Modeling and Control of Aerial Manipulation Vehicle with Visual sensor ⋆

Somasundar Kannan, Miguel A. Olivares-Mendez, Holger Voos

* *SnT-University of Luxembourg,4 rue Alphonse Weicker,L-2721 Luxembourg (e-mail: {somasundar.kannan, miguel.olivaresmendez, Holger.Voos}@uni.lu)*

Abstract: Modeling and control of a Quadrotor with robotic arm which uses vision sensor is discussed. A quadrotor model coupled with a two link manipulator is first developed and then the integrated control mechanism is investigated. An Image Based Visual Servo system is introduced and then used with the aerial manipulator to successfully perform specific tasks of positioning and stabilization during manipulation.

Keywords: Autonomous Vehicle, Control, Robotic manipulators, Robot Dynamics, vision

1. INTRODUCTION

Aerial Manipulation has been an active area of research in recent times. Mainly because the active tasking of Unmanned Aerial Vehicles (UAVs) increases the employability of these vehicles for various applications. For active tasking one would consider grasping, manipulation, grasping and transporting etc.

These interesting applications have their own challenges and have been the subject of various research activities. In Pounds et al. [2011] the load disturbance on a helicopter introduced by a gripped object was studied experimentally and stability bounds were determined. Here the manipulator was a simple gripper which holds the object between the skids of the helicopter. In Marconi et al. [2011] a ducted fan UAV interacting with the environment was modelled and controlled. In this case the interaction was modelled as a simple contact.

In Lipiello and Ruggiero [2012a] and Lipiello and Ruggiero [2012b] Cartesian Impedence control and redundancy had been studied using Euler-Lagrange formulation. In Orsag et al. [2013b] and Khalifa et al. [2012] a Newton-Euler approach had been used to model and control a quadrotor based manipulator. In Orsag et al. [2013a] a Lyapunov based Model Reference Adaptive Control was used to stabilize a quadrotor with multi degree of freedom manipulator. But due to the complexity of the system only rigid body dynamics of the quadrotor were considered.

In Kondak et al. [2013] a vision based sensor was used to control a helicopter with manipulator experimentally using a simple gripper attached to the fuselage. In Ghadiok et al. [2011] indoor experiments were performed with a quadrotor equipped with a gripper and an IR camera was used to grip an object with LED placed on it.

Contrary to these above mentioned research the key contribution of this article is the development of an Aerial Manipulation system with visual sensing where the complete nonlinearity of the quadrotor is considered along with 2-link manipulator dynamics. A Newton-Euler approach has been used to develop the aerial manipulator and Image Based Visual Servo Systems has been used to perform control task successfully.

The paper is structured as follows. First the quadrotor dynamic is briefly described, then a Recursive Newton-Euler (RNE) method for a Floating base 2-link Manipulator is formulated using the classic fixed base RNE method and integrated with the quadrotor model. This quadrotor with manipulator model is used as a testbed to perform a series of tests and understand the dynamic coupling that exist in the complex model by performing basic tasks such as quadrotor position tracking and manipulation in hovering condition. An Image Based Visual Servo (IBVS) System is briefly introduced and used by the quadrotor to perform visual servoing.

2. QUADROTOR DYNAMICS

Here the quadrotor dynamics is dealt briefly. Let $\mathbb{I} = \{E_x, E_y, E_z\}$ be the inertial frame, $\xi = (x, y, z)$ be the origin of the body fixed frame $\mathbb{A} = \{E_1^a, E_2^a, E_3^a, \}$. The rotation matrix R is defined by $R : \mathbb{A} \rightarrow \mathbb{I}$. Here v and Ω are linear and angular velocities in the body reference frame $\mathbb{A}$. The model used here is based on Pounds et al. [2010], Corke [2013]:

$$\begin{bmatrix} \mathbf{m} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{\mathrm{v}} \\ \dot{\Omega} \end{bmatrix} + \begin{bmatrix} 0 \\ \Omega \times \mathbf{I}\Omega \end{bmatrix} = \begin{bmatrix} \mathbf{G} + T \\ \mathbf{Q} + M \end{bmatrix} \tag{1}$$

where we have $\mathbf{G} = \mathbf{m}ge_3$ with $e_3 = [0\ 0\ 1]^T$, $T = \sum_i t_i$, $\mathbf{Q} = \sum_i q_i$, $M = \sum_i m_i$ and for $i \in \{N, S, E, W\}$ we have rotor thrust $t_i = C_T \rho A r^2 \varpi_i^2 \mathbf{t}_{flap}$, torque $q_i = C_Q \rho A r^3 \varpi_i |\varpi_i| e_3$ and moment $m_i = t_i \times \mathbf{d}_i$.

The physical parameters are mass $\mathbf{m}$, Inertia $\mathbf{I}$, gravity g, ρ the air density, r is the rotor radius with A the area of the rotor disc, $\mathbf{d}_i$ is the rotor distance from the quadrotor

⋆ This work was Supported by the Fonds National de la Recherche, Luxembourg (Project Code:4926925)

center of mass with $\mathbf{d}_N = (0\ d\ h)$, $\mathbf{d}_S = (0\ -d\ h)$, $\mathbf{d}_E = (d\ 0\ h)$, $\mathbf{d}_W = (-d\ 0\ h)$. Here d is the arm length of the quadrotor and h is the height of the rotors above the Center of Gravity. The term $\mathbf{t}_{flap}$ is a function of longitudinal and lateral flapping angles [Pounds et al., 2010] and the constants C_T and C_Q are nondimensional thrust and torque coefficients respectively.

The motion of the quadrotor is obtained by the variation of forces and moments on the airframe which can be obtained as a function of rotor speeds $\left(\varpi_1^2\ \varpi_2^2\ \varpi_3^2\ \varpi_4^2\right)^T = \mathbf{A}^{-1}\left(T\ \tau_x\ \tau_y\ \tau_z\right)^T$. Where $\mathbf{A}$ is a function of known parameters [Mahony et al., 2012, Hamel et al., 2002], T is the total upward thrust as defined earlier and $M = (\tau_x, \tau_y, \tau_z)$ are the rolling, pitching and net reaction (yawing) torque applied to the airframe of the quadrotor.

3. ROBOTIC ARM WITH A FLOATING BASE

Different modeling methods exist in literature but the Newton-Euler method is used here as it helps us to systematically integrate the robotic arm to the quadrotor model which was also based on Newton-Euler formulation.

The manipulator to be considered is an open-chain mechanism consisting of N joints numbered from 1 to N connecting $N+1$ rigid links numbered from 0 to N. Link 0 is the base of the manipulator where as N carries the end-effector.

Remark 1. For ease of understanding to deal with the floating base manipulator, first a fixed base algorithm of Newton-Euler method is discussed and then it will be briefly mentioned how this can be converted into a mobile base algorithm.

Here first the Kinematics is discussed and then the Recursive Newton-Euler (RNE) Formulation is detailed.

3.1 Kinematics

A standard Denavit-Hartenberg convention is followed. A joint i connects link $i-1$ to link i and hence the joint i moves link i. A link is specified by its length a_i and twist α_i. Similarly joints are described through link offset d_i and joint angle θ_i. Here the coordinate frame $\{i\}$ is attached to the far end of the link i and the axis of joint i is aligned with the z_i-axis. In this convention the transformation from link coordinate frame $\{i-1\}$ to frame $\{i\}$ is defined by rotation and translation as [Corke, 2013]

$$^{i-1}A_i(\theta_i, d_i, a_i, \alpha_i) = T_{Rz}(\theta_i)T_z(d_i)T_x(a_i)T_{Rx}(\alpha_i) \quad (2)$$

which can be given as

$$^{i-1}A_i = \begin{pmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3)$$

3.2 Newton-Euler Dynamic Equations of Motion

We begin with defining certain link specific parameters such as m_i which is the mass of the link i, r_i is the location of the center of mass of link i with respect to origin of link i coordinates and I_i is the inertia matrix of the link i. A Recursive formulation is discussed here which consist of Forward computation of velocities and accelerations of each link and Backward computation of forces and moments in each joints. The Recursive Newton-Euler formulation for fixed based was discussed in Luh et al. [1980], Walker and Orin [1982] .This method is further detailed here but only the rotational joint will be considered since it is the one being used here.

Forward Recursion Here the angular/linear velocities and accelerations of each link is calculated recursively in terms of its preceding link starting from the base to the end effector. The initial conditions for the base links are v_0, $\dot{v}_0$ and ω_0, $\dot{\omega}_0$ and the following equations are calculated

$$\omega_i = \omega_{i-1} + z_{i-1}\dot{q}_i \quad (4)$$
$$\dot{\omega}_i = \dot{\omega}_{i-1} + z_{i-1}\ddot{q}_i + \omega_{i-1} \times z_{i-1}\dot{q}_1 \quad (5)$$

$$v_i = \omega_i \times p_i^* + v_{i-1} \quad (6)$$
$$\dot{v}_i = \dot{\omega}_i \times p_i^* + \omega_i \times (\omega_i \times p_i^*) + \dot{v}_{i-1} \quad (7)$$

$$\ddot{r}_i = \dot{\omega}_i \times r_i + \omega_i \times (\omega_i \times r_i) + \dot{v}_i \quad (8)$$
$$N_i = I_i\dot{\omega}_i + \omega_i \times (I_i \cdot \omega_i) \quad (9)$$
$$F_i = m_i\ddot{r}_i \quad (10)$$

where $p^* = [d_i\ a_i\sin\theta_i\ a_i\cos\theta_i]^T$ and the above variables are defined here

- ω_i angular velocity of link i
- $\dot{\omega}_i$ angular acceleration of link i
- v_i linear velocity of origin of link i coordinates with respect to link $i-1$ coordinates
- $\dot{v}_i$ linear acceleration of origin of link i coordinates with respect to link $i-1$ coordinates
- $\ddot{r}_i$ linear acceleration of link i center of mass
- F_i total force exerted on link i
- N_i total moment exerted on link i
- q_i joint variable (θ_i) at joint i

Backward recursion After the velocities and accelerations of the links are computed, the joint forces can be computed for each link starting from the end-effector to the base. The required equations are

$$n_i = n_{i+1} + N_i + (p_i^* + r_i) \times F_i + p_i^* \times f_{i+1} \quad (11)$$
$$f_i = F_i + f_{i+1} \quad (12)$$
$$\tau_i = z_{i-1} \cdot n_i \quad (13)$$

where the above variables are

- n_i moment exerted on link i by link $i-1$
- f_i forces exerted on link i by link $i-1$
- τ_i torque exerted by actuator at joint i (rotational)

These equations can be used to compute the joint torque by using velocities, accelerations, forces and moments in the local link coordinate.

Conversion to Mobile base Manipulator A number of methods exists for floating base manipulators for example see Featherstone [2008]. One of the simplest way is to initialize the velocities and accelerations v_0, ω_0, $\dot{v}_0$, $\dot{\omega}_0$ at some non-zero value. These velocities and accelerations are in turn transmitted from one link to another by forward recursion and would result in an additional resultant torque.

Equation of Motion For a series of links the equation of motion can be written in a general form as [Corke, 2013, Luh et al., 1980]

$$Q_m = M_m(q)\ddot{q} + C_m(q,\dot{q})\dot{q} + F_m(\dot{q}) + G_m(q) + J_m(q)^T w_e \tag{14}$$

where $q,\ \dot{q}$ and $\ddot{q}$ are respectively the vector of generalized joint coordinates, velocities and accelerations, M_m is the joint-space inertia matrix, C_m is the Coriolis and centripetal coupling matrix, F_m is the friction force, G_m is the gravity loading and Q_m is the vector of generalized actuator forces at the generalized coordinate q. The last term gives the joint force due to wrench w_e applied at the end-effector and J_m is the manipulator Jacobian. Different methods exist to compute the above inverse dynamic equation (14) but the Method-1 of Walker and Orin [1982] will be used here. Without going into further details this method is based on the Recursive Newton Euler method discussed earlier.

4. QUADROTOR WITH ROBOTIC ARM

The coupled equations of motion for the quadrotor with manipulator is

$$\begin{bmatrix}\mathbf{m} & 0\\ 0 & \mathbf{I}\end{bmatrix}\begin{bmatrix}\dot{\mathrm{v}}\\ \dot{\Omega}\end{bmatrix} + \begin{bmatrix}0\\ \Omega\times\mathbf{I}\Omega\end{bmatrix} = \begin{bmatrix}\mathbf{G}+T\\ \mathbf{Q}+M\end{bmatrix} + \begin{bmatrix}f\\ n\end{bmatrix}. \tag{15}$$

where the vector $[f\ n]^T$ are forces and moments measured at the base of the manipulator that is exerted on the quadrotor, since the manipulator is attached to the base of the quadrotor. For the same reason the base of the manipulator has $v_0 = \mathrm{v},\ \dot{v}_0 = \dot{\mathrm{v}}$ and $\omega_0 = \Omega,\ \dot{\omega}_0 = \dot{\Omega}$

In order to analyse and study the complex dynamic behaviour of a quadrotor with a manipulator, a benchmark problem was developed by utilizing an existing highly nonlinear model with flapping characteristics such as that of Pounds et al. [2010] and the robotics simulation environment of Corke [2013] was used to implement the robotic arm and quadrotor model. The quadrotor of Pounds et al. [2010] is a 4 Kg vehicle with a maximum payload of 1 Kg. Hence it was a suitable choice for implementation of large manipulator system. A two link manipulator was chosen here with length $a_1 = a_2 = 0.5$ m and mass $m_1 = m_2 = 0.1$ Kg with inertia matrix $I = ma^2/12\ \mathrm{diag}(0,1,1)$. Here $m = m_1 = m_2$ and $a = a_1 = a_2$. Two types of problems that are visibly clear will be dealt immediately, but not before introducing the basic control structure.

4.1 Basic Control structure

The basic control structure of the quadrotor can be seen in Figure-1 which is a hierarchical structure. The control of manipulator is also added to the structure due to the dynamic coupling that exist between the two systems. Starting with the quadrotor control a clear time scale separation is evident from the structure. The inner loop which is faster attitude rate control uses a PD control for coupled pitch and roll dynamics. A simple controller could then be

$$U_{pr} = -K_p\varepsilon_{pr} - K_d\dot{\Theta}_{pr} \tag{16}$$

where we have $\varepsilon_{pr} = \Theta_{pr} - \Theta^*_{pr}$, with Θ^*_{pr} as the pitch and roll demand, U_{pr} gives us pitch (τ_y) and roll torque (τ_x). A similar PD control is also used for the yaw control loop

$$U_{yaw} = -K_p\varepsilon_{yaw} - K_d\Omega_z \tag{17}$$

where $\varepsilon_{yaw} = \psi^* - \psi$ is the yaw error with yaw demand ψ^*.

The outer-loop consists of a $x,\ y$ position control with PD type structure, the output of which is the pitch and roll demand [Corke, 2013]

$$\Theta^*_{pr} = K_1(\xi^*_{xy} - \xi_{xy} - K_2 v_{xy}) \tag{18}$$

where $\{x_e, y_e\} = \xi^*_{xy} - \xi_{xy}$ is the position error and v_{xy} is the position rate. For the height control an independent PD control

$$T = -K_p z_e - K_d v_z + w_0 \tag{19}$$

is used where $z_e = \xi^*_z - \xi_z$ is the z position error and v_z is the z position rate, along with compensation w_0 for the weight of the quadrotor.

For the manipulator control problem an independent joint control method was used preferable with PI-D configuration such as

$$U_m = (-K_p - \frac{1}{s}K_i)\tilde{q} - K_d\dot{q} + G_m(q) \tag{20}$$

where $\tilde{q}$ is a joint angle error, $G_m(q)$ is a gravity load torque which can be approximately added if computation is not possible [Corke, 2013].

The controller gains that are used here to perform the preliminary tests are: (1) For pitch-roll attitude control: $K_p = 15\times10^2,\ K_d = 0.1$, (2)$x,\ y$ position control: $K_1 = [-0.15, 0.15],\ K_2 = 2$,(3) yaw control: $K_p = 100,\ K_d = 1$ (4)height (z-position) control: $K_p = 100,\ K_d = 1$. For the manipulator arm control the controller gains are:$K_p = [100, 100],\ K_i = [100, 100],\ K_d = [2, 2]$

It must be emphasised that at this stage of discussion no special effort has been made to choose optimal controller parameters for quadrotor. Since the axis of rotation of the manipulator is in y-axis and the length is aligned in the x-axis of the quadrotor the roll-pitch dynamic was initially stabilized with larger gains before beginning the manipulator control.

Next two types of control problems are discussed which we come across in the aerial manipulation system.

4.2 Position control of quadrotor with free manipulator (Case-1)

The first problem is the position control of quadrotor with no controls applied to the manipulator. The initial condition of the quadrotor position is $[\xi^0_x, \xi^0_y, \xi^0_z] = [0, -1, -4]$ and for the manipulator is $[q_1, q_2] = [0.5, -0.5]$ The control task for the quadrotor is to move to the position $[\xi^*_x, \xi^*_y, \xi^*_z] = [1, -1, -4]$. A simple PD type control loop discussed above for the quadrotor is employed without any control effort to the manipulator and the respective position error and attitude response of the quadrotor is seen in Figure-2.

In Figure-2 from the position error plot we can see that asymptotically $x_e \to 0,\ \ y_e \cong 0$ where as $z_e < 0$. One of the reasons for $z_e < 0$ is that part of the upward thrust is used to displace the quadrotor in the x-direction by pitching downwards which is also visible in the Figure-2 (middle plot), where we can also see that the yaw angle

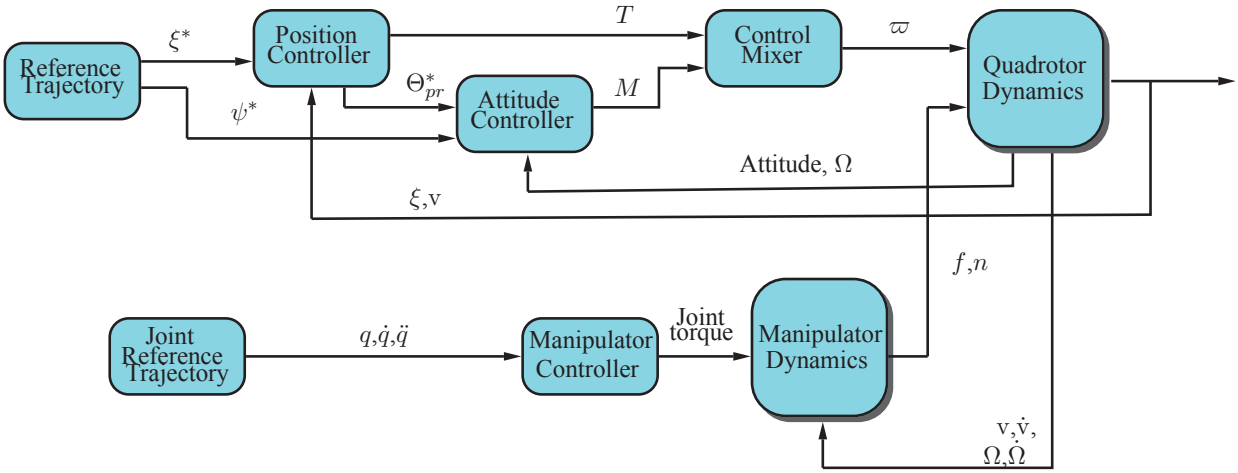

Fig. 1. Control scheme of Quadrotor with manipulator. Here attitude controller is the inner loop and position controller is the outer loop in the quadrotor control. The manipulator is controlled separately using independent joint control.

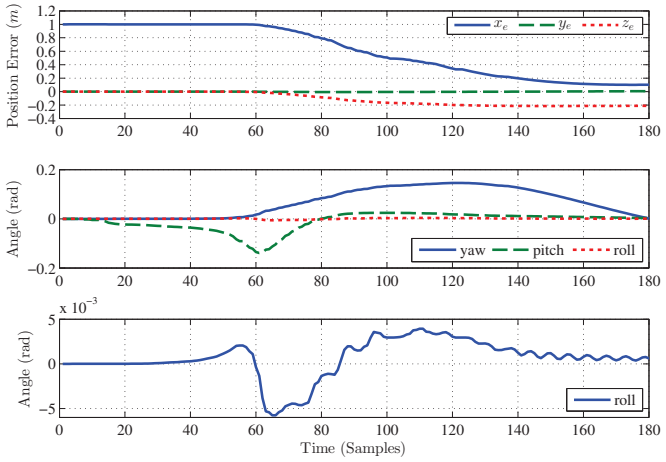

Fig. 2. (Case-1)Top:Position Error, Middle: Attitude, Bottom: Roll (Magnified)

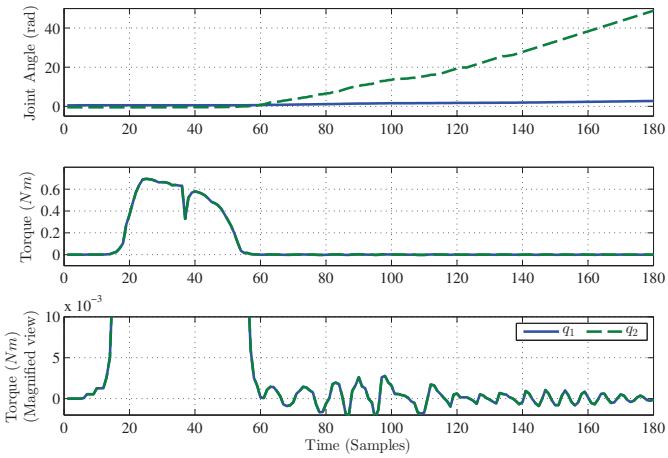

Fig. 3. (Case-1)Top: Joint Angles (radians), Middle:Joint Torque, Bottom: Magnified Joint Torque

deviates once again explaining the necessity to tune the yaw and height control loop. The minor oscillations in the roll angle can be seen in Figure-2 (Bottom) which is the axis of translation here.

In Figure-3 we can see the uncontrolled manipulator joint angles $\{q_1, q_2\}$ with the torque at the joints $\{q_1, q_2\}$ which is generated purely due to the motion of quadrotor. The quadrotor transmits the velocities and accelerations to the 2-link manipulator (since the manipulator is directly connected to the center of gravity of the quadrotor).

The respective forces and moments that are generated due to the motion of the quadrotor with uncontrolled manipulator can be seen in Figure-4. Here when considering the force vector we can see that $f_z = 2 \times 9.82 \times 0.1$ which is due to the weight of the 2-link manipulator and also f_y is due to the revolute joint which is aligned along the y-axis which can also be noted in the moment n_y along the y-axis. Since the motion of the quadrotor is along x-axis with strong coupling between the roll-pitch dynamic the oscillations in the roll moment n_x can be seen in the roll attitude in Figure-2.

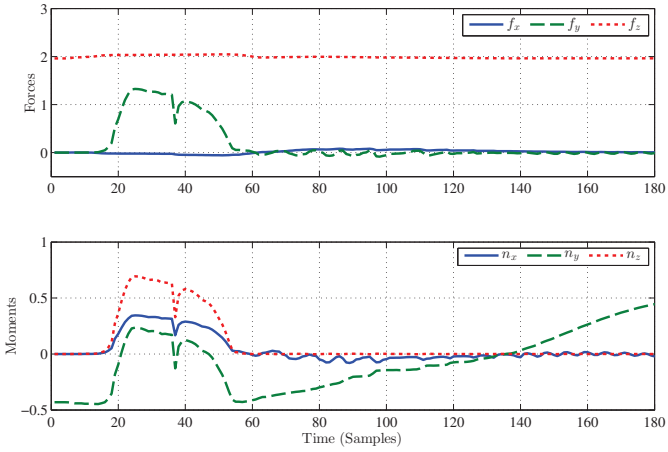

Fig. 4. (Case-1)Top: Forces exerted by the manipulator to the quadrotor, Bottom: Moments exerted by the manipulator to the quadrotor

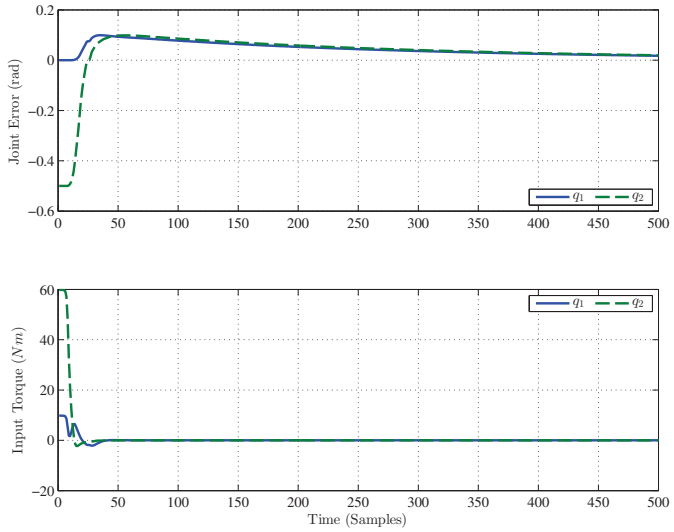

Fig. 5. (Case-2)Top: Joint Angle error (radians), Bottom:Joint Input Torque

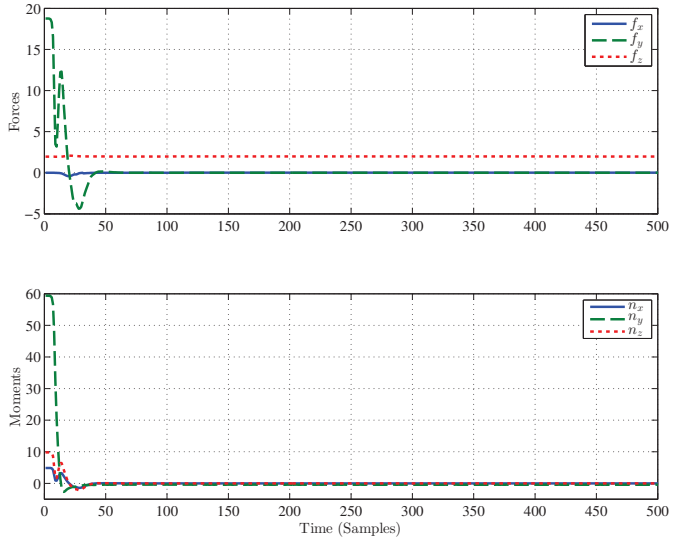

Fig. 6. (Case-2)Top: Forces exerted by the manipulator to the quadrotor, Bottom: Moments exerted by the manipulator to the quadrotor

4.3 Manipulator control with quadrotor in hover (Case-2)

In this scenario the task is to control the manipulator while maintaining the hover condition. The manipulator positions are initialized at $[q_1, q_2] = [0.5, -0.5]$ and the task is to attain a constant reference $[q_1, q_2] = [0.5, 0]$. The joint angle error and input control torque can be seen in Figure-5. The forces and moments exerted by the manipulator on the quadrotor can be seen in Figure-6 where the constant f_z due to the weight of the manipulator is evident in the height error z_e which is deviating from zero (Figure-7). Similar effect can be seen in the yaw attitude of the quadrotor (Figure-7). These observations remind us the type of effect the payload can have on the dynamics of the quadrotor.

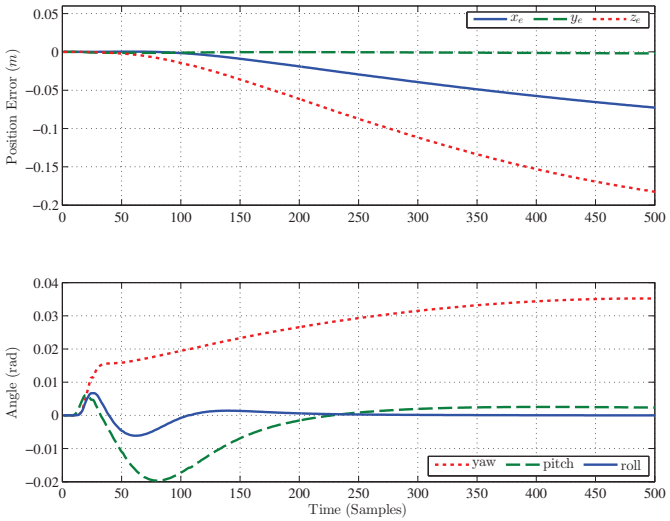

Fig. 7. (Case-2)Top:Position Error, Bottom: Attitude

5. VISUAL SERVOING

The fundamental approaches to Vision based control include Position Based Visual Servo (PBVS) and Image Based Visual Servo (IBVS). In the current research IBVS was used which is discussed next.

5.1 Image based Visual Servoing

The advantage of IBVS (as compared to PBVS) is that the estimation of relative pose is not required and it is implicit in the image feature information [Corke, 2013].

As discussed in Corke [2013](page-486) we have the spherical optical flow equation $\begin{pmatrix}\dot{\theta} & \dot{\phi}\end{pmatrix}^T = \mathbf{J}_{p,s}(\theta, \phi, R)\nu$ where we have ν as spatial velocity of the camera, the image Jacobian $\mathbf{J}_{p,s}(\theta, \phi, R)$ is given as

$$\mathbf{J}_{p,s} = \begin{pmatrix} -\frac{\cos\phi\cos\theta}{R} & -\frac{\sin\phi\cos\theta}{R} & -\frac{\sin\theta}{R} & \sin\phi & -\cos\phi & 0 \\ \frac{\sin\phi}{R\sin\theta} & -\frac{\cos\phi}{R\sin\theta} & 0 & \frac{\cos\phi\cos\theta}{\sin\theta} & \frac{\sin\phi\cos\theta}{\sin\theta} & -1 \end{pmatrix}. \quad (21)$$

Here $\mathbf{p} = (\theta, \phi)$ are the spherical coordinates and $R = \sqrt{(X^2 + Y^2 + Z^2)}$ is the distance from the camera origin to the world point.

The control law using the IBVS system is $\nu = J^+\dot{p}^*$, where $\dot{p}^*$ is the desired velocity of the features in the $\phi\theta$-space. A proportional control can be used to minimize the feature error $\dot{p}^* = \lambda(p^* \ominus p)$, where $\ominus$ is a modulo subtraction [Corke, 2013]. The quadrotor is an underactuated vehicle so we have $\nu = [\mathrm{v}_x^*, \mathrm{v}_y^*, \mathrm{v}_z^*, \Omega_z^*]$ which is the velocity demand to the quadrotor. The internal control loops of the visual servo system consists of a combination of P,PI and PD. The x, y velocity demand $\mathrm{v}_{xy}^* = \{\mathrm{v}_x^*, \mathrm{v}_y^*\}$ is minimized using a PI type controller: $\Theta_{pr}^* = (K_p + K_i\frac{1}{s})\varepsilon_\mathrm{v}$, where $\varepsilon_\mathrm{v} = \mathrm{v}^* - \mathrm{v}$ with output Θ_{pr}^* being the pitch-roll demand. This is followed by the attitude rate loop given by (16). A Proportional control $U_{yaw} = -K_p(\Omega_z^* - \Omega_z)$ is used to meet the yaw rate demand. A PI type controller $T = (-K_p - K_i\frac{1}{s})\varepsilon_{\mathrm{v}_z} + \mathrm{w}_0$ is used for the height rate loop where we have $\varepsilon_{\mathrm{v}_z} = \mathrm{v}_z^* - \mathrm{v}_z$ with T being the total upward thrust and w_0 being the weight of the quadrotor.

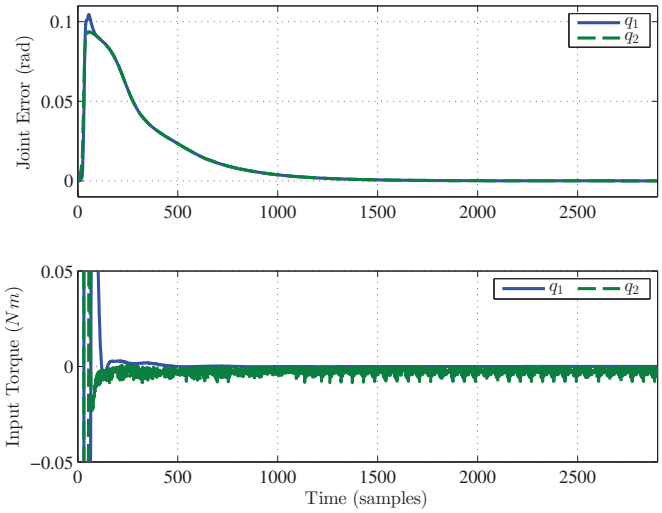

Fig. 8. (VS: Case-1) Vision based position tracking: Joint error and Torque

For the control of the robotic arm the controller is exactly the same as the previous case (20).

5.2 Aerial manipulation with visual servoing

Based on the above discussed visual servo system two types of tasks will be evaluated. The two tasks include positioning to a reference point and stabilization during manipulation at a specific reference point.

The controller gains that are used here to perform the preliminary tests are:(1)For x, y velocity control $K_p = [-0.15, 0.15]$, $K_i = [-0.1, 0]$ (2) pitch-roll attitude rate control: $K_p = 15 \times 10^2$, $K_d = 0.1$, (3) yaw rate control: $K_p = 15 \times 10^2$ (4)height (z-position) rate control: $K_p = K_p = 15 \times 10^2$, $K_i = 100$. For the manipulator arm control the controller gains are:$K_p = [100, 100]$, $K_i = [100, 100]$, $K_d = [2, 2]$. The outermost loop which is the visual servoing, a proportional control gain $\lambda = 1$ is used.

Positioning using visual sensor (VS: Case-1) Here the quadrotor starts at position $\{0, -1, -4\}$ and has to reach the point $\{0.5, -1, -4\}$ while holding the manipulator pose at $\{0.5, -0.5\}$. The manipulator joint error and torque can be seen in Figure-8. The effort in Figure-8 is the torque required to hold the manipulator at a given position during displacement of the quadrotor. The feature error and camera velocity can be seen in Figure-9. Here we can observe that the feature error norm tends to zero asymptotically, and the respective x-direction velocity is also visible.

Vision based stabilization during manipulation (VS: Case-2) Here the purpose is to stabilize the quadrotor at position $\{0, -1, -4\}$ while the manipulator changes the end effector position from $\{0.5, -0.5\}$ to $\{0.5, 0\}$. The manipulator joint error and torque required to attain the pose can be seen in Figure-10. The feature error norm and camera velocity can be seen in Figure-11. In Figure-10 we can see the higher torque required to move the link-2 to the specified position.

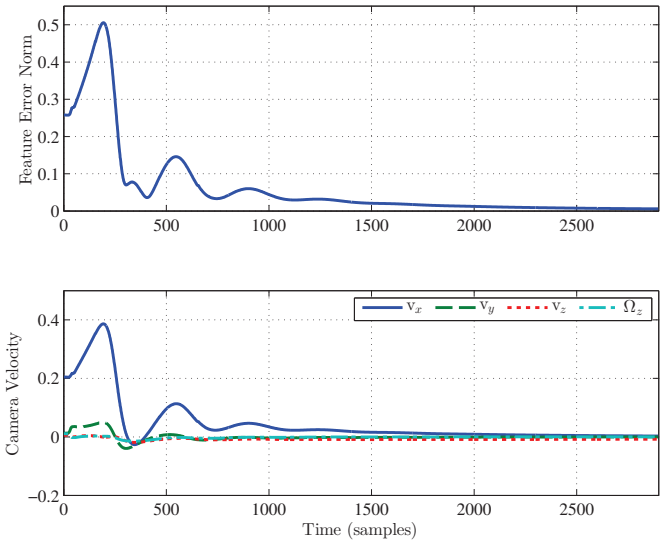

Fig. 9. (VS: Case-1) Vision based position tracking:Norm of the feature error and Camera velocity

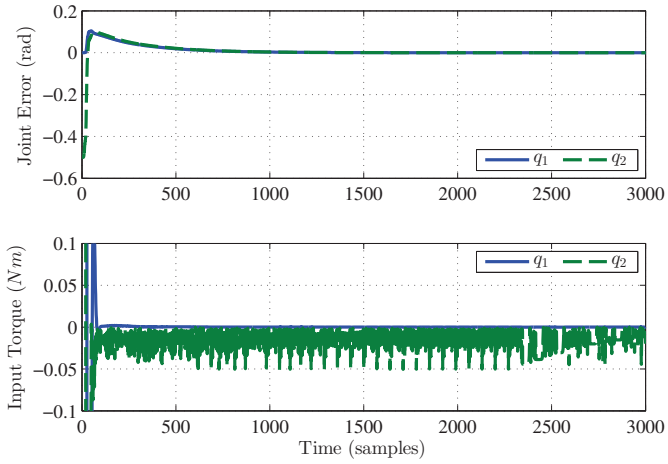

Fig. 10. (VS: Case-2) Vision based hovering while manipulation: Joint error and Torque

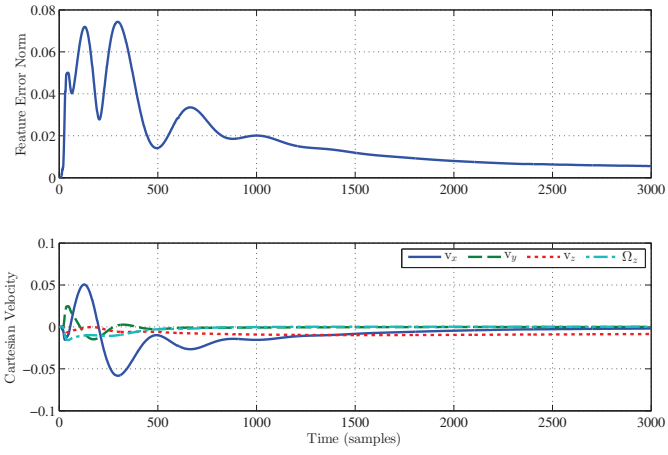

Fig. 11. (VS: Case-2) Vision based hovering while manipulation: Norm of the feature error and Camera velocity

6. CONCLUSION

In this research an aerial manipulation system was successfully developed using Recursive Newton Euler method. The dynamic coupling between the quadrotor and the developed two link manipulator was briefly studied while performing specific task. A visual servo system was also introduced and used with the quadorotor to perform precise positioning and hover stabilization with manipulation task. The future perspective of this research is to develop a single visual servo system which can be used for control of robotic arm and quadrotor together.

REFERENCES

P. I. Corke. *Robotics vision and Control:Fundamental Algorithms in MATLAB*. Springer, 2013.

R. Featherstone. *Rigid body dynamics algorithms*. Springer, 2008.

V. Ghadiok, J. Goldin, and W. Ren. Autonomous indoor aerial gripping using a quadrotor. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4645–4651, september 2011.

T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski. Dynamic modelling and configuration stabilization for an x-4 flyer. In *IFAC 15th Triennial World Conress*, 2002.

A. Khalifa, M. Fanni, A. Ramadan, and A. Abo-Ismail. Modeling and control of a new quadrotor manipulation system. In *2012 First International Conference on Innovative Engineering Systems (ICIES)*, pages 109–114, 2012.

K. Kondak, K. Kreiger, A. Albu-Schaeffer, M. Schwarzbach, M. Laiacker, I. Maza, A. Rodriguez-Castano, and A. Ollero. Closed-loop behaviour of an autonomous helicopter equipped with a robotic arm for aerial manipulation tasks. *International Journal of Advanced Robotic Systems*, 10, 2013.

V. Lipiello and F. Ruggiero. Cartesian impedence control of a uav with a robotic arm. In *10th IFAC Symposium on Robot Control*, 2012a.

V. Lipiello and F. Ruggiero. Exploiting redundancy in cartesian impedance control of UAVs equipped with a robotic arm. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3768–3773, 2012b.

J. Y. S. Luh, M. W. Walker, and R. P. C. Paul. On-line computational scheme for mechanical manipulators. *ASME Journal of Dynamic Systems, Measurements and Control*, 102(2):69–76, 1980.

R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicle: Modeling, estimation of quadrotor. *IEEE Robotics and Automation Magazine*, 19(3):20–32, 2012.

L. Marconi, R. Naldi, and L. Gentili. Modelling and control of a flying robot interacting with the environment. *Automatica*, 47:2571–2583, 2011.

M. Orsag, C. Korpela, S. Bogdan, and P. Oh. Lyapunov based model reference adaptive control for aerial manipulation. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 966–973, may 2013a.

M. Orsag, C. Korpela, and P. Oh. Modeling and control of mm-uav: Mobile manipulating unmanned aerial vehicle. *Journal of Intell Robot Sys*, 69:227–240, 2013b.

P. Pounds, R. Mahony, and P. Corke. Modelling and control of large quadrotor robot. *Control Engineering Practice*, 18:691–699, 2010.

P. E. I Pounds, D. R. Bersak, and A. M. Dollar. Grasping from the air: Hovering capture and load stability. In *2011 International conference on Robotics and Automation*, pages 2491–2498, 2011.

M. W. Walker and D. E. Orin. Efficient dynamic computer simulation of robotic mechanisms. *ASME Journal of Dynamic Systems, Measurements and Control*, 104(3): 205–211, 1982.