2 + d_3 c_{\theta_e} \end{cases} \tag{14}$$

Three equations with four unknown x'_b,y'_b,z_b and θ_2. It is a non-linear and non-homogeneous scheme. To resolve it, the Levenberg-Marquardt Algorithm will be used.

The variables x'_b,y'_b,z_b,ψ,θ_2 and θ_3 are now determined, and in order to obtain the values of x_b,y_b and r_0. It is useful to make the (13) system on homogeneous form. For that, the equation of the center of mass of the whole system is used and it will be defined as follows:

$$p_{cm} = \frac{1}{m_G}\left[m_b p_{cb}^T + \sum_{i=1}^{n} m_i p_{ci}^T + m_l p_{cl}^T\right] \tag{15}$$

where

$$m_G = m_b + \sum_{i=1}^{n} m_i + m_l$$

And

p_{cm}: Position vector of the center of mass of the whole system Q-PRR.

m_b: Mass of the multirotor.

m_i: Mass of the joint i .

m_l: Mass of the payload attached in the end-effector tool.

p_{cb}: Position vector of the center of mass of the Multirotor.

p_{ci}: Position vector of the center of mass of link i.

p_{cl}: Position vector of the center of mass of the payload l.

d_{cl}: Distance between the end-effector and the center of mass of the payload.

To simplify a dynamic model of the system, and to neglect additional terms in the inertial matrix, a following hypothesis can be written

Hypothesis 4.

- Manipulator links are considered to be homogeneous.

- The center of gravity of the multirotor coincides with its geometric center.

- Since $\theta_2 = \theta_3 = 0$, both z_b, z_e axes are still parallel.

Furthermore, the equation (15) can be expressed in the $\vec{B}$ frame as

$$\begin{cases} x_{cm} = \frac{1}{m_G}\left(\sum_{i=1}^{3} m_i\, r_0 + (\frac{1}{2}m_2 + m_3)d_2 s_2 + \frac{1}{2}m_3 d_3 s_{23} + x_{cl}\right) \\ y_{cm} = 0 \\ z_{cm} = \frac{1}{m_G}\left(\sum_{i=1}^{3} m_i(d_0 + d_1) + \frac{1}{2}(m_2 + m_3)d_2 c_2 + \frac{1}{2}m_3 d_3 c_{23} + z_{cl}\right) \end{cases} \tag{16}$$

Hypothesis 5.

- All position vectors are expressed with respect to $\vec{B}$ frame.

- For the free flight, it is assumed that $p_{cl} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$.

- When $p_{cm} = \begin{bmatrix} x_b & y_b & * \end{bmatrix}^T$, external forces and torques are neglected.

The manipulator arm generates torques and forces that affect the stability of the overall system, in this case, the center of gravity of the manipulator arm must be as close as possible to the vertical axis passing through the center of mass of the multirotor.

Therefore, a minimization problem must be dealt such as $z_{cm} < \epsilon$ presented as a function of three manipulator joints

$$f(r_0, \theta_2, \theta_3) = \frac{1}{m_G}\left[(m_1 + m_2 + m_3)r_0 + (\frac{1}{2}m_2 + m_3)d_2 s_{\theta_2} + \frac{1}{2}(m_3 d_3 s_{(\theta_2+\theta_3)})\right] \tag{17}$$

The reformulation of that equation to the constrained optimization problem of the Q-PRR position, which is encoded by a cost function that is to be minimized, is presented as follows

$$\begin{cases} \quad min \quad f(r_0, \theta_2, \theta_3) \\ Subject\ to \\ x_b' = x_b + c_\psi r_0 \\ y_b' = y_b + s_\psi r_0 \\ \quad where \quad r_{0_{min}} \leq r_0 \leq r_{0_{max}} \end{cases} \tag{18}$$

Regarding the boundary of the slide mechanism, care should be taken during the positioning of the arm prismatic joint. Let be $r_{0_{min}}$,$r_{0_{max}}$ boundaries of the first actuated joint r_0, its solutions are found when the distance of the center of mass to the z-axis is minimized, it should check conditions presented in the following algorithm

Algorithm 3 Test Algorithm for the r_0 value

Begin
Read r_0;
if $(-r_0^* \leq r_0 \leq r_0^*)$
    $x_b = x_b$
else
    $||x_b|| = ||x_b|| + (||r_0|| - ||r_0^*||)$;
    $x_b = sgn(x_b).||x_b||$;
End

where *sgn* symbol represents the sign function.

2.3. Kinematics modeling

By differentiating equations (6a) and (6b), the translational and angular velocities of the end-effector frame with respect to $\overrightarrow{E}$ are obtained

$$\begin{cases} \dot{p}_e = \dot{p}_b - (R_b\, p_{eb}^b)\widehat{\ }\, \omega_b \; + R_b\, \dot{p}_{eb}^b & \text{(19a)} \\ \omega_e = \omega_b + R_b\, \omega_{eb}^b & \text{(19b)} \end{cases}$$

$$\begin{cases} \dot{p}_e = \dot{p}_b - R_b\, \widehat{p}_{eb}^b\, T(\varphi_b)\, \dot{\varphi}_b \; + R_b\, \dot{p}_{eb}^b & \text{(20a)} \\ \omega_e = T(\varphi_b)\, \dot{\varphi}_b + R_b\, \omega_{eb}^b & \text{(20b)} \end{cases}$$

where $\dot{p}_b$, ω_b are the linear and angular velocities of the mobile frame $\overrightarrow{B}$ with respect to the $\overrightarrow{E}$ frame, respectively, and $\dot{p}_{eb}^b, \omega_{eb}^b$ for the translational and angular velocities of the end-effector fixed frame with respect to the mobile frame $\overrightarrow{B}$. Where $(\widehat{\cdot})$ is the hat map that denotes the skew-symmetric matrix operator defined as follows
$(\widehat{\cdot}) : \mathbb{R}^3 \longrightarrow SO(3)$, $\widehat{x}y = x \times y$, $\forall\, x, y \in \mathbb{R}^3$ Kamel et al. (2017).

$T(\varphi_b)$ is the transformation matrix between the time derivative of the Euler angles φ_b and the angular velocity of the multirotor ω_b:

$$T(\varphi_b) = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & s_\phi c_\theta \\ 0 & -s_\phi & c_\phi c_\theta \end{bmatrix}$$

Let $v^b_{eb} = \begin{bmatrix} \dot{p}^{bT}_{eb} & \omega^{bT}_{eb} \end{bmatrix}^T$ the (6×1) vector of the generalized velocity of the end-effector with respect to $\overrightarrow{B}$, it can be rewritten in terms of $\dot{q}_{eb}$ via the jacobian matrix J^b_{eb} of the manipulator,

$$v^b_{eb} = J^b_{eb}(q_{eb})\, \dot{q}_{eb} \tag{21}$$

According to (6a), (6b) and 21, the generalized end-effector velocity, $v_e = \begin{bmatrix} \dot{p}^T_e & \omega^T_e \end{bmatrix}^T$ with respect to the reference frame $\overrightarrow{E}$, can be expressed as

$$\begin{gathered} v_e = J_b\, T_A(\varphi_b)\, \dot{q}_b + J_{eb}\, \dot{q}_{eb} \\ v_e = \begin{bmatrix} J_b\, T_A(\varphi_b) & J_{eb} \end{bmatrix} \dot{\xi} \\ v_e = J_e\, \dot{\xi} \end{gathered} \tag{22}$$

.

Matrices J_b and J_{eb} are given by

$$J_b = \begin{bmatrix} I_3 & -R_b\, \widehat{p}^b_{eb} \\ 0_3 & I_3 \end{bmatrix}, \quad J_{eb} = \begin{bmatrix} R_b & 0_3 \\ 0_3 & R_b \end{bmatrix} J^b_{eb}\ , \quad T_A(\varphi_b) = \begin{bmatrix} I_3 & 0_3 \\ 0_3 & T(\varphi_b) \end{bmatrix}$$

where I_3,$0_3 \in \mathbb{R}^{3\times3}$ are the identity and zeros matrices, respectively. The jacobian matrix of the manipulator arm can be written as

$$J^b_{eb} = \begin{bmatrix} 1 & (d_2c_2 + d_3c_{23}) & d_3c_{23} \\ 0 & 0 & 0 \\ 0 & -(d_2s_2 + d_3s_{23}) & -d_3s_{23} \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{23}$$

Furthermore, J^b_{eb} can be partitioned into the (3×3) matrices as $J^b_{eb} = \begin{bmatrix} J^b_{eb_1} J^b_{eb_2} \end{bmatrix}^T$, and $J^b_{eb_1}$ and $J^b_{eb_2}$ can be computed by using relationship of the jacobian of the manipulator arm depicted in Siliciano et al. (2010).

2.4. Dynamics modeling

The dynamic model of Q-PRR can be derived by considering the Lagrange formulation. The Lagrangian is expressed by $\mathcal{L} = \mathcal{E} - \mathcal{U}$ where $\mathcal{E}$, $\mathcal{U}$ denote the kinematics and potential energy of the whole system, respectively. The Lagrange equations are given by

$$\frac{d}{dt}\frac{\delta \mathcal{L}}{\delta \dot{\xi}_i} - \frac{\delta \mathcal{L}}{\delta \xi_i} = u_i + u_{ext} \tag{24}$$

where $i = 1, ..., 6+n$ is the $i-th$ coordinate of ξ, the generalized variables and u_i is the $((6+n)\times 1)$ vector of generalized forces and torques.
The kinetic energy of the global system expressed in $\vec{B}$ frame is given by:

$$\mathcal{E} = \mathcal{E}_b + \sum_{i=1}^{n} \mathcal{E}_{ci} \tag{25}$$

Where $\mathcal{E}_b$ is the kinetic energy of multirotor and $\mathcal{E}_{ci}$ is the kinetic energy of the i link of manipulator arm.

The kinematic energy of the multirotor can be expressed as

$$\mathcal{E}_b = \frac{1}{2}\, m_b\, \dot{p}_b^T\, \dot{p}_b + \frac{1}{2}\, \omega_b^T\, R_b\, \boldsymbol{I}_b\, R_b^T\, \omega_b$$

where $\boldsymbol{I}_b$ and m_b are the inertia matrix and the mass of the multirotor expressed with the respect to $\vec{B}$ frame, respectively. Taking into account that $Q = R_b^T\, T_b$, the kinetic energy of multirotor can be rewritten on the new form as

$$\mathcal{E}_b = \frac{1}{2}\, m_b\, \dot{p}_b^T\, \dot{p}_b + \frac{1}{2}\, \dot{\varphi}_b^T\, Q^T\, \boldsymbol{I}_b\, Q\, \dot{\varphi}_b \tag{26}$$

whereas, the kinetic energy of $i-th$ link of the robotic manipulator is given by

$$\mathcal{E}_{ci} = \frac{1}{2}\, m_{ci}\, \dot{p}_{ci}^T\, \dot{p}_{ci} + \frac{1}{2}\omega_{ci}^T\, R_b\, R_{ci}^b\, \boldsymbol{I}_{ci}^i\, (R_{ci}^b)^T\, R_b^T\, \omega_{ci} \tag{27}$$

where m_{ci} and $\boldsymbol{I}_{ci}^i$ are the mass and the moment of inertia of the i-th link about their center of mass expressed in their body fixed frame. From (25), (26) and (27), the kinetic energy of the whole system can be written as:

$$\mathcal{E} = \frac{1}{2}\dot{\xi}^T \left(m_Q\, B_{Q1} + \sum_{i=1}^{n} m_{ci}\, B_{ci1}\, B_{ci1}^T \right) + \frac{1}{2}\dot{\xi}^T \left(B_{Q2}\, \boldsymbol{I}_Q B_{Q2}^T + \sum_{i=1}^{n} B_{ci2}\, \boldsymbol{I}_{ci}^i\, B_{ci2}^T \right) \dot{\xi} \tag{28}$$

$$\mathcal{E} = \frac{1}{2}\dot{\xi}^T B\,\dot{\xi} \tag{29}$$

B is the (3×3) matrix of (3×3) inertia matrix elements, $B \in \mathbb{M}^{(3\times3)}$ and $B_{ij} \in \mathbb{R}^{(3\times3)}$, more details of elements of B matrix are presented in Lippiello and Ruggiero (2012b).

The potential energy of the whole system is given by the sum of the multirotor and each link of the manipulator arm potential energies:

$$\mathcal{U} = \mathcal{U}_b + \sum_{i=1}^{n} \mathcal{U}_{ci} \tag{30}$$

The potential energy of the multirotor is given by:

$$\mathcal{P}_b = m_b g e_3^T p_b \tag{31}$$

On the other hand, the potential energy of each link i of manipulator is given by:

$$\mathcal{U}_{ci} = m_{ci} g e_3^T (p_b + R_b p_{ci}^b) \tag{32}$$

The total potential energy of overall system is the sum of equations (31) and (32) therefore:

$$\mathcal{U} = m_b g e_3^T p_b + \sum_{i=1}^{n} m_{ci} g e_3^T (p_b + R_b p_{ci}^b) \tag{33}$$

where g is the gravity acceleration value and $e_3 = [0 \quad 0 \quad 1]$ unit vector along z axis. Considering equations (24), (25), and (33), the dynamic model of the global system can be written as:

$$B(\xi_i)\ddot{\xi}_i + C(\xi_i, \dot{\xi}_i)\dot{\xi}_i + G(\xi_i) + D(\xi_i) = u + u_{ext} \tag{34}$$

$D = 0$ is therefore considered when aerodynamic forces are assumed to be negligible. G is a $(6+n)$ vector of gravitational terms given by deriving the potential energy as follows

$$G(\xi) = \frac{\delta \mathcal{P}}{\delta \xi} \tag{35}$$

C is the matrix of Coriolis and centrifugal terms given by

$$C_{ij} = \sum_{k=1}^{6+n} \frac{1}{2} \left(\frac{\delta b_{ij}}{\delta \xi_k} + \frac{\delta b_{ik}}{\delta \xi_j} - \frac{\delta b_{jk}}{\delta \xi_i} \right) \dot{\xi}_k \tag{36}$$

where $B(\xi_i)$ is (6+n) symmetric and positive define inertia matrix, and b_ij is its generic element, it represents the moment of inertia at Joint i. u is the vector of generalized forces at the i-th joint level. With the property $\dot{B}(\xi_i) - 2C(\xi_i, \dot{\xi}_i)$ is the skew symmetric matrix. External forces and moments are given by u_{ext}, and the vector of forces and torques generated by actuators is represented by u as follows

$$u = \begin{bmatrix} u_{f_b} \\ u_{\tau_b} \\ u_{\mu} \end{bmatrix} = \begin{bmatrix} R_b \, f_b \\ R_b^T \, T_{\varphi_b} \, \tau_b \\ \mu \end{bmatrix} \tag{37}$$

where

$$f_b = \begin{bmatrix} 0 \\ 0 \\ f_{bz} \end{bmatrix} \quad , \quad \tau_b = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \quad , \quad \mu = \begin{bmatrix} f_{r_0} \\ \tau_{\theta_2} \\ \tau_{\theta_3} \end{bmatrix}$$

f_{bz} is the collective thrust of the rotors directed along the vertical axis z_b, whereas τ_ϕ,τ_θ and τ_ψ are the three rotational torques acting around x_b,y_b and z_b-axes, respectively. In addition, f_{r_0} is the force applied to the prismatic joint along the x_b-axis, where $\tau_{\theta 2}$ and $\tau_{\theta 3}$ are the torques acting on the rotational joints of the robot arm. Furthermore, both f_{bz} and τ_b are related to the four actuation forces output by the multirotor motors f_i via the following relation

$$\begin{bmatrix} f_{bz} \\ \tau_b \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ c & -c & c & -c \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{38}$$

where l is the distance from each motor to the multirotor center of mass. When c is the drag factor.

3. Q-PRR Controller

The proposed control strategy consists of three-layers, the inverse kinematics algorithm called (NAIK) computes from the desired position and orientation of the end-effector, the joint variables of the Q-PRR within the upper-layer. Then, in the middle layer, an internal loop is implemented to compute desired pitch and roll angles using the proportional integral derivative controller. In the bottom layer, the Model-Free controller is designed to track desired trajectories. The whole scheme of the proposed controller is presented in Fig (3). The desired position will be introduced, then the inverse kinematic algorithm computes a center of mass position and joint values of the manipulator when a position is reached, the robot arm move with respect to the best configuration that ensures the stability and keeps the attitude angles close to zero. Nevertheless, where there is a difference in the distances among points, i.e desired and center of mass of the multirotor, the compensation task for the prismatic joint value will be changed.

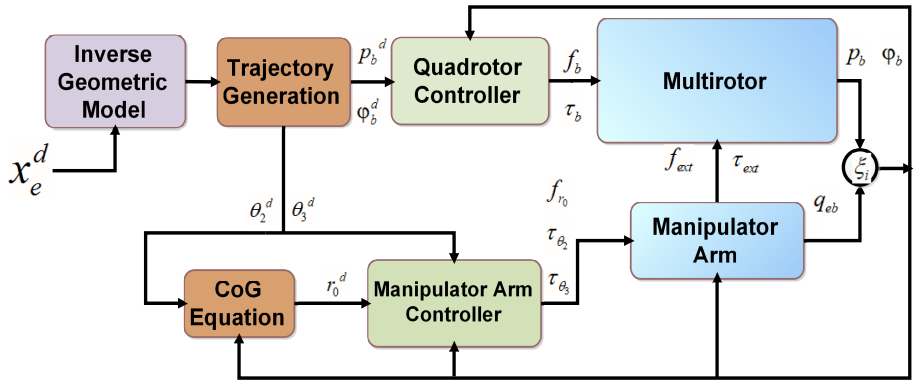


Figure 3: The control block diagram of the Q-PRR

3.1. Multirotor Controller

From the relation in (37), u_{f_b} can be rewritten in the three axes with the following equation

$$u_{f_b} = \begin{bmatrix} u_{f_{bx}} \\ u_{f_{by}} \\ u_{f_{bz}} \end{bmatrix} = \begin{bmatrix} (c_\psi s_\theta c_\phi + s_\psi s_\phi) f_{bz} \\ (s_\psi s_\theta c_\phi - c_\psi s_\phi) f_{bz} \\ c_\theta c_\phi f_{bz} \end{bmatrix} \tag{39}$$

The thrust force and the roll and pitch angle control vector can be extracted from the equation (39), and by assuming that $f_{bz} \neq 0$, $\theta_d \neq \pm\frac{\pi}{2}$ and $\phi_d \neq \pm\frac{\pi}{2}$. Desired ϕ_d and θ_d angles can be computed as follows

$$\begin{cases} \theta_d = arctan\left(\dfrac{u_{f_{bx}} c_{\psi_d} + u_{f_{by}} s_{\psi_d}}{u_{f_{bz}}}\right) & \text{(40a)} \\ \phi_d = arcsin\left(\dfrac{u_{f_{bx}} s_{\psi_d} - u_{f_{by}} c_{\psi_d}}{u_{f_{bz}}}\right) & \text{(40b)} \end{cases}$$

The input control vector of the desired θ_d and ϕ_d can therefore be expressed as

$$\begin{cases} u_{\theta_d} = \ddot{\theta}_d + k_{\theta,1}\dot{e}_{\theta_d} + k_{\theta,2} e_{\theta_d} & \text{(41a)} \\ u_{\phi_d} = \ddot{\phi}_d + k_{\phi,1}\dot{e}_{\phi_d} + k_{\phi,2} e_{\phi_d} & \text{(41b)} \end{cases}$$

where $\dot{e}_{\theta_d} = (\dot{\theta_d} - \dot{\theta})$, and $\dot{e}_{\phi_d} = (\dot{\phi_d} - \dot{\phi})$ are the trajectory errors. $k_{\theta,1}$,$k_{\theta,2}$,$k_{\phi,1}$ and $k_{\phi,2}$ are positive gains.

3.2. System control based on the Free model controller

Ultra-local model control consists of attempting to approximate by input and output calculations that can be compensated by control in order to obtain a better output tracking. It includes the creation of a pure ultra-local system model, which replaces equation of the whole dynamics system in(34) and it can be written as follows

$$y^{(v)}(t) = F(t) + \alpha\, u(t) \tag{42}$$

In the case of this study, the order v can be equal to 2, which leads to the simplified equation as

$$\ddot{y}(t) = F(t) + \alpha\, u(t) \tag{43}$$

where F represents the real dynamics of the model, it is a continuously updated value and it could be approximated to attenuate the noise that could damage the output-system performances. y is the system output that represent the coordinate joints of multirotor and the manipulator arm, u is the control input. α is the estimate of the input factor which can be designed by the operator. The principle of this method is to estimate the unknown term F. Several estimation methods have been used in the literature. In this paper, the method proposed in Younes et al. (2016) is used. It should be noted that the estimation value is true for a short period of time and should be modified continuously Fliess and Join (2013); Levant (2003).

$$F(t) \approx \widehat{F}(t) = F(t-\epsilon) = \ddot{y}(t-\epsilon) - \alpha u(t-\epsilon) \tag{44}$$

The estimation method using differentiators depicted in Younes et al. (2016) is used, where ϵ should be sufficiently small.

The intelligent (iPID) controller for a second order system ($v = 2$) is implemented for the Q-PRR model. Therefore, the model-free control law can be written as follows

$$u(t) = \frac{\widehat{F}(t) - \ddot{y}_d(t) - K_P e(t) - K_I \int_0^t e(\tau)d\tau - K_D \dot{e}(t)}{\alpha} \tag{45}$$

The $\widehat{F}$ is a real-time estimation of F, where $y_d(t)$ is the desired trajectory and its error e is defined as $e(t) = y_d(t) - y(t)$. The usual proportional integral derivative tuning gains are defined as, K_P,K_I,$K_D \in \mathbb{R}$, they are positive constants for each controlled variable.

At each iteration, the value of F will be calculated and the new value predicts the system dynamics and injects the adjustment into the system input.

4. Simulation results and discussion

The aerial manipulator controller can be subdivided into two parts. First, the overall system controller is designed using the inverse dynamic controller. This approach is based on the concept of having a control vector u, as a function of the system state, capable of achieving a linear form input / output relationship. The result is an accurate linearization of the system dynamics obtained by nonlinear state feedback, the modelling structure and functions of the inverse dynamic controller can be explained for more details inBouzgou (2019). The second part consists of the model-free control, This approach assumes the system to be a black box and manipulates the input/output of the model, and it uses a local linear approximation of a process model that is accurate for a short period time and a fast estimator is used to update this approximation. In this case, the efficiency of the proposed controller is tested with three scenarios, first where the Q-PRR is in free motion, i.e. without external forces and torques. Second, under a gust of wind. Finally, where the system is under a pulse signal as abrupt external forces. The disturbance force has the same value projected on $\overrightarrow{e_1}$, $\overrightarrow{e_2}$ and $\overrightarrow{e_3}$ axes of the inertial frame $\overrightarrow{E}$.

The Q-PRR is designed using SolidWorks, and the simulation was executed in the Matlab/Simulink. Manipulator masses and inertias values have been estimated from the mass properties tools and are presented in the table (2).

Table 2: Dynamic parameters of each Link

Link	mass $[g]$	length $[cm]$	$I_{xx}[kg.m^2]$	$I_{yy}[kg.m^2]$	$I_{zz}[kg.m^2]$
Base	33	6	1.67e-6	1.32e-6	2.56e-6
1	117.2	32	1.981e-5	1.972e-5	2.28e-5
2	277.3	53	2.274e-5	5.013e-5	2.2767e-5
3	96.1	28	4.33e-6	7.99e-6	3.94e-6

Although the off-diagonal components called inertia products are ignored due to the symmetry of the multi-rotor when the axes passing through the center of mass are respected. Parameters of the multirotor are presented in the table (3)

Table 3: Dynamic parameters of Multirotor

Multirotor	mass [kg]	I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}
	3.029	6.98664e-3	7.23191e-3	11.57173e-3	$-3.6e$-7	2.876e-4	$-8.01e$-5

For simulation tests, the acceleration $\ddot{\xi}_i$ of the Q-PRR have been calculated using equations 34 and 45, and it can be written as

$$\ddot{\xi}_i(t) = B(\xi_i)^{-1}(t)\left[\frac{1}{\alpha}\widehat{F}(t) - \ddot{y}_d(t) - K_P e(t) - K_I\int_0^t e(\tau)d\tau - K_D\dot{e}(t) - C(\xi_i,\dot{\xi}_i)\dot{\xi}_i(t) - G(\xi_i)(t) + D(\xi_i)(t) + u_{ext}(t)\right] \quad (46)$$

The Q-PRR should reach a desired point in a 3-D space. After the take-off, the multirotor tracks a z reference trajectory and when it arrives at the desired altitude, the multirotor hovers for a horizontal point along the x-axis then along y-axis. This trajectory is equivalent to avoiding obstacles in the shape of a cube. When the desired position is reached, the manipulation tasks of the robot arm can be started to ensure the desired position of the end-effector. Therefore, the total simulation time is 60 sec, and the reference trajectory is expressed as:

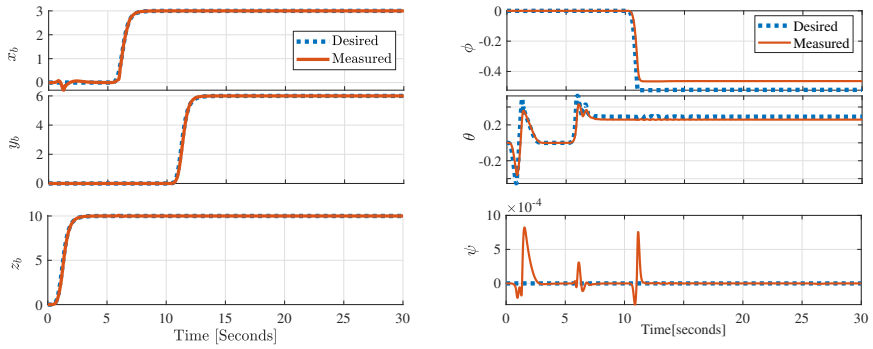
$$\sigma(t) = \begin{cases} 0, & 0s \le t \le t_1 \\ L_{Q_p}\frac{(t-t_1)^5}{(t-t_1)^5+(T_s-t+t_1)^5}, & t_1 \le t \le t_2 \\ L_{Q_p}, & t_2 \le t \le t_f \end{cases}$$

with $T_s = 5$ seconds, $t_f = 60$ seconds, and each coordinates joints value is presented in the table 4.

Table 4: Desired position value for the Multirotor center of mass and joint coordinates of robot arm

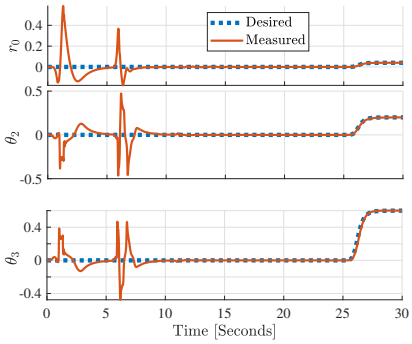
$\sigma(t)$	L_{Q_p}	$t_1[s]$	$t_2[s]$
x_r	$3[m]$	5	10
y_r	$6[m]$	10	15
z_r	$10[m]$	0	5
r_{0_r}	$0.04[rad]$	25	45
$r_{\theta_{2_r}}$	$0.2[rad]$	25	45
$r_{\theta_{3_r}}$	$0.6[rad]$	25	45

The inverse dynamic model is explicitly based on the system model and compensates, through inverse dynamics, non-linear dynamic terms of the model, by decoupling the interactions between system joint. The Proportional-Integral-Derivative controller is used in the first bloc of positions, which the values of its parameters are, (K_{P_x}=1.4, K_{I_x}=1.5, K_{D_x}=0.1) and the PI controller for the robot arm ($K_{P_{r_0}}$=2.5, $K_{I_{r_0}}$=2.7), ($K_{P_{\theta_2}}$=3.9, $K_{I_{\theta_2}}$=4.1), ($K_{P_{\theta_3}}$=2.5, $K_{I_{\theta_3}}$=2.9). Figures 4 depict responses of all coordinate joints to follow the desired trajectory. For the started simulation to t=10 sec, the stability of the robot arm is not ensured, and the tracking performance is slightly worse, with some extra oscillations and overflow from starting t=8 sec.



(a) Position Error $[m]$

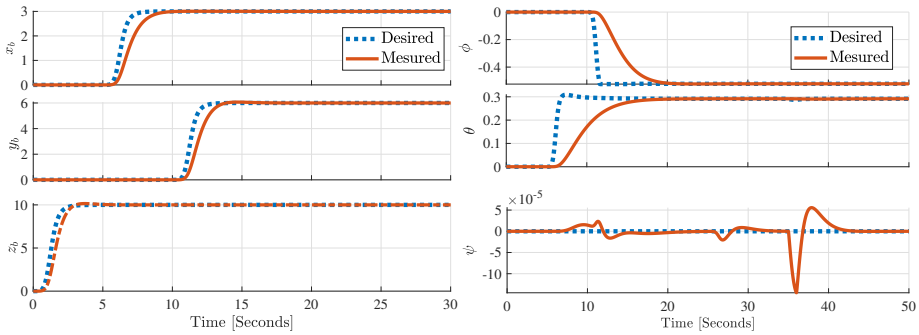
(b) Angle Errors $[rad]$



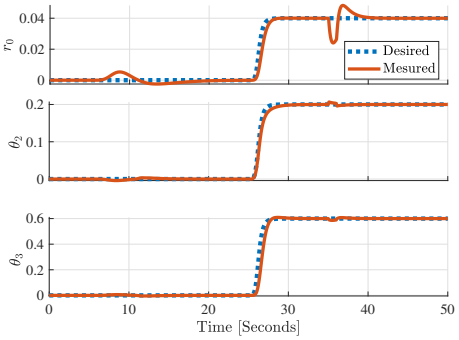
(c) Joints Errors $[m, rad, rad]$

Figure 4: Inverse Dynamic model applied to the Q-PRR with PID Position Controller (PID-IDC)

The proportional-integral controller is used to resolve the overflow of the desired position of the robot arm. The system's stability to track desired trajectory is expected. Therefore, a noise signal is added as a disturbance to validate the proposed controller. Figure 5 shows the system output with the pulse signal of the disturbance noise applied at t=35 sec with a simulation period of 1.67%. Nevertheless the disturbance on the robot arm is displayed from t=7 sec to t=12 sec and it represents 12.6 percent of the final output signal.



(a) Position $[m]$ (b) Angles $[rad]$



(c) Joints $[m, rad, rad]$

Figure 5: (PID-IDC-PI) Controller applied to the Q-PRR

The tracking of the desired orientation angles is very slow, the output reaches the desired final value in $8s$ where that accuracy is not appropriate. Therefore, Model-Free Control (MFC) is implemented in this section, as a local model-based control technique, especially in non-linear modes, MFC is constantly updated with input-output behaviour to overcome the un-modelled system dynamics and uncertainties. The system flight is performed in two steps, the first is to carry the multirotor to the desired position, the second is when the UAV has achieved location, the arm manipulation process will start. Three aspects will be examined, the first is when the system is in a free environment, .i.e there are no disturbance forces applied on the Q-PRR. The second is when the flight is affected by a gust of wind or random disturbances. The third case is an improvement of the previous one when adding a control part to keep the system

close to the desired position. MFC works with a classical PID controller, that combination is known as intelligent-PID controller. Gains of controller using Simulink/Matlab are chosen in order to have no oscillations and to reach desired values as soon as possible. Control parameters used are shown in the Table (5).

Table 5: Attitude control gains

Attitude	K_p	K_I	K_D
Pitch	30	20	4
Roll	30	20	1.5
Yaw	10	18	2.5

The experimental results for the desired trajectory are summarized bellow, in Figure 6, a 3-D space trajectory figure depicts the allure of multirotor hovering. In addition, the position (x, y, and z) response of the UAV center of gravity is depicted in Figures (7, 8 and 9) Simulation results of the Model Free Control

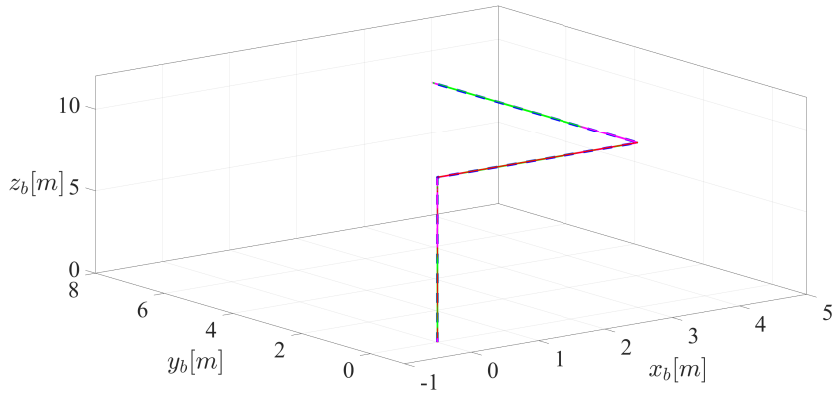
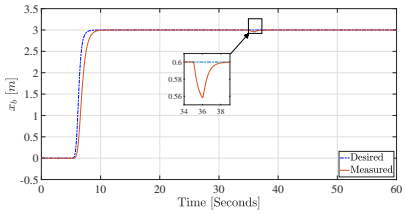


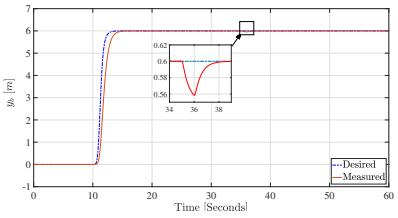
Figure 6: 3D trajectory path

are compared with results extracted by using the inverse dynamics method described in Bouzgou (2019), when the desired trajectory was same. Figures 4

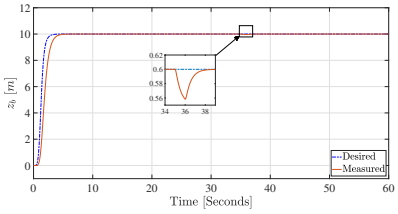
depict final values of all coordinate joint positions when the end-effector follows the desired trajectory, the simulation starts from t=10 sec, therefore, the robot arm stability is not guaranteed. In second simulation with model free control (MFC), The performance and behaviour tracking is better. However, for the first simulation, the tracking performance is slightly worse, with some extra oscillations and overflow from starting to t=8 sec.



(a) Position $[m]$

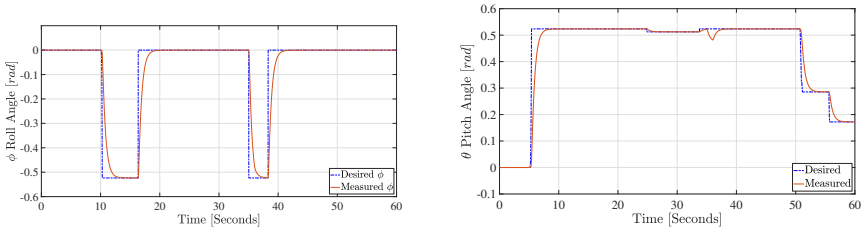


(b) Angles $[rad]$

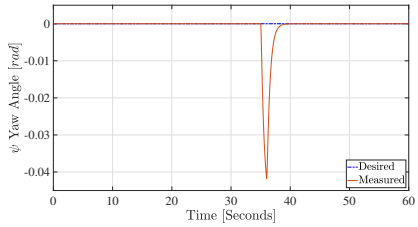


(c) Joints $[m, rad, rad]$

Figure 7: Position of the Q-PRR using the (i-PID)-MFC Controller

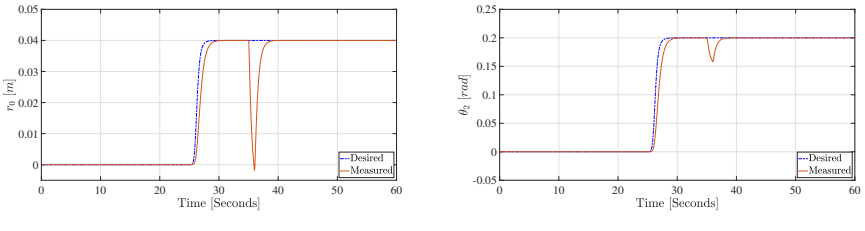


(a) Position $[m]$ (b) Angles $[rad]$

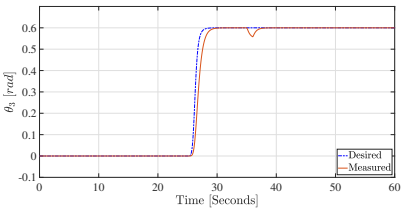


(c) Joints $[m, rad, rad]$

Figure 8: Orientation of the Q-PRR using the (i-PID)-MFC Controller



(a) Position $[m]$ (b) Angles $[rad]$



(c) Joints $[m, rad, rad]$

Figure 9: Positions of the Q-PRR arm using the (i-PID)-MFC Controller

A comparison between the different controller strategies is proved by using Euclidean RMSE (Root Mean Square Error) to validate the efficiency of

methods applied, with a dynamic inverse method and Model Free Control with disturbance. Table (6) shows the RMSE and ISE values for different control strategies. Figure 10 shows a position, attitude and position joints of the last simulation.

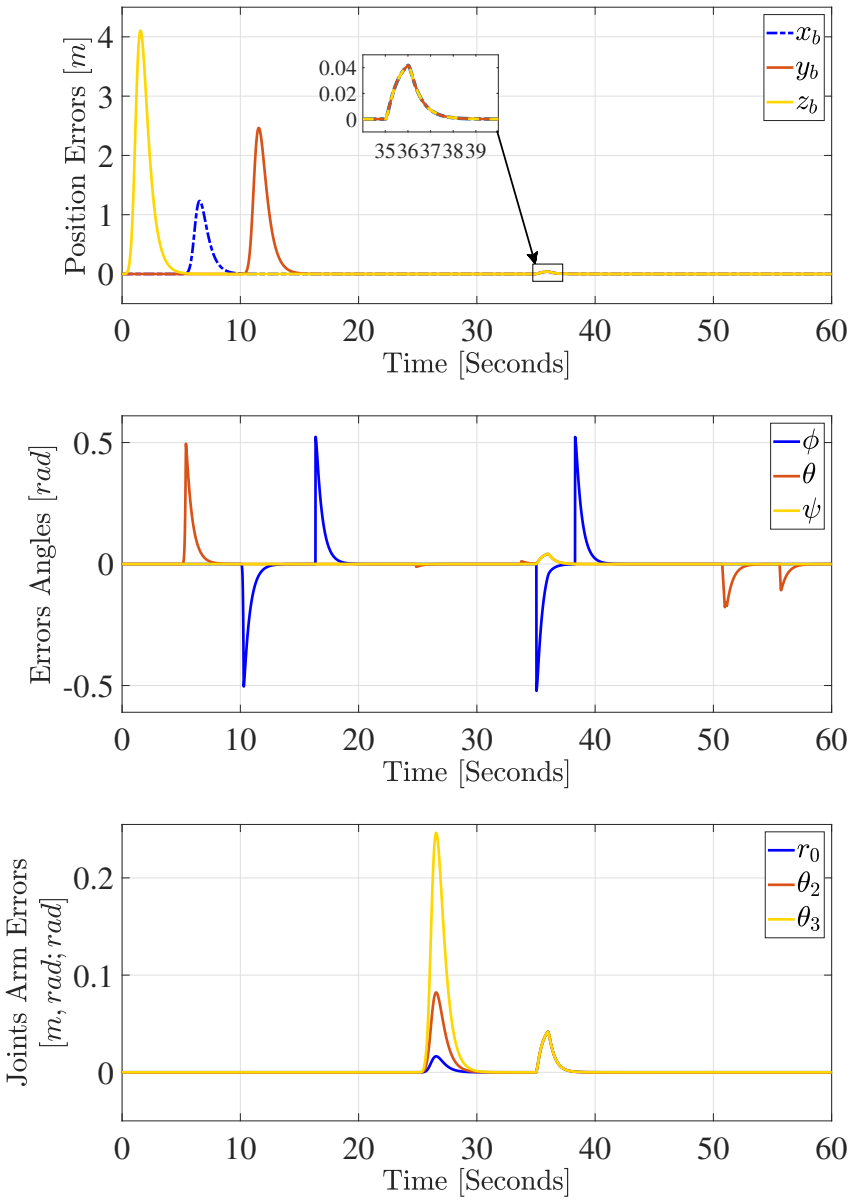


Figure 10: Q-PRR Errors during fly and under disturbance forces applied at t=35 sec

When a disturbance force is applied to the Q-PRR system in three directions of the $\overrightarrow{E}$ reference frame at t=35sec, the error increases with 1 percent for the position and 0.6 percent for the attitude, and the error on the robot arm's joint

position increases to 12.5 percent due to the base arm's relationship with the Multirotor orientation. In addition, the results show the appropriateness of the proposed control strategy.

Table 6: Metric Comparison

	RMSE	ISE
DIC	0.0024	$2.4012e-04$
DIC+PID	0.0124	0.0410
MFC+i-PID	0.0049	$2.4012e-05$

It can be noticed that MFC-PID controller decreases the MFC controller RMSE value by around 48%. As a result, the hovering and manipulation performance of the Q-PRR which uses intelligent MFC-PID structure becomes significantly better compared to the case when the MFC controller is only used.

5. Conclusions and future work

The main topic of this paper is the design and modelling of novel aerial manipulator structure. A prismatic joint that can keep and stabilize the system center of gravity is implemented. A new algorithm for the inverse kinematics problem is presented for the Q-PRR and its solutions are resolved on the principle of a numerical approach that is combined with the Levenberg-Marquardt algorithm. The proposed design may be used by applying of disturbance uncertainty associated with external forces, torques and unknown flying environment to ensure stability and tracking of the multirotor end-effector and center of gravity. Three different controllers and scenarios are presented, demonstrating the efficiency of the proposed approach and verifying by simulation that the Free model controller is robust compared to the inverse dynamic method when the disturbance forces are introduced. The next step will be the robustness test by attaching different masses on the end-effector and for several joints position of the manipulator arm. More simulations will need to be done to support this, especially in situations where the mass is unknown.

References

Acosta J, De Cos C, Ollero A. Accurate control of aerial manipulators outdoors. a reliable and self-coordinated nonlinear approach. Aerospace Science and Technology 2020;99:105731. doi:10.1016/j.ast.2020.105731.

Aeroarms ;URL: https://inspection-robotics.com/aeroarms/.

Aeroworks ;URL: http://www.aerowork-724s2020.eu/.

Airobots ;URL: http://airobots.dei.unibo.it/.

arcas . Aerial robotics cooperative assembly system ;URL: http://www.arcas-project.eu/.

Backus SB, Dollar AM. Design optimization of a prismatic-revolute-revolute joint hand for grasping from unconstrained vehicles. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers; volume 58189; 2017. p. V05BT08A002. doi:10.1115/DETC2017-67222.

Backus SB, Odhner LU, Dollar AM. Design of hands for aerial manipulation: Actuator number and routing for grasping and perching. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2014. p. 34–40. doi:10.1109/IROS.2014.6942537.

Bouzgou K. BLNLBYAFZ. A novel aerial manipulation design, modelling and control for geometric com compensation. 16th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2019) 2019;Vol. 2:475–82. doi:10.5220/0007951404750482.

Bouzgou K, Ahmed-Foitih Z. Geometric modeling and singularity of 6 dof fanuc 200ic robot. In: Fourth edition of the International Conference on the Innovative Computing Technology (INTECH 2014). IEEE; 2014. p. 208–14. doi:10.1109/INTECH.2014.6927745.

Bouzgou K, Benchikh L, Nouveliere L, Ahmed-Foitih Z, et al. Pd sliding mode controller based decoupled aerial manipulation. In: 17th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2020). 2020. p. 484–9. doi:0.5220/0009856704840489.

Bouzgou K, Benchikh L, Nouveliere L, Bestaoui Y, Ahmed-Foitih Z. A new classification and aerial manipulation q-prr design. In: 24th International Conference on Methods and Models in Automation and Robotics (MMAR 2019). 2019. doi:10.13140/RG.2.2.24863.79520.

Cho S, Shim DH. Development of a vision-enabled aerial manipulator using a parallel robot. Transactions of the Japan Society for Aeronautical and Space Sciences, Aerospace Technology Japan 2017;15(APISAT-2016):a27–36. doi:10.2322/tastj.15.a27.

Danko TW, Chaney KP, Oh PY. A parallel manipulator for mobile manipulating uavs. In: 2015 IEEE international conference on technologies for practical robot applications (TePRA). IEEE; 2015. p. 1–6. doi:10.1109/TePRA.2015.7219682.

Danko TW, Oh PY. Design and control of a hyper-redundant manipulator for mobile manipulating unmanned aerial vehicles. Journal of Intelligent & Robotic Systems 2014;73(1):709–23. doi:10.1007/s10846-013-9935-2.

Fliess M, Join C. Model-free control. International Journal of Control 2013;86(12):2228–52. doi:10.48550/arXiv.1305.7085.

Forte F, Naldi R, Macchelli A, Marconi L. Impedance control of an aerial manipulator. In: 2012 American Control Conference (ACC). IEEE; 2012. p. 3839–44. doi:10.1109/ACC.2012.6315568.

Fumagalli M, Naldi R, Macchelli A, Forte F, Keemink AQ, Stramigioli S, Carloni R, Marconi L. Developing an aerial manipulator prototype: Physical interaction with the environment. IEEE robotics & automation magazine 2014;21(3):41–50. doi:10.1109/MRA.2013.2287454.

Garimella G, Kobilarov M. Towards model-predictive control for aerial pick-and-place. In: 2015 IEEE international conference on robotics and automation (ICRA). IEEE; 2015. p. 4692–7. doi:10.1109/ICRA.2015.7139850.

Heredia G, Jimenez-Cano A, Sanchez I, Llorente D, Vega V, Braga J, Acosta J, Ollero A. Control of a multirotor outdoor aerial manipulator. In: 2014 IEEE/RSJ international conference on intelligent robots and systems. IEEE; 2014. p. 3417–22. doi:10.1109/IROS.2014.6943038.

Huber F, Kondak K, Krieger K, Sommer D, Schwarzbach M, Laiacker M, Kossyk I, Parusel S, Haddadin S, Albu-Schäffer A. First analysis and experiments in aerial manipulation using fully actuated redundant robot arm. In: 2013 IEEE/RSJ international conference on intelligent robots and systems. IEEE; 2013. p. 3452–7. doi:10.1109/IROS.2013.6696848.

Ibrahim IN, Pavol B, Al Akkad MA, Karam A. Navigation control and stability investigation of a hexacopter equipped with an aerial manipulater. In: 2017 21st International Conference on Process Control (PC). IEEE; 2017. p. 204–9. doi:10.1109/PC.2017.7976214.

Imanberdiyev N, Kayacan E. A fast learning control strategy for unmanned aerial manipulators. Journal of Intelligent & Robotic Systems 2019;94(3):805–24. doi:10.1007/s10846-018-0884-7.

Kamel B, Yasmina B, Laredj B, Benaoumeur I, Zoubir AF. Dynamic modeling, simulation and pid controller of unmanned aerial vehicle uav. In: 2017 Seventh International Conference on Innovative Computing Technology (INTECH). IEEE; 2017. p. 64–9. doi:10.1109/INTECH.2017.8102445.

Kannan S, Bezzaoucha S, Guzman SQ, Dentler J, Olivares-Mendez MA, Voos H. Hierarchical control of aerial manipulation vehicle. In: AIP Conference Proceedings. AIP Publishing LLC; volume 1798; 2017. p. 020069. doi:10.1063/1.4972661.

Keemink AQ, Fumagalli M, Stramigioli S, Carloni R. Mechanical design of a manipulation system for unmanned aerial vehicles. In: 2012 IEEE international conference on robotics and automation. IEEE; 2012. p. 3147–52. doi:10.1109/ICRA.2012.6224749.

Khalifa A, Fanni M, Ramadan A, Abo-Ismail A. New quadrotor manipulation system: Inverse kinematics, identification and ric-based control. International Journal of Recent advances in Mechanical Engineering (IJMECH) 2015;4(3):39–58. doi:10.14810/ijmech.2015.4304.

Kim J, Nguyen HN, Lee D. Preliminary control design on spherically-connected multiple-quadrotor manipulator system. In: 2015 12th international conference on ubiquitous robots and ambient intelligence (URAI). IEEE; 2015. p. 206–7. doi:10.1109/URAI.2015.7358871.

Kim S, Choi S, Kim HJ. Aerial manipulation using a quadrotor with a two dof robotic arm. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2013. p. 4990–5. doi:10.1109/IROS.2013.6697077.

Klug C, Schmalstieg D, Gloor T, Arth C. A complete workflow for automatic forward kinematics model extraction of robotic total stations using the denavit-hartenberg convention. Journal of Intelligent & Robotic Systems 2019;95(2):311–29. doi:10.1007/s10846-018-0931-4.

Korpela C, Orsag M, Pekala M, Oh P. Dynamic stability of a mobile manipulating unmanned aerial vehicle. In: 2013 IEEE international conference on robotics and automation. IEEE; 2013. p. 4922–7. doi:10.1109/ICRA.2013.6631280.

Korpela CM, Danko TW, Oh PY. Mm-uav: Mobile manipulating unmanned aerial vehicle. Journal of Intelligent & Robotic Systems 2012;65(1):93–101. doi:10.1007/s10846-011-9591-3.

Levant A. Higher-order sliding modes, differentiation and output-feedback control. International journal of Control 2003;76(9-10):924–41. doi:10.1080/0020717031000099029.

Lippiello V, Ruggiero F. Cartesian impedance control of a uav with a robotic arm. IFAC Proceedings Volumes 2012a;45(22):704–9. doi:10.3182/20120905-3-HR-2030.00158.

Lippiello V, Ruggiero F. Exploiting redundancy in cartesian impedance control of uavs equipped with a robotic arm. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2012b. p. 3768–73. doi:10.1109/IROS.2012.6386021.

Marconi L, Basile F, Caprari G, Carloni R, Chiacchio P, Hurzeler C, Lippiello V, Naldi R, Nikolic J, Siciliano B, et al. Aerial service robotics: The airobots perspective. In: 2012 2nd International Conference on Applied Robotics for the Power Industry (CARPI). IEEE; 2012. p. 64–9. doi:10.1109/CARPI.2012.6473361.

Mellinger D, Lindsey Q, Shomin M, Kumar V. Design, modeling, estimation and control for aerial grasping and manipulation. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2011. p. 2668–73. doi:10.1109/IROS.2011.6094871.

Mersha AY, Stramigioli S, Carloni R. Exploiting the dynamics of a robotic manipulator for control of uavs. In: 2014 IEEE international conference on robotics and automation (ICRA). IEEE; 2014. p. 1741–6. doi:10.1109/ICRA.2014.6907086.

Orsag M, Korpela C, Oh P. Modeling and control of mm-uav: Mobile manipulating unmanned aerial vehicle. Journal of Intelligent & Robotic Systems 2013;69(1):227–40. doi:10.1007/s10846-012-9723-4.

Orsag M, Korpela CM, Bogdan S, Oh PY. Hybrid adaptive control for aerial

manipulation. Journal of intelligent & robotic systems 2014;73(1):693–707. doi:10.1007/s10846-013-9936-1.

Pereira PO, Zanella R, Dimarogonas DV. Decoupled design of controllers for aerial manipulation with quadrotors. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2016. p. 4849–55. doi:10.1109/IROS.2016.7759712.

Pierri F, Muscio G, Caccavale F. An adaptive hierarchical control for aerial manipulators. Robotica 2018;36(10):1527–50. doi:10.1017/S0263574718000553.

Ruggiero F, Trujillo MA, Cano R, Ascorbe H, Viguria A, Peréz C, Lippiello V, Ollero A, Siciliano B. A multilayer control for multirotor uavs equipped with a servo robot arm. In: 2015 IEEE international conference on robotics and automation (ICRA). IEEE; 2015. p. 4014–20. doi:10.1109/ICRA.2015.7139760.

Siliciano B, Sciavicco L, Villani L, Oriolo G. Robotics: modelling, planning and control. New YorN, NY, USA: Springer 2010;:415–8doi:10.1007/978-1-84628-642-1.

Staub N, Mohammadi M, Bicego D, Prattichizzo D, Franchi A. Towards robotic magmas: Multiple aerial-ground manipulator systems. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2017. p. 1307–12. doi:10.1109/ICRA.2017.7989154.

Thomas J, Polin J, Sreenath K, Kumar V. Avian-inspired grasping for quadrotor micro uavs. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers; volume 55935; 2013. p. V06AT07A014. doi:10.1115/DETC2013-13289.

Xilun D, Pin G, Kun X, Yushu Y. A review of aerial manipulation of small-scale rotorcraft unmanned robotic systems. Chinese Journal of Aeronautics 2019;32(1):200–14. doi:10.1016/j.cja.2018.05.012.

Yang B, He Y, Han J, Liu G. Rotor-flying manipulator: modeling, analysis, and control. Mathematical Problems in Engineering 2014;2014. doi:10.1155/2014/492965.

Yang H, Lee D. Hierarchical cooperative control framework of multiple quadrotor-manipulator systems. In: 2015 ieee international conference on robotics and automation (ICRA). IEEE; 2015. p. 4656–62. doi:10.1109/ICRA.2015.7139844.

Younes YA, Drak A, Noura H, Rabhi A, Hajjaji AE. Robust model-free control applied to a quadrotor uav. Journal of Intelligent & Robotic Systems 2016;84(1):37–52. doi:10.1007/s10846-016-0351-2.