Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

Feedback linearization control of quadrotor with tiltable rotors under wind gusts

Abdul-Wahid A. Saif*

Systems Engineering Department, KFUPM, Dhahran 31261, Saudi Arabia

ARTICLE INFO

Article history:
Received 6 June 2017
Received in revised form
15 August 2017
Accepted 10 September 2017

Keywords:
Quadrotor
UAV
Tilting rotor modeling
Feedback linearization
Optimization
Wind gust

ABSTRACT

Quadrotors are popular unmanned aerial vehicles that have a plethora of applications for civilian and military purposes. This is due to their superior agility and maneuverability which widens the span of applications. In this paper, a feedback linearization controller is designed to control all the states of the over actuated quadrotor with tilting rotors that was developed by the author and his colleagues. The controller is introduced in a novel approach that overcomes the problem of nonlinear inputs and that decouples the system into completely two independent subsystems while rejecting wind gusts. In addition, an optimization algorithm is introduced to choose among the possible sets of inputs based on energy consumption minimization. The results demonstrate that the quadrotor with tilted rotor can effectively attain the desired trajectory in the presence of wind disturbance.

1. Introduction

Quadrotors have been extensively used for various civilian and military purposes. Examples of such applications are surveillances, traffic monitoring, rescue missions, patrolling forests in case of fire outbreak, warfare, and other risky missions. Modeling for the conventional quadrotor exists in numerous literatures for instance in Bouabdallah et al. (2007), Pounds et al. (2010), and Voos (2009). These model structures possess limitations regarding the orientation coupling, because of insufficient inputs, hence making it incongruous for some particular tasks. In light of this, different modeling schemes have been employed Ryll et al. (2012) and Oner et al. (2008) in quadrotors with tilt wing to address some deficiencies of the conventional quadrotor to improve the actuation capacity. Additional inputs have only been successfully aimed at decoupling the quadrotor orientation either to achieve forward flight mode or moving sideways while hovering as the case may be. This is particularly useful in applications requiring more maneuverability.

However, a thrust tilting approached Elfeky et al. (2013, 2014) and Hua et al. (2013) were applied to a conventional quadrotor to decouple the orientation, with the aim of reducing additional input. Each rotor is designed in a way such that it can tilt around two axes with respect to fixed body frame. In this design, the number of inputs becomes twelve instead of four in the conventional quadrotors. Therefore, it is enough to have six inputs to fully actuate this modified quadrotor. Also twelve inputs may be used to impose arbitrary trajectories and other requirements such as disturbance rejection. With this design, each of the twelve states (outputs) (6 positions/orientations - 6 translational/rotational speeds) can be controlled independently and freely. In addition, the novelty of this system is that, it can perform all the desired control objectives with half of its actuators faulty. Various control techniques have been used successfully to control conventional quadrotor within the range of their capabilities. In Bouabdallah et al. (2007), Pounds et al. (2010), and Ryll et al. (2012), linear control laws have been extensively employed. However in Voos (2009), Altug et al. (2005), Senkul and Altug (2014), Morel and Leonessa (2006), Liu et al. (2013), Li and Wang (2013) and Saif et al. (2012) non-linear control methods were also successfully employed. For the quadrotor with tilt-wing different control methods have been applied for instance, an output feedback control in Ryll et al. (2012), LQR control in Oner et al. (2008) and PID control in Salih et al. (2010) and Elfeky et al. (2013) to achieve different flight modes.

In this work, we shall exploit the capability of quadrotor with tilted rotors to tackle the issues of orientation decoupling. Each rotor possesses two degrees of freedom, thus increasing the number of

* Corresponding Author.
Email Address: awsaif@kfupm.edu.sa

https://doi.org/10.21833/ijaas.2017.010.021
2313-626X/© 2017 The Authors. Published by IASE.

input to twelve. Although, six inputs are sufficient for full actuation, additional inputs can be utilized for different flight modes and wind gust rejection. Linearized feedback controllers are used to control the quadrotor*.* This paper is organized as follows. Section 2 presents the dynamic model of quadrotor with two degree of freedoms tilting rotors. The wind gust modelling is stated in Section 3. The design of feedback controllers based on feedback linearization is given in Section 4. Section 5, discusses the optimization techniques that are used to select 3 inputs out of 12 such that an objective function related to the power consumption is minimized. Simulation results that show the effectiveness of the designed controller are shown in Section 6. Finally, the paper conclusion is given in section 7.

2. System model

2.1. Frames and rotation matrices

The quadrotor consist of five rigid bodies that are connected together and are in relative motion around them. These five bodies are the quadrotor body itself Band four propellers $\mathrm{P_i}$ attached to the body. Denote the world inertial frame by $\mathrm{F_E}$: $\{\mathrm{O_E}, \mathrm{X_E}, \mathrm{Y_E}, \mathrm{Z_E}\}$ and $\mathrm{F_B}$: $\{\mathrm{O_B}, \mathrm{X_B}, \mathrm{Y_B}, \mathrm{Z_B}\}$ be the quadrotor body frame attached to its center of gravity. In addition, the frames of the rotors are taken to be parallel to each other and parallel to the quadrotor body frame and are given by $\mathrm{F_{P_i}}$: $\{\mathrm{O_{P_i}}, \mathrm{X_{P_i}}, \mathrm{Y_{P_i}}, \mathrm{Z_{P_i}}\}, \mathrm{i} = 1, \dots, 4$.

The orientation of each of the rotors is controlled by two rotations with respect to the rotor-fixed frame; α_i, a rotation about $\mathrm{Y_{P_i}}$, and β_i, about $\mathrm{Z_{P_i}}$. This rotation creates another rotating frame for the rotors denoted by $\mathrm{F_{\bar{P}_i}}$: $\{\mathrm{O_{\bar{P}_i}}, \mathrm{X_{\bar{P}_i}}, \mathrm{Y_{\bar{P}_i}}, \mathrm{Z_{\bar{P}_i}}\}$, $\mathrm{i} = 1, \dots, 4$ as shown in Fig. 1.

Unlike the conventional quadrotor, in this proposed design (Elfeky et al., 2013, 2014), when the rotors are aligned along$\mathrm{Z_{P_i}}$, say, rotor 1 and rotor 4 are assumed to rotate counter-clock-wise CCW, while rotor 2 and rotor 3 rotate clock-wise CW as shown in Fig. 2. Let the forward direction be along$\mathrm{X_B}$. This design can serve for many critical applications. The fact that the motions are completely decoupled and that the quadrotor doesn't need to pitch to go forward nor to roll for lateral motions; this fact makes our design very suitable for sensitive payload as it provides a very smooth ride. Surveillance and monitoring could be improved as the quadrotor can fly at precise attitudes with precise speeds and orientations which make it very suitable for military application.

Let $\mathrm{R_{\bar{P}_i}^{P_i}}$ be the rotational matrix from the rotors-rotating frame $\mathrm{O_{\bar{P}_i}}$ to the rotors-fixed frame$\mathrm{O_{P_i}}$. Since the rotors' fixed frames$\mathrm{O_{P_i}}$ are parallel to the body-fixed frame $\mathrm{O_B}$ at the center of gravity, then

$$\mathrm{R_{\bar{P}_i}^{P_i}} = \mathrm{R_{\bar{P}_i}^{B}} = \begin{bmatrix} c\beta_i c\alpha_i & -s\beta_i & c\beta_i s\alpha_i \\ s\beta_i c\alpha_i & c\beta_i & s\beta_i s\alpha_i \\ -s\beta_i & 0 & c\alpha_i \end{bmatrix} \tag{1}$$

where c(.) and s(.)denote cos(.) and sin(.) respectively.

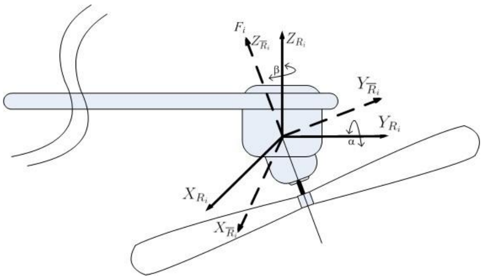

Fig. 1: Tilt angles of the rotor w.r.t fixed body frames

A full rotation of the quadrotor body with respect to the inertial frame can be described by three consecutive rotations about the three body axes, roll rotation Φ about the body x-axis, pitch rotation Θ about the body y-axis and yaw rotation Ψabout the body z-axis. Then $\mathrm{R_B^E}$ is the body transformation matrix with respect to the earth inertial frame, and is given by

$$\begin{aligned} \mathrm{R_B^E} &= \mathrm{R_\Psi} \cdot \mathrm{R_\Theta} \cdot \mathrm{R_\Psi} \\ &= \begin{bmatrix} c\Psi & -s\Psi & 0 \\ s\Psi & c\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\Theta & 0 & s\Theta \\ 0 & 1 & 0 \\ -s\Theta & 0 & c\Theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\Phi & -s\Phi \\ 0 & s\Phi & c\Phi \end{bmatrix} \\ &= \begin{bmatrix} c\Psi c\Theta & -s\Psi c\Phi + c\Psi s\Theta s\Phi & s\Psi s\Phi + c\Psi s\Theta c\Phi \\ s\Psi c\Theta & c\Psi c\Phi + s\Psi s\Theta s\Phi & -c\Psi s\Phi + s\Psi s\Theta c\Phi \\ -c\Theta & c\Theta s\Phi & c\Theta c\Phi \end{bmatrix} \end{aligned} \tag{2}$$

The relationship between the body-fixed angular velocity vector $[\mathrm{p} \quad \mathrm{q} \quad \mathrm{r}]^{\mathrm{T}}$ and Euler-Angles rates $[\dot{\Phi} \quad \dot{\Theta} \quad \dot{\Psi}]^{\mathrm{T}}$ is given by

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -c\Theta \\ 0 & c\Phi & s\Phi c\Theta \\ 0 & -s\Phi & c\Phi c\Theta \end{bmatrix} \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} \tag{3}$$

2.2. Quadrotor dynamics

Let ω_i be the rotational speed of the rotor i. Then the lifting thrust is given as $b\omega^2{}_i$ and the drag moment is given as $d\omega^2{}_i$, where b and d are the thrust and drag moment constants respectively. Therefore, the thrust components (force) of the i^th^ rotor at the body C.G. are then given by

$$\mathrm{F_i} = \begin{bmatrix} c\beta_i c\alpha_i & -s\beta_i & c\beta_i s\alpha_i \\ s\beta_i c\alpha_i & c\beta_i & s\beta_i s\alpha_i \\ -s\beta_i & 0 & c\alpha_i \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ b\omega_i^2 \end{bmatrix} \tag{4}$$

The moments of a titled rotor consist of the drag moment and the moments generated by the thrust components. These two components are expressed as

$$\mathrm{M_i} = \begin{bmatrix} c\beta_i c\alpha_i & -s\beta_i & c\beta_i s\alpha_i \\ s\beta_i c\alpha_i & c\beta_i & s\beta_i s\alpha_i \\ -s\beta_i & 0 & c\alpha_i \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ b\omega_i^2 \delta(i) \end{bmatrix} + \mathrm{r_i} \times \mathrm{F_i} \tag{5}$$

where δ(i) = [1,1,−1,−1] and r_i is the vector from center of gravity to the reference point of the rotors, i.e. $\mathrm{r_1} = [\mathrm{l}, 0, -\mathrm{h}]$, $\mathrm{r_2} = [0, \mathrm{l}, -\mathrm{h}]$, $\mathrm{r_3} = [-\mathrm{l}, 0, -\mathrm{h}]$, $\mathrm{r_4} = [0, -\mathrm{l}, -\mathrm{h}]$ where h and l are the vertical and horizontal displacements from the center of gravity to the rotors respectively. Denote by η the quadrotor position vector and by Ω the body angular velocities vector which are given by$\eta = [x \quad y \quad z]^T$and $\Omega = [p \quad q \quad r]^T$.Then the summation of forces acting on the quadrotor body are given by the dynamic equation:

$$m\ddot{\eta} = mg_z - K\dot{\eta} + R_B^E \sum_{i=1}^{4} F_i + D \tag{6}$$

where m is the mass of the quadrotor, $g_z = [0 \ 0 \ g]^T$ and K is the matrix of drag constants, and is given by $K = \mathrm{diag}\{K_1 \quad K_2 \quad K_3\}$and $D = [d_1 \quad d_2 \quad d_3]^T$ is the disturbances components of wind gust that will be discussed later. The rotation dynamic equation is then given by:

$$I\dot{\Omega} = -(\Omega \times I\Omega) - M_G - M_f + \sum_{i=1}^{4} M_i + M_d \tag{7}$$

where I is the inertia matrix of the quadrotor, and is given by $I = \mathrm{diag}\{I_x \quad I_y \quad I_z\}$.

M_G is the gyroscopic forces, and is given by

$$M_G = \sum_{i=1}^{4} I_R (\Omega \times \bar{\omega}_i)\, \delta(i) \tag{8}$$

I_R is the rotor moment of inertia and

$$\bar{\omega}_i = \begin{bmatrix} c\beta_i c\alpha_i & -s\beta_i & c\beta_i s\alpha_i \\ s\beta_i c\alpha_i & c\beta_i & s\beta_i s\alpha_i \\ -s\beta_i & 0 & c\alpha_i \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \omega_i \end{bmatrix} \tag{9}$$

M_d represents a random disturbance moment, M_f is the drag/friction moments with K_4, K_5 and K_6 representing the drag coefficients (Habib et al., 2011) and is given by:

$$M_f = [K_{4p} \quad K_{5q} \quad K_{6r}]^T$$

and

$$M_d = [m_{dp} \quad m_{dq} \quad m_{dr}]^T.$$

Finally, the equations of motion are

$$\begin{bmatrix} \ddot{\eta} \\ \dot{\Omega} \end{bmatrix} = \begin{bmatrix} g_z - \frac{K}{m}\dot{\eta} + \frac{R_B^E}{m} \sum_{i=1}^{4} F_i + D \\ -(I^{-1}\Omega \times \Omega) - I^{-1}M_G - I^{-1}M_f + I^{-1} \sum_{i=1}^{4} M_i + I^{-1}M_d \end{bmatrix} \tag{10}$$

where F_i and M_i are given by Eqs. 4 and 5 respectively.

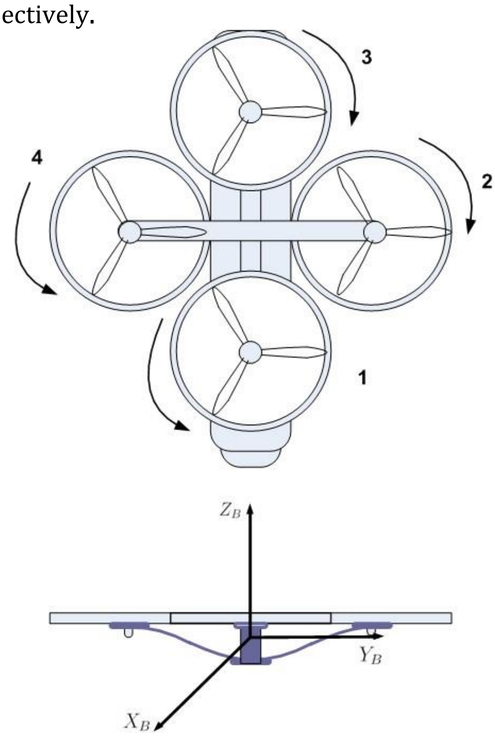

Fig. 2: Quadrotor with each rotor tilting about two axes

3. Wind gust modeling

The modeling approach used in this work is based on Solovyev et al. (2015). This approach takes the following into consideration:

- The effect of wind velocity change (increasing or decreasing).
- Gust duration.
- Wind velocity change with respect to altitude.
- Wind direction change.

The wind force expression, depending on the effective influence area on the quadrotor, is also derived. However, this model is suitable, based on the finding in Habib et al. (2011), where the effect of wind gust in small quadrotors is significantly correlated to the rate of increase or duration of a gust rather than the magnitude of the gust.

At any point in time the effect of wind felt at the different elements of the body is assumed to have equal magnitude and direction. The wind model velocity V takes the form:

$$|V| = \begin{cases} V_{oi}, & t \le t_{oi} \\ V_{oi} + \dfrac{|V_{mi} - V_{oi}|}{2}\left(1 - \cos\left(\dfrac{\pi(t - t_{oi})}{d_{ni} - t_{oi}}\right)\right), t_{oi} < t \le d_{ni}, & V_{mi} \ge V_{oi} \\ V_{oi} + \dfrac{|V_{mi} - V_{oi}|}{2}\left(\cos\left(\dfrac{\pi(t - t_{oi})}{d_{ni} - t_{oi}}\right) - 1\right), t_{oi} < t \le d_{ni}, & V_{mi} < V_{oi} \\ V_{mi}, & t \le t_m \end{cases}$$

where,

t_m; represents the maximum flight time.

n; represents a discrete random variable to determine the number of wind steps for t_m.

V_{oi}; represents the wind velocity before each step.

t_{oi}; represents a discrete random variable to determine each wind step start.

d_{ni}; represents a discrete random variable to determine each duration of gust.

V_m; represents each gust magnitude.

Simulation example of the model given in Li and Wang (2013) for $t_0 = [0; 9; 16; 19]s$, $V_m = [1; 4.5; 0; 1]m/s$, $d_n = [7; 5; 2; 5]s$, $t_m = 25s$ and $V_0 = 0.5m/s$ is shown in Fig. 3. However the following limitations apply when generating random values:

$$n \in \left[0, \frac{t_m}{10}, \right];\ d_i \in [0, t_{i+1} - t_i]; v_i \in [0, V_{max}];$$
$$\frac{|v_i - v_{i-1}|}{d_i} < a \text{ (a: restriction of the rate of step rise)}$$

v_i; represents a discrete random variable to determine each gust magnitude.

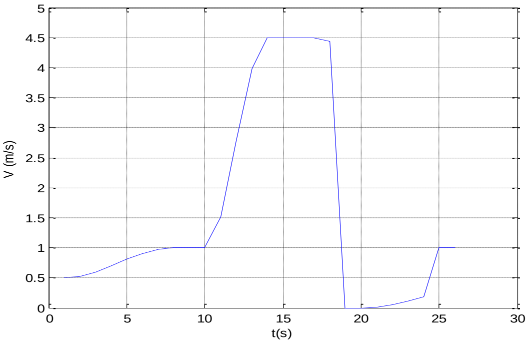

Fig. 3: Simulation showing wind velocity before and after wind gust

Also, the point at which wind blows as a wind direction be the azimuth (Ψ_w), measured from the north through east. Wind direction changes at each wind velocity step is given by:

$$\Psi_{w(i+1)} = \Psi_{w(i)} + \Delta\Psi_{w(i)}$$

where $\Delta\Psi_w$ is the random value of wind direction change. Since the wind velocity changes with altitude, the average wind velocity is determined by:

$$V_{cz} = V_{oz}(z/z_o)^p$$

where,
V_{cz}; wind velocity at the altitude of z.
V_{0z}; specified wind velocity at the altitude of z_o.
p; energetic wind profile index.
The wind force is given by:

$$F_w = S_e A V_{cz}^2$$

where,
S_e is the effective area influenced by the wind and A is a conversion factor to Nm².

With reference to the influence force, we decompose into the following components for more appropriate or easier application:

$$F_{wx} = S_e A V_{cz}^2 \cos(\Psi_w); \quad F_{w_y} = S_e A V_{cz}^2 \sin(\Psi_w)$$

For simplicity, the quadrotor surface area is represented as a cylinder. So the surface area:

$$S_k = \mu 2\pi rh + \sigma^2 \pi r^2$$

The right hand representing the sum of lateral area and bases and μ, σ representing the fill factors here. Therefore, if wind affects only half of the quadrotor the effective area will be given by:

$$S_{e_x} = \mu\pi rh\cos(\theta) + \sigma\pi r^2 \sin(\theta)$$
$$S_{e_y} = \mu\pi rh\cos(\varphi) + \sigma\pi r^2 \sin(\varphi)$$

with θ and φ representing the pitch and yaw angles.

4. Controller design

4.1. Problem formulation

A quick preview of the modeling of this tiltable UAV reveals that the system presented has more inputs than degrees of freedom; that is to say, it is overactuated. Taking that into account, it might be desirable to find six control inputs -forces and moments- that reflect a direct effect on the six degrees of freedom. The choice that's most related to the degrees of freedom is the set of forces and moments along the three axes. Let $F_v = [F_x \ \ F_y \ \ F_z]^T$ and $M_v = [M_x \ \ M_y \ \ M_z]^T$ be the control inputs which are related to the actual 12 inputs through the following static equations:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \sum_{i=1}^{4} F_i \tag{11}$$

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \sum_{i=1}^{4} M_i \tag{12}$$

Note that the 6 control inputs $[F_x \ F_y \ F_z \ M_x \ M_y \ M_z]$ are composed of nonlinear combination of 12 actual inputs $[\omega_i \ \alpha_i \ \beta_i], i = 1,...,4$. This means that for each set of control inputs, the system has many combinations of actual inputs. The way to find the best combination is discussed in the next section.

For convenience, the equations of forces and moments (6) and (7) will be treated as two subsystems in terms of the new control inputs for the rest of our analysis where Eq. 13 represents subsystem

Σ_1 and Eq. 14 represents subsystem Σ_2 where

$$\Sigma_1:\ m\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} - \begin{bmatrix} K_1\dot{x} \\ K_2\dot{y} \\ K_3\dot{z} \end{bmatrix} + R_B^E \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \tag{13}$$

$$\Sigma_2:\ I\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = -\left(\begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I\begin{bmatrix} p \\ q \\ r \end{bmatrix}\right) - M_G + \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \tag{14}$$

To construct the formulation of feedback linearization for tracking problem for Σ_1, let e_η be the error in the position

$$e_\eta = \eta_d - \eta$$

where η is the position vector, η_d is the desired position vector. Dividing by m and subtracting η_d from both sides of Eq. 13

$$\ddot{\eta} - \ddot{\eta}_d = g_z - (K/m)\dot{\eta} + \left(R_B^E/m\right)F - \ddot{\eta}_d$$

then

$$\ddot{e}_\eta = \{-g_z + (K/m)\dot{\eta} - \left(R_B^E/m\right)F + \ddot{\eta}_d\} \qquad (15)$$

The control input for this subsystem is F. To introduce feedback linearization for this part of the system through the control input, denoted by F_v, let

$$F_v = m\left(R_B^E\right)^{-1}\{-g_z + (K/m)\dot{\eta} + \ddot{\eta}_d + K_{vF}\dot{e}_\eta + K_{pF}e_\eta\} \; (16)$$

A similar procedure is carried out for Σ_2. Let

$$M_v = (\Omega \times I\Omega) - I\dot{\Omega} + M_G + IK_{vM}e_\Omega + IK_{pM}\int e_\Omega dt \qquad (17)$$

where the constants K_{vF}, K_{pF}, K_{vM} and K_{pM} are to be determined. The closed loop configuration is shown in Fig. 4.

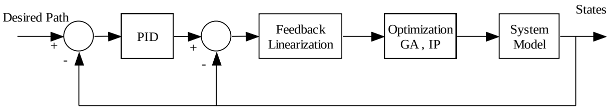

Fig. 4: The control design architecture

4.2. Feedback linearization and stability analysis

In this subsection, the stability of the system is studied and stated by the following two theorems.

Theorem 1: Subsystems Σ_1 and Σ_2 are asymptotically stable under control inputs (16) and (17) respectively if the following two conditions are satisfied

- $Kp_F = Kp_M = I$
- K_{vF} and K_{vM} are positive definite matrices

Proof: For Σ_1, choose the Lyapunov function

$$V(e_\eta) = 1/2\left(e_\eta^T e_\eta + \dot{e}_\eta^T \dot{e}_\eta\right) \geq 0 \qquad (18)$$

then

$$\begin{aligned}\dot{V}(e_\eta) &= e_\eta^T\dot{e}_\eta + \dot{e}_\eta^T\ddot{e}_\eta = \dot{e}_\eta^T e_\eta + \dot{e}_\eta^T\ddot{e}_\eta = \dot{e}_\eta^T\left(e_\eta + \ddot{e}_\eta\right) \\ &= \dot{e}_\eta^T\{e_\eta - g_z + (K/m)\dot{\eta} - \left(R_B^E/m\right)F + \ddot{\eta}_d\}\end{aligned} \qquad (19)$$

A necessary and sufficient condition for the error e_η to be locally stable is that $\dot{V}(e_\eta) \leq 0$. To check this condition, the linearization controller is introduced through the input F_v as follows

$$F_v = m\left(R_B^E\right)^{-1}\{-g_z + (K/m)\dot{\eta} + \ddot{\eta}_d + K_{vF}\dot{e}_\eta + e_\eta\} \qquad (20)$$

Substituting F_v into F in Eq. 19 yields

$$\dot{V}(e_\eta) = -K_{vF}\dot{e}_\eta^T\dot{e}_\eta \leq 0 \qquad (21)$$

Comparing the expression of F_v in Eq. 20 with F_v in Eq. 16 yields $K_{pF} = I$ while K_{vF} is required to be a positive definite matrix.

For Σ_2, choose the Lyapunov function

$$V(\Omega) = \tfrac{1}{2}\left(\int e_\Omega^T \int e_\Omega + e_\Omega^T e_\Omega\right) \geq 0 \qquad (22)$$

Then

$$\begin{aligned}\dot{V}(e_\Omega) &= e_\Omega^T \int e_\Omega + e_\Omega^T\dot{e}_\Omega = e_\Omega^T\left(\int e_\Omega + \dot{e}_\Omega\right) \\ &= e_\Omega^T\left(\int e_\Omega + (I^{-1}\Omega \times \Omega) + I^{-1}M_G - I^{-1}M + \dot{\Omega}_d\right)\end{aligned} \qquad (23)$$

Similarly, to check the stability condition, the feedback linearization controller is introduced through the input

$$M_v = (\Omega \times I\Omega) + I\dot{\Omega}_d + M_G + IK_{vM}e_\Omega + I\int e_\Omega \qquad (24)$$

Substituting M_v into M in Eq. 23

$$\dot{V}(e_\Omega) = -K_{vM}e_\Omega^T e_\Omega \leq 0 \qquad (25)$$

Similarly again, comparing the expression of M_v in Eq. 24 with M_v in Eq. 17 yields $K_{pM} = I$ while K_{vM} is required to be a positive definite matrix.

Theorem 2: Assume the drag constant K is not known exactly and is estimated with $\widehat{K}$, subsystem Σ_1 is asymptotically stable if the following conditions are satisfied

- $Kp_F = Kp_M = I$
- K_{vF} and K_{vM} are positive definite matrices
- Drag constant K is replaced with $\widehat{K}$, where
- $\dot{\widehat{K}} = \operatorname{diag}\left\{\frac{\dot{e}_{(1)}\dot{\eta}_{(1)}}{m} \quad \frac{\dot{e}_{(2)}\dot{\eta}_{(2)}}{m} \quad \frac{\dot{e}_{(3)}\dot{\eta}_{(3)}}{m}\right\}$

Proof: The drag constant appears only in subsystem 1. Choose the Lyapunov function

$$V(e_\eta, \widetilde{K}) = 1/2\left(e_\eta^T e_\eta + \dot{e}_\eta^T\dot{e}_\eta\right) + 1/2 s^T\widetilde{K}^T\widetilde{K}s \geq 0 \qquad (26)$$

where

$s = [1 \quad 1 \quad 1]^T$

and

$\widetilde{K} = \widehat{K} - K.$

then

$$\begin{aligned}\dot{V}(e_\eta, \widetilde{K}) &= e_\eta^T\dot{e}_\eta + \dot{e}_\eta^T\ddot{e}_\eta + s^T\widetilde{K}^T\dot{\widetilde{K}}s \\ 2s^T\widetilde{K}^T\widetilde{K}s \quad &= \dot{e}_\eta^T e_\eta + \dot{e}_\eta^T\ddot{e}_\eta + s^T\widetilde{K}^T\dot{\widetilde{K}}s = \dot{e}_\eta^T\left(e_\eta + \ddot{e}_\eta\right) + s^T\widetilde{K}^T\dot{\widetilde{K}}s \\ &= \dot{e}_\eta^T\left[e_\eta - g_z + (K/m)\dot{\eta} - (R_B^E/m)F + \ddot{\eta}_d\right] + s^T\widetilde{K}^T\dot{\widetilde{K}}s\end{aligned} \qquad (27)$$

But $\dot{\widetilde{K}} = \dot{\widehat{K}} - \dot{K}^{\cdot}$ and since K is constant, this yields $\dot{\widetilde{K}} = \dot{\widehat{K}}^{\hat{}\cdot\cdot}$ Then $\dot{V}(e_\eta, \widetilde{K})$ becomes

$$\begin{aligned}\dot{V}(e_\eta, \widetilde{K}) = \; & \dot{e}_\eta^T\left[e_\eta - g_z + (K/m)\dot{\eta} - (R_B^E/m)F + \ddot{\eta}_d\right] \\ & + s^T\widetilde{K}^T\dot{\widehat{K}}s\end{aligned} \qquad (28)$$

Introducing the feedback linearization controller through the input

$$F = m\left(R_B{}^E\right)^{-1}\left(-g_z + \left(\widehat{K}/m\right)\eta^{\cdot} + \ddot{\eta}_d + K_{\nu F}e^{\cdot}{}_\eta + e_\eta\right) \quad (29)$$

Substituting in $\dot{V}(e_\eta, \widetilde{K})$ and after some manipulations

$$\dot{V}(e_\eta, \widetilde{K}) = -K_{\upsilon F}(\dot{e}_\eta^T \dot{e}_\eta) - \dot{e}_\eta^T(\widetilde{K}/m)\dot{\eta} + s^T \widetilde{K}^T \dot{\hat{K}} s \tag{30}$$

Note that $\widetilde{K}$ and $\dot{\hat{K}}$ are 3x3 diagonal matrices while $e^{\cdot}$ and $\eta^{\cdot}$ are 3x1 matrices.

To analyze the second and the third terms of Eq. 30 further, the matrices are broken down to their basic elements

$$\begin{aligned}
&-\dot{e}_\eta^T \frac{\widetilde{K}}{m}\dot{\eta} + s^T \widetilde{K}^T \dot{\hat{K}} = \\
&-\frac{1}{m}\left[\dot{e}_{(1)}\widetilde{K}_{(1,1)}\dot{\eta}_{(1)} + \dot{e}_{(2)}\widetilde{K}_{(2,2)}\dot{\eta}_{(2)} + \dot{e}_{(3)}\widetilde{K}_{(3,3)}\dot{\eta}_{(3)}\right] \\
&+\left[\widetilde{K}_{(1,1)}\dot{\hat{K}}_{(1,1)} + \widetilde{K}_{(2,2)}\dot{\hat{K}}_{(2,2)} + \widetilde{K}_{(3,3)}\dot{\hat{K}}_{(3,3)}\right] \\
&= \widetilde{K}_{(1,1)}\left[\dot{\hat{K}}_{(1,1)} - \frac{\dot{e}_{(1)}\dot{\eta}_{(1)}}{m}\right] + \widetilde{K}_{(2,2)}\left[\dot{\hat{K}}_{(2,2)} - \frac{\dot{e}_{(2)}\dot{\eta}_{(2)}}{m}\right] \\
&+\widetilde{K}_{(3,3)}\left[\dot{\hat{K}}_{(3,3)} - \frac{\dot{e}_{(3)}\dot{\eta}_{(3)}}{m}\right]
\end{aligned} \tag{31}$$

since $\dot{\hat{K}}$ is defined as $\dot{\hat{K}} = \operatorname{diag}\left\{\frac{\dot{e}_{(1)}\dot{\eta}_{(1)}}{m} \quad \frac{\dot{e}_{(2)}\dot{\eta}_{(2)}}{m} \quad \frac{\dot{e}_{(3)}\dot{\eta}_{(3)}}{m}\right\}$. Then

$$\dot{V} = -K_{\upsilon F}(\dot{e}_\eta^T \dot{e}_\eta) \leq 0 \tag{32}$$

5. Optimization

We mean by Optimization here is the process of selecting the best (optimum) element -or set of elements- from a set of available alternatives according to some preset criteria. In the situation presented here, it is a question of finding the best -according to some cost function- combination of inputs that satisfy the control equations.

As mentioned in the previous section, the forces F and moments M are used as inputs to the system to avoid dealing with the nonlinearity present in the actual inputs ω_i, α_i and β_i Feedback linearization control is also achieved through F and M which means that the controller chooses values of F and M to be fed as inputs to the system. It has been also shown in Elfeky et al. (2013) that with only two opposite rotors running, there's always a combination of ω_i, α_i and β_i that satisfy any arbitrary values of F and M. This means that with full actuation, there are many combinations of ω_i, α_i and β_i that satisfy any arbitrary values of F and M. The question addressed in this section is how to choose one combination of ω_i, α_i and β_i from the set of many combinations.

Several optimization techniques are available in the literature. However, few of them deal with nonlinear problem with constraints. Two optimization techniques are tested and compared as a proof of concept and to complete the controller design discussed in the previous section. Those techniques are Genetic algorithm and Interior-point algorithm. This part specifically has high potential for improvement in both theory and implementation. It must be clear that this is not a typical optimal control problem where the objective is to find optimal gains for the controller, instead, the controller gains are already determined and the purpose of optimization is to map the forces and moments used in the controller to the actual 12 inputs of the system through the static equations relating them.

Interior point (IP) is an optimization class of algorithms that can solve linear and nonlinear convex optimization problems. The algorithm reaches an optimal solution by traversing the interior of the feasible region rather than the surface of the region. Convex optimization and particularly interior point are covered in Nesterov and Nemirovskii (1994) and Renegar (2001).

Genetic Algorithm (GA) belongs to a class of algorithms called Evolutionary Algorithms (EA). EA generates solutions using nature-inspired techniques such as inheritance, mutation, selection, and crossover. GA is covered in details in Goldberg (1989), Homaifar et al. (1994), Fonseca and Fleming (1995), and Michalewicz et al. (1992).

5.1. The optimization problem formulation

It's first desired to find a cost function for the optimization problem, which means finding some criteria to choose among the many sets of inputs. The most typical and critical objective in UAV application is the minimization of energy consumption during the flight. The cost function can be written as follow:

$$J(\omega, \alpha, \beta) = \sum_{i=1}^{n}\{\omega_1 \sum_{i=1}^{4} \omega_i^3 + \omega_2 \sum_{i=1}^{4}(\Delta\alpha_i)^2 + \omega_2 \sum_{i=1}^{4}(\Delta\beta_i)^2\} \tag{33}$$

where the first term is the power consumption by the rotors. The cubic exponent comes from the assumption that the torque is proportional to the square of the angular velocity, while the power is the torque time the angular velocity. The second and the third terms penalize tilting movements (cost of energy) and n is the number of samples over which the optimization is performed.

The optimization problem can then be written as follows

$$\min \quad J(\omega, \alpha, \beta)$$

Subject to

$$\begin{gathered}
\begin{bmatrix} F_{x(j)} \\ F_{y(j)} \\ F_{z(j)} \end{bmatrix} = \sum_{i=1}^{4} \begin{bmatrix} 0 & 0 & c\beta_{(i,j)}s\alpha_{(i,j)} \\ 0 & 0 & s\beta_{(i,j)}s\alpha_{(i,j)} \\ 0 & 0 & c\alpha_{(i,j)} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ b\omega_{(i,j)}^2 \end{bmatrix}, \ j = 1, \dots, 4 \\
\begin{bmatrix} M_{x(j)} \\ M_{y(j)} \\ M_{z(j)} \end{bmatrix} = \sum_{i=1}^{4} \begin{bmatrix} 0 & 0 & c\beta_{(i,j)}s\alpha_{(i,j)} \\ 0 & 0 & s\beta_{(i,j)}s\alpha_{(i,j)} \\ 0 & 0 & c\alpha_{(i,j)} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ d\omega_{(i,j)}^2 \delta_{(i)} \end{bmatrix} + r_i \times F_i \\
0 \leq \omega_i \leq 10000 \\
|\beta_i| \leq \pi/2 \\
|\alpha_i| \leq \pi/6
\end{gathered} \tag{34}$$

Two algorithms are tested to perform the optimization problem, Genetic Algorithm and interior-point algorithm. These are implemented using MATLAB functions ga and fmincon. Two algorithms were developed for each method, an offline and recursive algorithms. The objective of the recursive algorithm is to minimize the time for optimization to converge by optimizing for one sample with 12 inputs only at a time instead of optimizing for the 12 inputs in the whole flight samples. In Genetic Algorithm, the recursive technique is developed by setting the final population in one time sample to be the initial population in the next time sample. While in interior point optimization, the recursive algorithm is developed by setting the final solution at one time sample to be the initial set point in the next sample. Those techniques work only if the sampling time is small enough so that the there's no big difference between two consecutive samples.

6. Simulation results

In this section, the study results will be reported. First the simulation, while considering no wind gust and disturbance is presented. After that the results with wind gust is stated.

First, an initial comparison is carried out to choose which optimization algorithm would be more suitable. The criteria for comparison are convergence time and optimality in terms of cost function. However, the first priority here is time for practical implementation purposes. The four algorithms are tested for 1, 2 and 3 samples of a flight simulation. The results are listed in Table 1 which includes the convergence time and the optimal value of the objective function.

Table 1: Convergence time and objective function comparison

Samples		GA	RGA	IP	RIP
1	conv. time	898.0 s	898.0 s	0.6 s	0.6 s
	J	1.20	1.20	0.27	0.27
2	conv. time	12947.9 s	5357.4 s	2.4 s	1.3 s
	J	59.80	1.55	0.55	2.43
3	conv. time	38979.0 s	10497.4 s	13.9 s	2.0 s
	J	44.94	2.16	0.82	2.70

It's evident from the results that Recursive Interior-Point (RIP) technique is more practical for our purpose. Although conventional IP technique is superior in optimality, which is expected, RIP takes

Significantly, less time to converge which makes it the best candidate for online implementation? Further study is carried out on the developed RIP to investigate its behavior in terms of convergence time. Fig. 5 shows the convergence time against number of samples optimized. The curve is almost linear except near the first few samples as shown in Fig. 6. The slope of the linear part is equal to 0.270 m/sample. This means that the algorithm takes around 270 ms to optimize one sample of 12 inputs. This result is not suitable for online implementation, therefore all flight simulations are implemented offline. However, this result is very promising for future research.

To test the control algorithm, the quadrotor is commanded to elevate up to 10m, tilt forward with 30 degrees and perform a circle while pointing to its center. To perform a circle, the quadrotor is commanded to follow sinusoidal paths along x and y axes. The flight is an example of a surveillance mission where the quadrotor might be taking panoramic photos of a certain target. This complex flight demonstrates the strength of both the design and control technique.

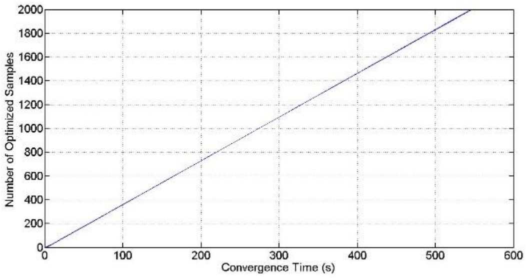

Fig. 5: Convergence time for recursive interior-point algorithm

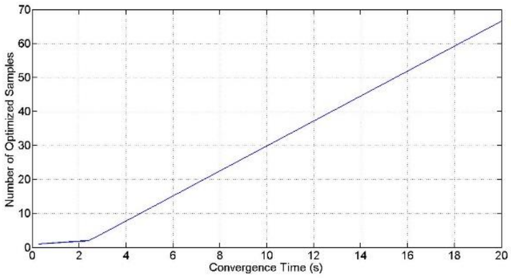

Fig. 6: Convergence time for recursive interior-point algorithm for small time

The optimization is done offline, i.e., and prior to the flight. The parameters of the system are assumed to be known. The control parameters are chosen, according to Theorem 1. $K_{pF} = K_{pM} = I$ while $K_{vF} = K_{vM}$ are taken to be equal to 3I, where I here is the 3x3 identity matrix. Fig. 7 shows both the desired and actual positions of the quadrotor in x, y and z axes while Fig. 8 shows the first derivative of the position vector and the 3-D position is showed in Fig. 9. It's evident from the figures that the path was followed with high accuracy.

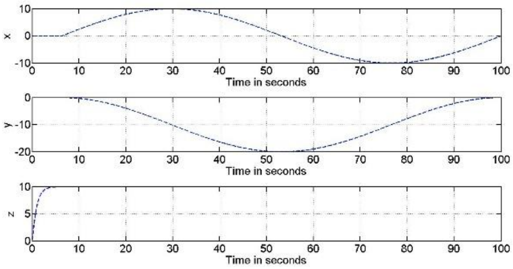

Fig. 7: x, y and z positions of the quadrotor

The orientation vector of the quadrotor is shown in Fig. 10 in radians. It can be seen that the angle reached π/6 and the angle went from 0 to 2π to keep pointing towards the center of the circle while the quadrotor is completing a full circle. This means that Yaw angle is also performing a complete circle simultaneously.

In Figs. 11 and 12, the input angles α_i and $\beta_i$generated by optimization are shown. One objective of optimization was to limit the change in each of the angles in order to minimize energy consumption. The figures show minimal variations in the angles. Figs. 13 and 14 show the resulting forces and moments.

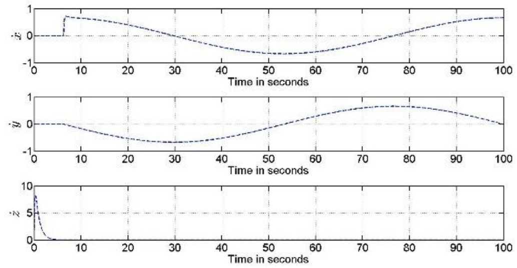

Fig. 8: Translational velocities $\dot{x}$, $\dot{y}$ and $\dot{z}$

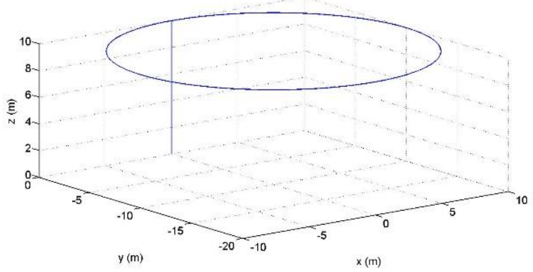

Fig. 9: A 3D plot of the flight on x,y and z axes

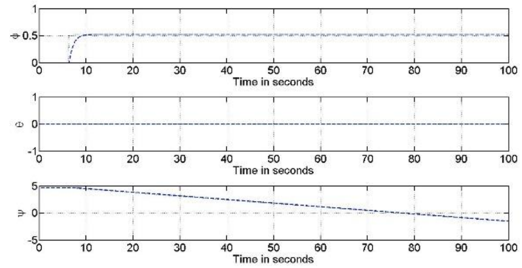

Fig. 10: Quadrotor orientations: Φ, Θ, and Ψ

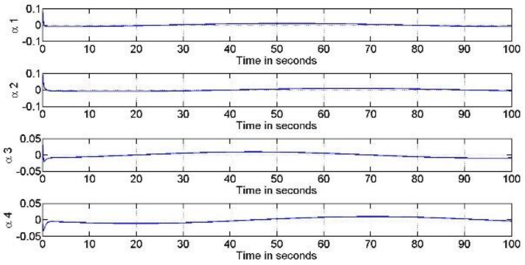

Fig. 11: Generated α_i by optimization

Finally, the simulation is done with wind gust and disturbances. Fig. 15 shows the elevation position of the command to take off vertically and then thrust further in x direction to a height of [10, 0, 50] m. In Fig. 16, the 3D plots are shown. Here, the controller is able to track the coordinate under wind disturbance with a negligible error due to wind. Fig. 17 which depicts the coordinates of the quadrotor moving in the x direction with a slight error in the y position as shown due to wind gust disturbance. Fig. 18 shows some fluctuation in the velocities due to effect of wind disturbance. Fig. 19 and Fig. 20 shows the angles and the angular velocities are unaffected under wind disturbance. This shows the capability of the quadrotor itself to move towards a certain direction without compromising the attitude under wind disturbance.

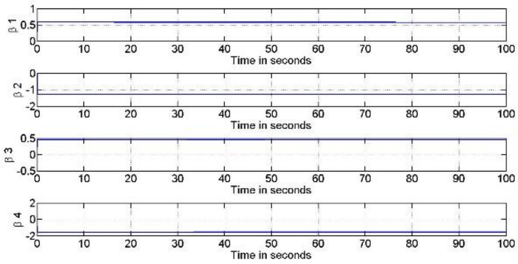

Fig. 12: β_iGenerated by optimization

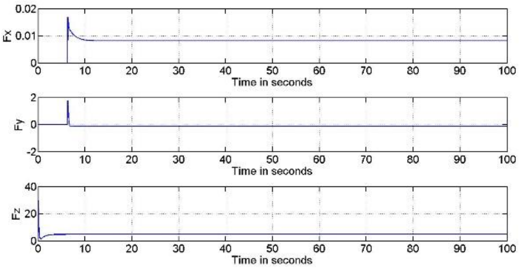

Fig. 13: Desired forces generated by the controller

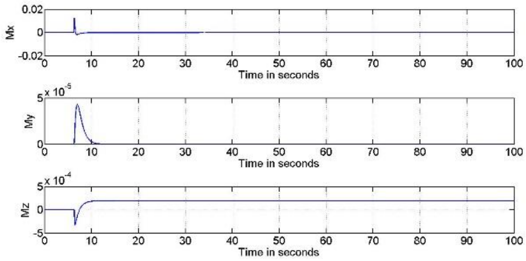

Fig. 14: Desired moments generated by the controller

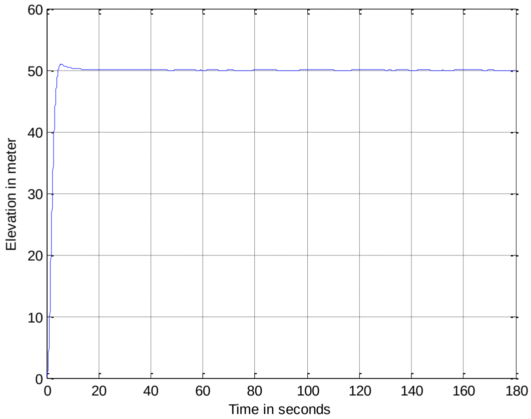

Fig. 15: Position of elevation and thrust further to a [10, 0, 50] m location under wind gust

7. Conclusion

Feedback linearization proved to stabilize and control the attitude of the quadrotor with tilting rotors. All the degrees of freedom can be controlled freely and independently. This allows for more complex and precise maneuvers that weren't possible with conventional quadrotor. In summary, this control algorithm highlights the advantages of the novel design by taking advantage of the high mechanical flexibility available. In fact, the controller decouples the model of the quadrotor into two completely separate systems, systems of forces and positions, and a system of moments and orientations. The two systems can be studied and controlled absolutely independently. The only connection between the two systems appears in the final stage of optimization where the six equations are solved together to find the 12 inputs. The optimization part is developed to choose the best inputs according to energy consumption which is very critical in all UAV applications.

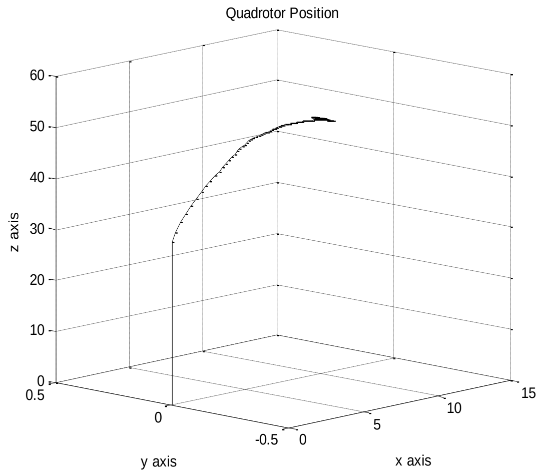

Fig. 16: 3D plot of Quadrotor position for elevation and thrust further to a [10, 0, 50] m location under wind gust

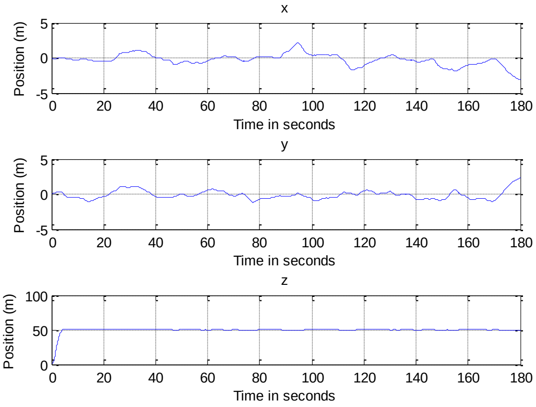

Fig. 17: x, y and z position for elevation and thrust further to a [10, 0, 50] m location under wind gust

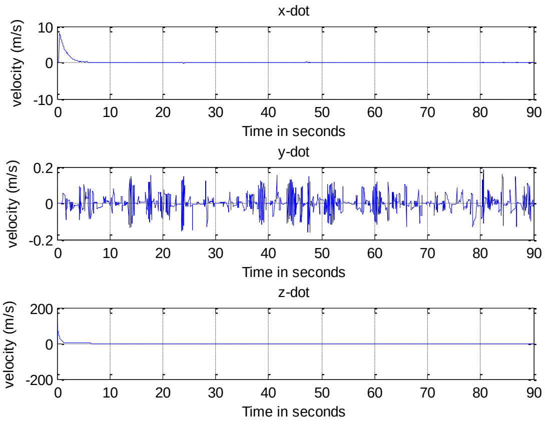

Fig. 18: Velocities for elevation and thrust further to a [10, 0, 50] m location under wind gust

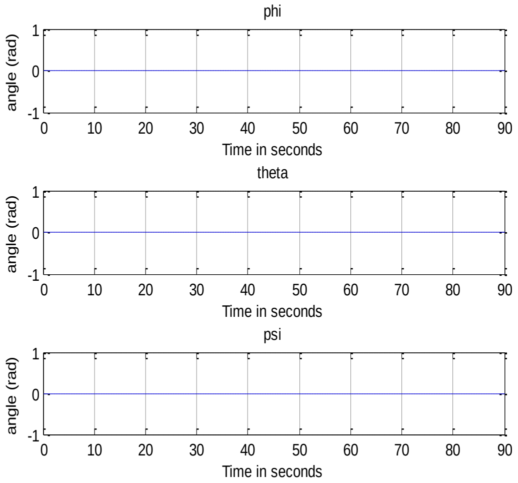

Fig. 19: Orientation angles for elevation and thrust further to a [10, 0, 50] m location under wind gust

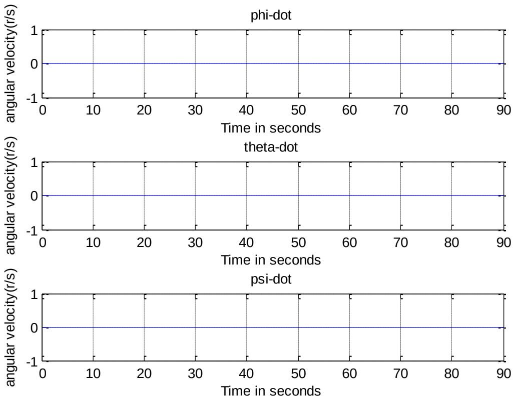

Fig. 20: Angular velocities for elevation and thrust further to a [10, 0, 50] m location under wind gust

A nested PD control loop is adopted to control the quadrotor position and cancel disturbances with additional inputs due to the quadrotor structure. The wind model is derived from an approach which takes into account the wind velocity change, the wind gust step change, the variation of wind velocity with increase in height and the changes in wind direction.

Finally, we can say that the control and optimization part show very promising results. The results demonstrate that the quadrotor with tilted rotor can effectively attain the desired trajectory in the presence of wind disturbance. However, there is still a substantial space for improvements. Future research may include a wide variety of angles and considering different scenarios of uncertainty in the parameters, states and disturbance. The optimization part as well can be further improved to consider different techniques and online implementation of optimization.

Other control techniques can also be developed for this system. Particularly, model predictive control MPC is a candidate due to the fact that it combines control and optimization under constraints.

Acknowledgment

The author would like to thank the deanship for scientific research, DSR at KFUPM for their support through project numbers IN100036 and IN41048.

References

Altug E, Ostrowski JP, and Taylor CJ (2005). Control of a quadrotor helicopter using dual camera visual feedback. The International Journal of Robotics Research, 24(5): 329-341.

Bouabdallah S, Becker M, and Siegwart R (2007). Autonomous miniature flying robots: coming soon!-research, development, and results. IEEE Robotics Automation Magazine, 14(3): 88-98.

Elfeky M, Elshafei M, Saif AWA, and Al-Malki MF (2013). Quadrotor helicopter with tilting rotors: Modeling and simulation. In the World Congress on Computer and Information Technology, IEEE, Sousse, Tunisia: 1-5. https://doi.org/10.1109/WCCIT.2013.6618768

Elfeky M, Elshafei M, Saif AWA, and Al-Malki MF (2014). Quadrotor with tiltable rotors for manned applications. In the 11th International Multi-Conference on Systems, Signals and Devices, IEEE, Barcelona, Spain: 1-5. https://doi.org/10.1109/SSD.2014.6808791

Fonseca CM and Fleming PJ (1995). An overview of evolutionary algorithms in multi-objective optimization. Evolutionary Computation, 3(1): 1–16.

Goldberg DE (1989). Genetic algorithms in search, optimization and machine learning. Addison-Wesley, Boston, Massachusetts, USA.

Habib M, Quimby PW, Chang S, Jackson K, and Cummings ML (2011). Wind gust alerting for supervisory control of a micro aerial vehicle. In the IEEE Aerospace Conference, IEEE, Big Sky, USA: 1-7. https://doi.org/10.1109/AERO.2011.5747536

Homaifar A, Qi CX, and Lai SH (1994). Constrained optimization via genetic algorithms. Simulation, 62(4): 242–253.

Hua MD, Hamel T, and Samson C (2013). Control of VTOL vehicles with thrust-direction tilting. Available online at: https://arxiv.org/pdf/1308.0191.pdf

Li Y and Wang G (2013). Quad-Rotor airship modeling and simulation based on backstepping control. International Journal of Control and Automation, 6(5): 369-384.

Liu H, Derawi D, Kim J, and Zhong Y (2013). Robust optimal attitude control of hexarotor robotic vehicles. Nonlinear Dynamics, 74(4): 1155-1168.

Michalewicz Z, Janikow CZ, and Krawczyk JB (1992). A modified genetic algorithm for optimal control problems. Computers Mathematics with Applications, 23(12): 83–94.

Morel Y and Leonessa A (2006). Direct adaptive tracking control of quadrotor aerial vehicles. In the ASME International Mechanical Engineering Congress and Exposition, Dynamic Systems and Control Division, Chicago, Illinois, USA: 155-161. https://doi.org/10.1115/IMECE2006-15516

Nesterov Y and Nemirovskii A (1994). Interior-point polynomial algorithms in convex programming. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, USA.

Oner KT, Cetinsoy E, Unel M, and Aksit MF (2008). Dynamic model and control of a new quadrotor unmanned aerial vehicle with tiltwing mechanism. World Academy of Science, Engineering and Technology, 45: 58-63.

Pounds P, Mahony R, and Corke P (2010). Modelling and control of a large quadrotor robot. Control Engineering Practice, 18(7): 691-699.

Renegar J (2001). A mathematical view of interior-point methods in convex optimization. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, USA.

Ryll M, Bülthoff HH, and Giordano PR (2012). Modeling and control of a quadrotor UAV with tilting propellers. In the IEEE International Conference on Robotics and Automation, IEEE, Saint Paul, MN, USA: 4606-4613. https://doi.org/10.1109/ICRA.2012.6225129

Saif AA, Dhaifullah M, Al-Malki M, and Shafie ME (2012). Modified backstepping control of quadrotor. In the 9th International Multi-Conference on Systems, Signals and Devices, IEEE, Chemnitz, Germany: 1-6. https://doi.org/10.1109/SSD.2012.6197975

Salih AL, Moghavvemi M, Mohamed HA, and Gaeid KS (2010). Modelling and PID controller design for a quadrotor unmanned air vehicle. In the IEEE International Conference on Automation Quality and Testing Robotics, IEEE, Cluj-Napoca, Romania, 1: 1-5. https://doi.org/10.1109/AQTR.2010.5520914

Senkul F and Altug E (2014). Adaptive control of a tilt-roll rotor quadrotor UAV. In the International Conference on Unmanned Aircraft Systems, IEEE, Orlando, USA: 1132-1137. https://doi.org/10.1109/ICUAS.2014.6842367

Solovyev VV, Finaev VI, Zargaryan YA, Shapovalov IO, and Beloglazov DA (2015). Simulation of wind effect on a quadrotor flight. ARPN Journal of Engineering and Applied Science, 10(4): 1535-1538.

Voos H (2009). Nonlinear control of a quadrotor micro-UAV using feedback-linearization. In the IEEE International Conference on Mechatronics, IEEE, Malaga, Spain: 1-6. https://doi.org/10.1109/ICMECH.2009.4957154