ISSN 2176-5480

22nd International Congress of Mechanical Engineering (COBEM 2013)
November 3-7, 2013, Ribeirão Preto, SP, Brazil
Copyright © 2013 by ABCM

MULTIROTOR POSITION TRACKING USING A LINEAR CONSTRAINED MODEL PREDICTIVE CONTROLLER

Igor Afonso Acampora Prado
Davi Antônio dos Santos
Praça Marechal Eduardo Gomes, 50, Vila das Acácias, 12228-900, São José dos Campos, SP, Brazil
igorap@ita.br, davists@ita.br

Abstract. *The interest for multirotor unmanned aerial vehicles (UAVs) is currently growing due to their high maneuverability, simplified mechanics, capability to perform vertical take-off and landing as well as hovering flight. These characteristics make them a promising technology suitable for applications such as surveillance of indoor environments. This paper faces the problem of safely controlling the trajectory of a multirotor UAV by taking into consideration a conic constraint on the total thrust vector. This problem is solved using a linear state-space model predictive controller (MPC) formulation, whose optimization is made handy by replacing the original conic constraint set by a circumscribed pyramidal space that renders a linear set of inequalities on the total thrust vector. The control variables computed by the MPC is converted to the corresponding principal Euler angle/axis, which in turn is transformed to the modified Rodrigues parameters (MRP) representing the set-point for some inner attitude control loop. The proposed method is evaluated on the basis of computational simulations, which shows its effectiveness in tracking both square and spiral trajectories while respecting the conic bounds imposed on the total thrust vector.*

Keywords: *Trajectory control, unmanned aerial vehicle, model predictive control, multirotor, aerial robotics.*

1. INTRODUCTION

The multirotor-type unmanned aerial vehicle (UAV) technology has attracted a great deal of interest of the academia and industry and, consequently experienced a rapid improvement. This interest is justified by features such as their simplified mechanics, low cost, high maneuverability, and vertical take-off and landing (VTOL) capability. Multirotor UAVs has the potential to be employed in applications too risky to human beings or simply intended to increase the efficiency of certain tasks. Examples of applications are building exploration (Hoffman *et al.*, 2008), traffic monitoring, mapping of agricultural areas, search and rescue (Alexis *et al.*, 2011).

The multirotors have six degrees of freedom (DOF): three of translational motion and three others corresponding to the rotation. However, they feature only four independent control variables: three torque components and the magnitude of the total thrust vector. Therefore, the first difficulty in designing a control system for them could seem to be the need for facing this underactuation characteristic. However, it is necessary to consider that in most of the applications with a market potential, one is in fact concerned to control only four DOFs: the three-dimensional position and the heading angle. The other DOFs must only to be stabilized. A popular control system structure that obviates the dynamics underactuation and allows to control the aforementioned four DOFs is illustrated in Figura 1. The main goal of this scheme is make the true vehicle position $\mathbf{r} \in \mathbb{R}^3$ track some desired position command $\mathbf{r}_\mathrm{d} \in \mathbb{R}^3$. The navigation system is responsible for providing estimated values, in engineering units, of the vehicle attitude $\mathbf{d} \in \mathbb{R}^3$, angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$, position $\mathbf{r}$ and linear velocity $\dot{\mathbf{r}} \in \mathbb{R}^3$. The system structure is divided in two control loops, where the inner loop is responsible for controlling the vehicle attitude and the outer loop carries out the trajectory control by providing references of attitude commands $\mathbf{d}_\mathrm{ref} \in \mathbb{R}^3$, seeing that the aircraft needs to perform an inclination, in relation to the rotor's plane, to accelerate in a desired direction. Moreover, the outer loop computes the total thrust magnitude $f \in \mathbb{R}$ to control the vertical acceleration. On the other hand, the attitude control block calculates the torque $\boldsymbol{\tau} \in \mathbb{R}^3$ that makes the aircraft follow the attitude commands $\mathbf{d}_\mathrm{ref}$.

Many efforts have been made by various researchers to devise autonomous control strategies to solve the position tracking problem. Mistler *et al.* (2001) used the feedback linearization method to guide the aircraft through a reference trajectory. Castillo *et al.* (2004) applied the nested saturated controller to positioning the vehicle and stabilize its attitude. In Bouabdallar and Siegwart (2005), the vehicle model was partitioned into two subsystems: one part describing the translation and the other one modeling rotation. Then, sliding mode and backstepping techniques were used to control the vehicle. Raffo *et al.* (2010) proposed a similar structure utilizing model predictive controller (MPC) to track the reference trajectory and a nonlinear H_∞ to stabilize the rotational motion. They took into account aerodynamic disturbances and parametric as well as structural uncertainties. Lopes *et al.* (2011) designed a single MPC controller to reach simultaneous position control and attitude stabilization.

This paper proposes a new control strategy to face the position tracking problem under constraints on the total thrust vector. Particularly, the thrust inclination is bounded by a maximum allowable value, while the thrust magnitude is bounded by a minimum and a maximum permitted value. The corresponding constraint set is a piece of a cone with an inferior and a superior spherical lids. In order to obtain a linear and convex constraint set, the original conic space is replaced by one circumscript pyramid. On the other hand, a linear dynamic model for the translation is obtained by feedback linearization. An state-space MPC relying on this linear model and on the pyramidal constraint set is then designed to solve the so-posed problem. The control inputs computed by the MPC are then used to compute an attitude command and a thrust magnitude command. The remainder of the paper is organized as follows. Section 2 describes the multirotor translational dynamics and defines the position tracking problem. Section 3 provides a solution to this problem. Section 4 presents computational simulations. Finally, Section 5 presents the paper's conclusions.

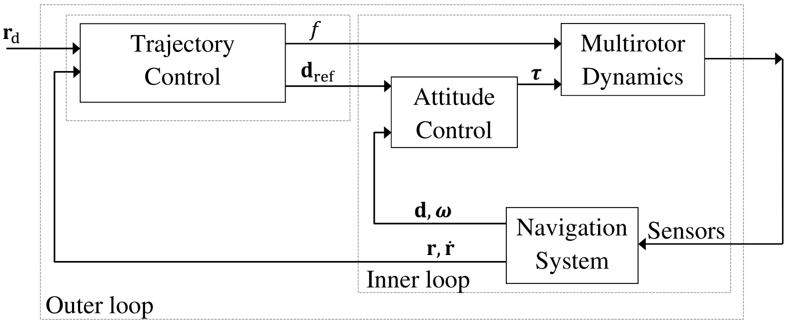

Figure 1. Block diagram structure.

2. PROBLEM STATEMENT

Consider the multirotor vehicle and the three Cartesian coordinate systems (CCS) illustrated in Figure 2. Assume that the vehicle has a rigid structure. The body CCS $S_\mathrm{B} \triangleq \{X_\mathrm{B}, Y_\mathrm{B}, Z_\mathrm{B}\}$ is fixed to the structure and its origin coincides with the center of mass (CM) of the vehicle. The reference CCS $S_\mathrm{R} \triangleq \{X_\mathrm{R}, Y_\mathrm{R}, Z_\mathrm{R}\}$ is Earth-fixed and its origin is at point O. Finally, the CCS $S_{\mathrm{R}'} \triangleq \{X_{\mathrm{R}'}, Y_{\mathrm{R}'}, Z_{\mathrm{R}'}\}$ is defined to be parallel to S_R, but its origin is shifted to CM. Assume that S_R is an inertial frame.

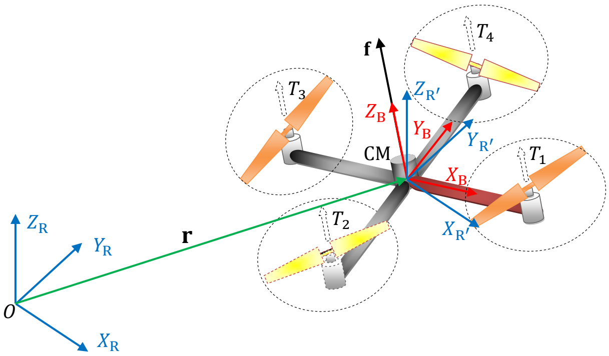

Figure 2. The Cartesian coordinate systems.

Invoking the second Newton's law and neglecting disturbance forces, the translational dynamics of the multirotor illustrated in Figure 2 can be immediately described in S_R by the following second order differential equation:

$$\ddot{\mathbf{r}} = \frac{1}{m}\mathbf{f} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}, \tag{1}$$

where $\mathbf{r} \triangleq [r_\mathrm{x}\ r_\mathrm{y}\ r_\mathrm{z}]^\mathrm{T} \in \mathbb{R}^3$ is the position vector of CM, $\mathbf{f} \triangleq [f_\mathrm{x}\ f_\mathrm{y}\ f_\mathrm{z}]^\mathrm{T} \in \mathbb{R}^3$ is the total thrust vector, m is the mass of the vehicle, and g is the gravitational acceleration. As illustrated in Figure 2, $\mathbf{f}$ is perpendicular to the rotor plane.

Define the inclination angle $\phi \in \mathbb{R}$ of the rotor plane to be the angle between Z_B and $Z_{\mathrm{R}'}$. The angle ϕ can thus be expressed by

$$\phi \triangleq \cos^{-1}\frac{f_\mathrm{z}}{f}, \tag{2}$$

where $f \triangleq \|\mathbf{f}\|$.

Define the position tracking error $\tilde{\mathbf{r}} \in \mathbb{R}^3$ as

$$\tilde{\mathbf{r}} \triangleq \mathbf{r} - \mathbf{r}_{\mathrm{d}}, \tag{3}$$

where $\mathbf{r}_{\mathrm{d}} \triangleq [r_{\mathrm{d,x}}\ r_{\mathrm{d,y}}\ r_{\mathrm{d,z}}]^{\mathrm{T}} \in \mathbb{R}^3$ is a position command.

Problem 1. Let $\phi_{\max} \in \mathbb{R}$ denote the maximum allowable value of ϕ and $f_{\min} \in \mathbb{R}$ and $f_{\max} \in \mathbb{R}$ denote, respectively, the minimum and maximum admissible values of f. The problem is to find a control law for **f** that minimizes $\tilde{\mathbf{r}}$, subject to the inclination constraint $\phi \leq \phi_{\max}$ and to the force magnitude constraint $f_{\min} \leq f \leq f_{\max}$.

3. PROBLEM SOLUTION

This section proposes a solution to Problem 1 based on a MPC strategy. Subsection 3.1 describes the system by a continuous and discrete-time linear dynamic model. Subsection 3.2 shows how to obtain the linear constraints to the control inputs. Subsection 3.3 presents an incremental-input state-space MPC formulation for computing the control forces. Finally, Subsection 3.4 shows how to convert the control forces into commands to the lower level loops.

3.1 State-Space Model

Define the state vector $\mathbf{x} \triangleq [r_x\ \dot{r}_x\ r_y\ \dot{r}_y\ r_z\ \dot{r}_z]^{\mathrm{T}} \in \mathbb{R}^6$ and the control input vector $\mathbf{u} \in \mathbb{R}^3$

$$\mathbf{u} \triangleq \frac{1}{m}\mathbf{f} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}. \tag{4}$$

Using (4), (1) can be immediately rewritten as a continuous-time linear state-space model

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \tag{5}$$

with

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{6}$$

and

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{6\times 3}. \tag{7}$$

Define the controlled output vector $\mathbf{y} \in \mathbb{R}^3$ as

$$\mathbf{y} \triangleq \mathbf{C}\mathbf{x}, \tag{8}$$

where

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 6} \tag{9}$$

Let $\mathbf{x}(k) \in \mathbb{R}^6$, $\mathbf{u}(k) \in \mathbb{R}^3$ and $\mathbf{y}(k) \in \mathbb{R}^3$ denote, respectively, the state vector, the control input vector and the controlled output vector all described in discrete-time domain. Using the Euler discretization method with a sample time of $T_s = 50$ ms, the discretized version of (5) and (8) is obtained as

$$\begin{array}{lcl} \mathbf{x}(k+1) & = & \mathbf{A}_{\mathrm{d}}\mathbf{x}(k) + \mathbf{B}_{\mathrm{d}}\mathbf{u}(k) \\ \mathbf{y}(k) & = & \mathbf{C}_{\mathrm{d}}\mathbf{x}(k) \end{array}, \tag{10}$$

where

$$\mathbf{A}_{\rm d} = \begin{bmatrix} 1 & 0.05 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.05 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0.05 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{6\times 6}, \tag{11}$$

$$\mathbf{B}_{\rm d} = \begin{bmatrix} 0.0013 & 0 & 0 \\ 0.05 & 0 & 0 \\ 0 & 0.0013 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.0013 \\ 0 & 0 & 0.05 \end{bmatrix} \in \mathbb{R}^{6\times 3}, \tag{12}$$

and $\mathbf{C}_{\rm d} \in \mathbb{R}^{3\times 6}$ remains equal to $\mathbf{C}$.

3.2 Thrust Vector Constraints

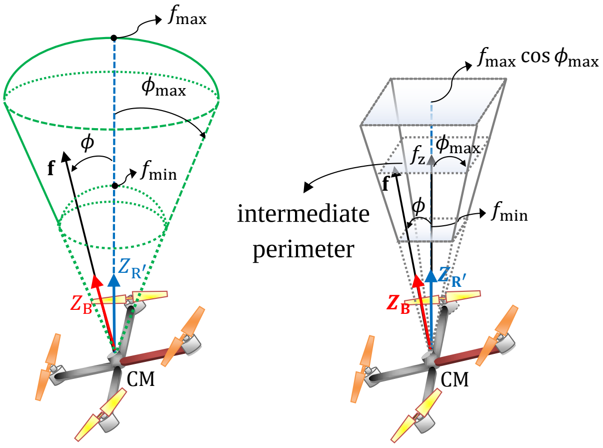

Figure 3. (a) Real constraint space; (b) Linearized constraint space.

Using (4), the thrust magnitude constraint $f_{\rm min} \leq f \leq f_{\rm max}$ can be rewritten in terms of $\mathbf{u}$ as

$$f_{\rm min} \leq \left\| m\mathbf{u} + m \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \right\| \leq f_{\rm max}, \tag{13}$$

$$f_{\rm min} \leq m\sqrt{u_{\rm x}^2 + u_{\rm y}^2 + (u_{\rm z} + g)^2} \leq f_{\rm max}. \tag{14}$$

Assuming that $0 \leq \phi_{\rm max} < \pi/2$ rad, the inclination constraint $\phi \leq \phi_{\rm max}$ can be replaced by

$$\cos\phi \geq \cos\phi_{\rm max}. \tag{15}$$

Using (2), (15) can be rewritten as

$$\frac{f_{\rm z}}{f} \geq \cos\phi_{\rm max}, \tag{16}$$

which in turn, using (4), can be rewritten in terms of $\mathbf{u}$, yielding

$$\frac{u_{\rm z} + g}{\sqrt{u_{\rm x}^2 + u_{\rm y}^2 + (u_{\rm z} + g)^2}} \geq \cos\phi_{\rm max}. \tag{17}$$

In short, (14) and (17) give the nonlinear constraints expressed in terms of the components of the control vector $\mathbf{u}$. From the original form of the constraints as defined in Problem 1, one can visualize them as a conic space with spherical inferior and superior lids, as illustrated in Figure 3a. Note that, besides nonlinear, the original constraint space is non-convex.

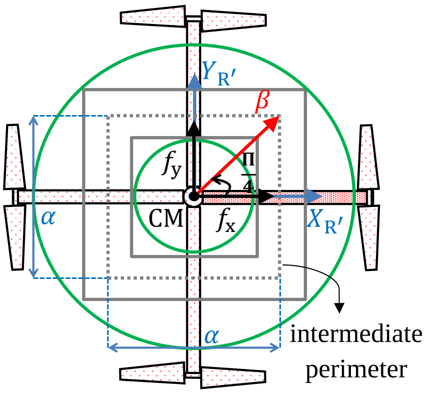

Figure 4. Constraints top view.

3.2.1 Linear Constraints

A linear suboptimal constraint space is now obtained as being the pyramidal space circumscribed in the original constraint space of Figure 3a. The new space is depicted in Figure 3b.

From Figure 3b, the new constraint along the $Z_{\mathrm{R}'}$ axis can immediately expressed as

$$f_{\min} \leq f_{\mathrm{z}} \leq f_{\max} \cos \phi_{\max}. \tag{18}$$

Consider an arbitrary f_{z} and let it represent the $Z_{\mathrm{R}'}$ axis coordinate of the dotted perimeter represented in Figure 3b. The corresponding projection on the $X_{\mathrm{R}'} - Y_{\mathrm{R}'}$ plane is given in Figure 4. The later figure defines the distances α and β and enables to expresses them immediately as

$$\alpha = 2\beta \cos \frac{\pi}{4}, \tag{19}$$

and

$$\beta = f_{\mathrm{z}} \tan \phi_{\max}. \tag{20}$$

Now, replace (20) into (19) to obtain

$$\alpha = \sqrt{2} f_{\mathrm{z}} \tan \phi_{\max} \tag{21}$$

and note that the linear constraints along the horizontal axes are given by $-\alpha/2 \leq f_{\mathrm{x}}, f_{\mathrm{y}} \leq \alpha/2$ to finally obtain

$$-\frac{\sqrt{2}}{2} f_{\mathrm{z}} \tan \phi_{\max} \leq f_{\mathrm{x}} \leq \frac{\sqrt{2}}{2} f_{\mathrm{z}} \tan \phi_{\max}, \tag{22}$$

and

$$-\frac{\sqrt{2}}{2} f_{\mathrm{z}} \tan \phi_{\max} \leq f_{\mathrm{y}} \leq \frac{\sqrt{2}}{2} f_{\mathrm{z}} \tan \phi_{\max}. \tag{23}$$

Rewriting (18), (22)-(23) in matrix form yields

$$\mathbf{\Lambda}\,\mathbf{f} \leq \boldsymbol{\lambda}, \tag{24}$$

where

$$\mathbf{\Lambda} = \begin{bmatrix} -1 & 0 & -\frac{\sqrt{2}}{2} \tan \phi_{\max} \\ 1 & 0 & -\frac{\sqrt{2}}{2} \tan \phi_{\max} \\ 0 & -1 & -\frac{\sqrt{2}}{2} \tan \phi_{\max} \\ 0 & 1 & -\frac{\sqrt{2}}{2} \tan \phi_{\max} \\ 0 & 0 & -1 \\ 0 & 0 & 1 \end{bmatrix} \tag{25}$$

and

$$\boldsymbol{\lambda} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -f_{\min} \\ f_{\max}\cos\phi_{\max} \end{bmatrix}. \tag{26}$$

Finally, using (4), (24) can be immediately reformulate in terms of **u** as

$$\bar{\boldsymbol{\Lambda}}\mathbf{u} \leq \bar{\boldsymbol{\lambda}}, \tag{27}$$

where

$$\bar{\boldsymbol{\Lambda}} \triangleq m\boldsymbol{\Lambda} \tag{28}$$

and

$$\bar{\boldsymbol{\lambda}} \triangleq \boldsymbol{\lambda} - m\boldsymbol{\Lambda}\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}. \tag{29}$$

3.3 Model Predictive Controller

Consider the discrete-time state-space model of (10). Its incremental-input version is given by

$$\begin{array}{lcl} \boldsymbol{\xi}(k+1) & = & \tilde{\mathbf{A}}\boldsymbol{\xi}(k) + \tilde{\mathbf{B}}\Delta\mathbf{u}(k) \\ \mathbf{y}(k) & = & \tilde{\mathbf{C}}\boldsymbol{\xi}(k) \end{array}, \tag{30}$$

where

$$\boldsymbol{\xi}(k) = \begin{bmatrix} \Delta\mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix} \in \mathbb{R}^9, \tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_\mathrm{d} & \mathbf{0}_{6\times3} \\ \mathbf{C}_\mathrm{d}\mathbf{A}_\mathrm{d} & \mathbf{I}_3 \end{bmatrix} \in \mathbb{R}^{9\times9}, \tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{B}_\mathrm{d} \\ \mathbf{C}_\mathrm{d}\mathbf{B}_\mathrm{d} \end{bmatrix} \in \mathbb{R}^{9\times3} \text{ and } \tilde{\mathbf{C}} = \begin{bmatrix} \mathbf{0}_{3\times6} & \mathbf{I}_3 \end{bmatrix} \in \mathbb{R}^{3\times9},$$

where $\Delta\mathbf{x}(k) \triangleq \mathbf{x}(k) - \mathbf{x}(k-1) \in \mathbb{R}^6$ denotes the increments of the state vector, $\Delta\mathbf{u}(k) \triangleq \mathbf{u}(k) - \mathbf{u}(k-1) \in \mathbb{R}^3$ is the control input increments vector, $\mathbf{I}_3$ represents an identity matrix with dimension 3×3 as well as $\mathbf{0}_{6\times3}$ is a matrix of zeros with dimension 6×3.

On the basis of (30), one can be expressed the prediction equation to $\mathbf{y}(k)$ as

$$\hat{\mathbf{y}}_\mathrm{N} = \mathbf{G}\Delta\hat{\mathbf{u}}_\mathrm{M} + \mathbf{F}, \tag{31}$$

with

$$\mathbf{G} = \begin{bmatrix} \tilde{\mathbf{C}}\tilde{\mathbf{B}} & \mathbf{0}_{3\times3} & \dots & \mathbf{0}_{3\times3} \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{B}} & \dots & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{\mathrm{N}-1}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{\mathrm{N}-2}\tilde{\mathbf{B}} & \dots & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{\mathrm{N}-\mathrm{M}}\tilde{\mathbf{B}} \end{bmatrix} \tag{32}$$

and

$$\mathbf{F} = \begin{bmatrix} \tilde{\mathbf{C}}\tilde{\mathbf{A}} \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^2 \\ \vdots \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^\mathrm{N} \end{bmatrix} \boldsymbol{\xi}(k), \tag{33}$$

where N represents the prediction horizon, M denotes the control horizon, $\hat{\mathbf{y}}_\mathrm{N} \in \mathbb{R}^{3\mathrm{N}\times1}$ expresses the vector that contains the predicted values of the controlled output up to N steps in the future. Similarly, $\Delta\hat{\mathbf{u}}_\mathrm{M} \in \mathbb{R}^{3\mathrm{M}\times1}$ represents the predicted increments of the control input vector.

Define the operator $[\bullet]_{\mathrm{N}}$ that stacks N copies of a column vector. In addition, let

$$\mathbf{K}=\begin{bmatrix}\mu_1 & 0 & 0\\ 0 & \mu_2 & 0\\ 0 & 0 & \mu_3\end{bmatrix}\in\mathbb{R}^{3\times 3} \tag{34}$$

and

$$\mathbf{W}=\begin{bmatrix}\rho_1 & 0 & 0\\ 0 & \rho_2 & 0\\ 0 & 0 & \rho_3\end{bmatrix}\in\mathbb{R}^{3\times 3} \tag{35}$$

diagonal matrices that contain weightings of the controlled output ($\mu_1,\ \mu_2,\ \mu_3$) and the control input increments ($\rho_1,\ \rho_2,\ \rho_3$). Therefore, the cost function can be written as follows:

$$J(\Delta\hat{\mathbf{u}}_{\mathrm{M}})=(\hat{\mathbf{y}}_{\mathrm{N}}-[\mathbf{r}_{\mathrm{d}}]_{\mathrm{N}})^{\mathrm{T}}\mathbf{Q}(\hat{\mathbf{y}}_{\mathrm{N}}-[\mathbf{r}_{\mathrm{d}}]_{\mathrm{N}})+\Delta\hat{\mathbf{u}}_{\mathrm{M}}^{\mathrm{T}}\mathbf{R}\Delta\hat{\mathbf{u}}_{\mathrm{M}}, \tag{36}$$

where

$$\mathbf{Q}=\begin{bmatrix}\mathbf{K} & \mathbf{0}_{3\times3} & \cdots & \mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3} & \mathbf{K} & \cdots & \mathbf{0}_{3\times3}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \cdots & \mathbf{K}\end{bmatrix} \tag{37}$$

and

$$\mathbf{R}=\begin{bmatrix}\mathbf{W} & \mathbf{0}_{3\times3} & \cdots & \mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3} & \mathbf{W} & \cdots & \mathbf{0}_{3\times3}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \cdots & \mathbf{W}\end{bmatrix}. \tag{38}$$

By replacing (31) into (36), the cost function can be immediately rewritten as

$$\begin{aligned}J(\Delta\hat{\mathbf{u}}_{\mathrm{M}}) &= (\mathbf{G}\Delta\hat{\mathbf{u}}_{\mathrm{M}}+\mathbf{F}-[\mathbf{r}_{\mathrm{d}}]_{\mathrm{N}})^{\mathrm{T}}\mathbf{Q}(\mathbf{G}\Delta\hat{\mathbf{u}}_{\mathrm{M}}+\mathbf{F}-[\mathbf{r}_{\mathrm{d}}]_{\mathrm{N}})+\Delta\hat{\mathbf{u}}_{\mathrm{M}}^{\mathrm{T}}\mathbf{R}\Delta\hat{\mathbf{u}}_{\mathrm{M}}\\ &= \tfrac{1}{2}\Delta\hat{\mathbf{u}}_{\mathrm{M}}^{\mathrm{T}}\mathbf{\Gamma}\Delta\mathbf{u}_{\hat{\mathrm{M}}}+\mathbf{\Psi}^{\mathrm{T}}\Delta\hat{\mathbf{u}}_{\mathrm{M}}+(\mathbf{F}-[\mathbf{r}_{\mathrm{d}}]_{\mathrm{N}})^{\mathrm{T}}\mathbf{Q}(\mathbf{F}-[\mathbf{r}_{\mathrm{d}}]_{\mathrm{N}}),\end{aligned} \tag{39}$$

where $\mathbf{\Gamma}=2(\mathbf{G}^{\mathrm{T}}\mathbf{Q}\mathbf{G}+\mathbf{R})$ is the positive definite Hessian matrix and $\mathbf{\Psi}=2\mathbf{G}^{\mathrm{T}}\mathbf{Q}(\mathbf{F}-\mathbf{r}_{\mathrm{N,d}})$.

To solve the optimization problem it is necessary rewriting the set of inequalities (27) in terms of the predicted increments of the control input vector $\Delta\hat{\mathbf{u}}_{\mathrm{M}}$ using the following expression

$$\hat{\mathbf{u}}_{\mathrm{M}}=[\mathbf{u}(k-1)]_{\mathrm{M}}+\mathbf{T}_{\mathrm{M}}\Delta\hat{\mathbf{u}}_{\mathrm{M}}, \tag{40}$$

where

$$\mathbf{T}_{\mathrm{M}}=\begin{bmatrix}\mathbf{I}_3 & 0_{3\times3} & \cdots & 0_{3\times3}\\ \mathbf{I}_3 & \mathbf{I}_3 & \cdots & 0_{3\times3}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{I}_3 & \mathbf{I}_3 & \cdots & \mathbf{I}_3\end{bmatrix} \tag{41}$$

is a lower block-triangular matrix of identities and $[\bullet]_{\mathrm{M}}$ is an operator that stacks M copies of a column vector, $\hat{\mathbf{u}}_{\mathrm{M}}\in\mathbb{R}^{3\mathrm{M}\times1}$ denotes the predictions of the control input up to M steps in the future as well as $\mathbf{u}(k-1)\in\mathbb{R}^3$ represents the control input used one sampling instant before.

Therefore, rewrite (27) in the prediction form

$$\bar{\bar{\mathbf{\Lambda}}}\hat{\mathbf{u}}_{\mathrm{M}}\leq[\bar{\boldsymbol{\lambda}}]_{\mathrm{M}}, \tag{42}$$

where

$$\bar{\bar{\mathbf{\Lambda}}}=\begin{bmatrix}\bar{\mathbf{\Lambda}} & \mathbf{0}_{6\times3} & \cdots & \mathbf{0}_{6\times3}\\ \mathbf{0}_{6\times3} & \bar{\mathbf{\Lambda}} & \cdots & \mathbf{0}_{6\times3}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \cdots & \bar{\mathbf{\Lambda}}\end{bmatrix}. \tag{43}$$

Then, replacing (40) into (42) the minimization of the cost (39) is subject to the following linear constraints on $\Delta\hat{\mathbf{u}}_{\mathrm{M}}$

$$\bar{\bar{\boldsymbol{\Lambda}}}\mathbf{T}_{\mathrm{M}}\Delta\hat{\mathbf{u}}_{\mathrm{M}} \leq [\bar{\boldsymbol{\lambda}}]_{\mathrm{M}} - \bar{\bar{\boldsymbol{\Lambda}}}[\mathbf{u}(k-1)]_{\mathrm{M}}. \tag{44}$$

The set of constraints (44) get the form $\mathbf{S}\Delta\hat{\mathbf{u}}_{\mathrm{M}} \leq \mathbf{b}$. Therefore, the optimization problem is a quadratic programming, in other words it is a quadratic cost subject to linear constraints and can be solved for some numerical algorithms which are available on the literature.

Remark. Only the first element of the optimized control increments sequence $\Delta\mathbf{u}^*(k) = [\mathbf{I}_3\ \mathbf{0}_{3\times3(\mathrm{M}-1)}]\Delta\hat{\mathbf{u}}^*_{\mathrm{M}}$ is utilized. The control command that is effectively applied on the plant is obtained by $\mathbf{u}^*(k) = \mathbf{u}(k-1) + \Delta\mathbf{u}^*(k)$. The optimization process is repeated at the next sampling instant based on new sensors measurements.

3.4 Computing Thrust Magnitude and Attitude Commands

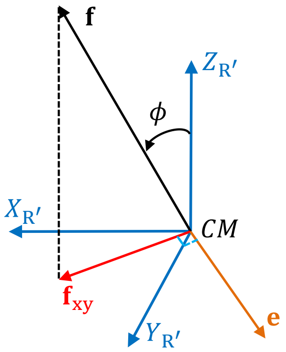

Figure 5. Relation between the thrust vector **f** and its horizontal projection $\mathbf{f}_{\mathrm{xy}}$.

After the optimization process of the control input, it is necessary to convert **u** to the total thrust f and the attitude reference angles, which will be utilized as a set-point for some inner attitude control loop. Thus, rewriting (4) as

$$\begin{bmatrix} f_{\mathrm{x}} \\ f_{\mathrm{y}} \\ f_{\mathrm{z}} \end{bmatrix} = m \begin{bmatrix} u_{\mathrm{x}} \\ u_{\mathrm{y}} \\ u_{\mathrm{z}} + g \end{bmatrix}, \tag{45}$$

Then, by taking the norm of (45), we can define f as

$$f = m\sqrt{u_{\mathrm{x}}^2 + u_{\mathrm{y}}^2 + (u_{\mathrm{z}} + g)^2}. \tag{46}$$

The attitude commands for the inner control loop need to be computed from the vector **f**. However, it is can be easily seen that there are infinite attitude of S_{B} to represent the multirotor orientation in S_{R}, taking in account when the Z_B axis coincides with **f**. Therefore, it is necessary to specify an unique attitude to the aerial robot. It can be done utilizing the attitude represented by the principal Euler angle/axis (ϕ, **e**). Define the principal Euler axis $\mathbf{e} \in \mathbb{R}^3$, shown in Figure (5), as

$$\mathbf{e} = \frac{Z_{\mathrm{R}'} \times \mathbf{f}_{\mathrm{xy}}}{\|Z_{\mathrm{R}'} \times \mathbf{f}_{\mathrm{xy}}\|}, \tag{47}$$

where $\mathbf{f}_{\mathrm{xy}} \triangleq [f_{\mathrm{x}}\ f_{\mathrm{y}}]^{\mathrm{T}} \in \mathbb{R}^2$ denotes the horizontal projection of **f**.

Utilizing the (2) and (47), it is possible to represent the attitude of S_{B} in relation to S_{R} utilizing, for example, modified Rodrigues parameters (MRP)

$$\mathbf{m}_{\mathrm{ref}} = \mathbf{e}\tan\frac{\phi}{4}, \tag{48}$$

where $\mathbf{m}_{\mathrm{ref}} \triangleq [m_{\mathrm{x}}\ m_{\mathrm{y}}\ m_{\mathrm{z}}]^{\mathrm{T}} \in \mathbb{R}^3$ represents the reference of attitude commands for some inner attitude control loop.

4. COMPUTATIONAL SIMULATIONS

All simulations were performed using the Simulink environment in Matlab®. The translational dynamics was simulated by integrating equation (1) through the Runge-Kutta method with an integration step of 1 ms. To solve the quadratic

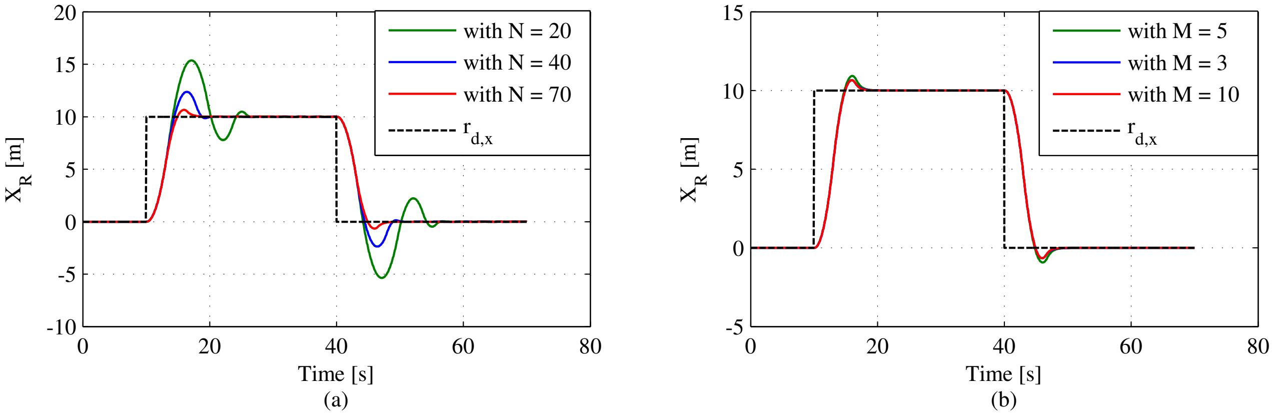

Figure 6. Adjustment of prediction and control horizon. (a) Effects of N on the response. (b) Effects of M on the response.

programming was utilized the *quadprog* function in Matlab's Optimization ToolBox. The vehicle has a mass of $m = 1$ kg and the gravitational acceleration is $g = 9.81$ m/s^2.

The first part of the study consisted in determining the appropriate values for the N and M. For this, it was admitted an initial condition for the vehicle position equal to $\mathbf{r}_{\rm i} = [0\ 0\ 10]^{\rm T}$.Then, it was applied a pulse signal in the $X_{\rm R}$ axis direction as illustrated in Figure 6. In order to evaluate the influences of N on the system response, M was adjusted to 10 and four values to N were tested ($N = 20, 40, 70$ and 80). To these simulations, it was taken into consideration $\phi_{\rm max} = 10$ deg, $f_{\rm max} = 10$ N, $f_{\rm min} = 2$ N, $\mu = [1\ 1\ 1]^{\rm T}$ and $\rho = [0.5\ 0.5\ 0.5]^{\rm T}$. The difference between $N = 70$ and $N = 80$ was negligible, thus it was preferable to choose $N = 70$ in order to avoid excessive computational calculations.

To investigate the effects of M on the system response, the previously value chosen to N was utilized and the same scenario was applied, taking into account three different values to the control horizon ($M = 3, 5$ and 10). As shown in Figure 6b, the best response occurs when $M = 10$.

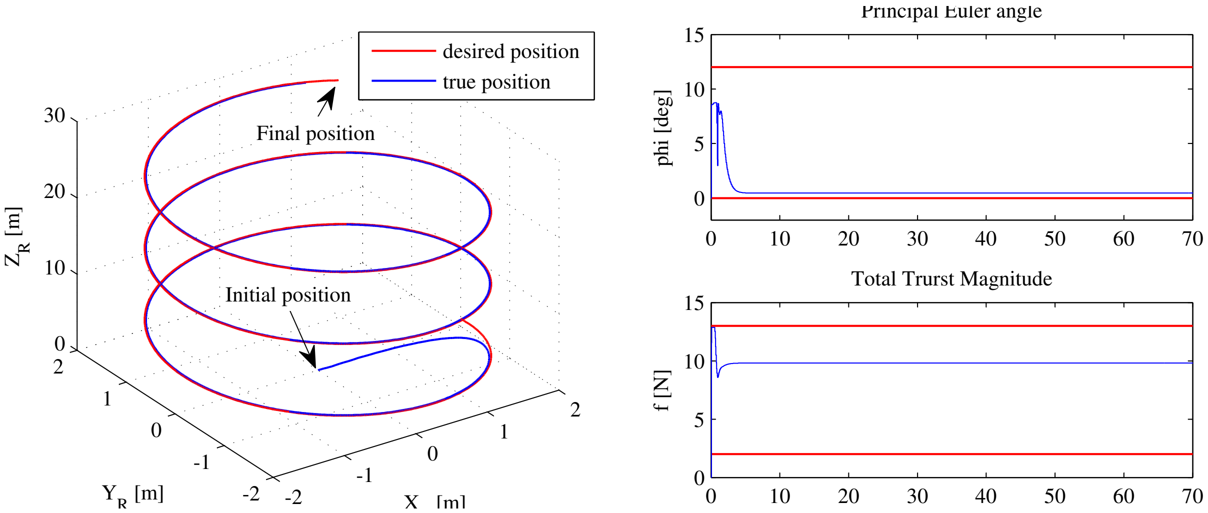

Figure 7. Spiral trajectory tracking and respective graphics of ϕ and f.

In the second simulation was verified the behavior of the vehicle following a spiral trajectory with $r_{\rm d,x}(t) = 2\cos 0.2t$, $r_{\rm d,y}(t) = -2\cos 0.2t$ and $r_{\rm d,z}(t) = 0.3t + 1$. The constraints imposed to the inclination and magnitude of **f** were replaced by $\phi_{\rm max} = 12$ deg and $f_{\rm max} = 13$ N, however other parameter settings were unchanged. The initial position of the vehicle were adjusted as $\mathbf{r}_{\rm i} = [0\ 0\ 0]^{\rm T}$, purposely to starting far from the beginning of the reference trajectory. Figure 7 shows that the proposed control strategy is able to make the vehicle follow the reference trajectory without violate the constraints imposed on ϕ and f, which are represented by the red lines on their respective graphics.

Finally, the system was subject to abrupt position commands. The MPC parameter settings were the same used in the second test. The initial position of the vehicle was adjusted to $\mathbf{r}_{\rm i} = [0\ 0\ 3]^{\rm T}$. In $t = 10$ s a step command in the $X_{\rm R}$ axis direction was applied on the system (Figure 8). One can see in Figure 9 that the inclination angle ϕ and the total thrust magnitude f suffered an abrupt change, however the controller was able to maintain the constraints inviolate. At the same time, in the $Z_{\rm R}$ graphic, one can see v$_1$ that shows a slight variation in the aircraft altitude. It happened because when the

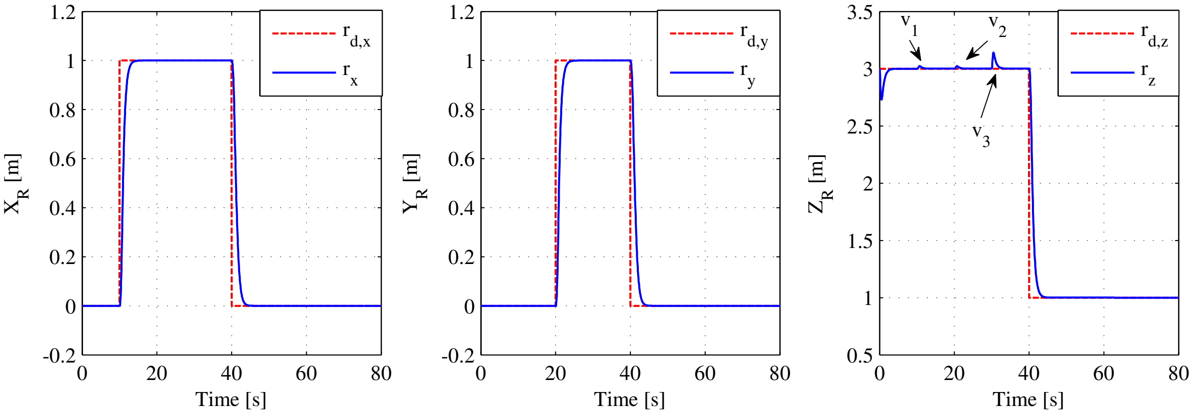

Figure 8. Abrupt reference changes.

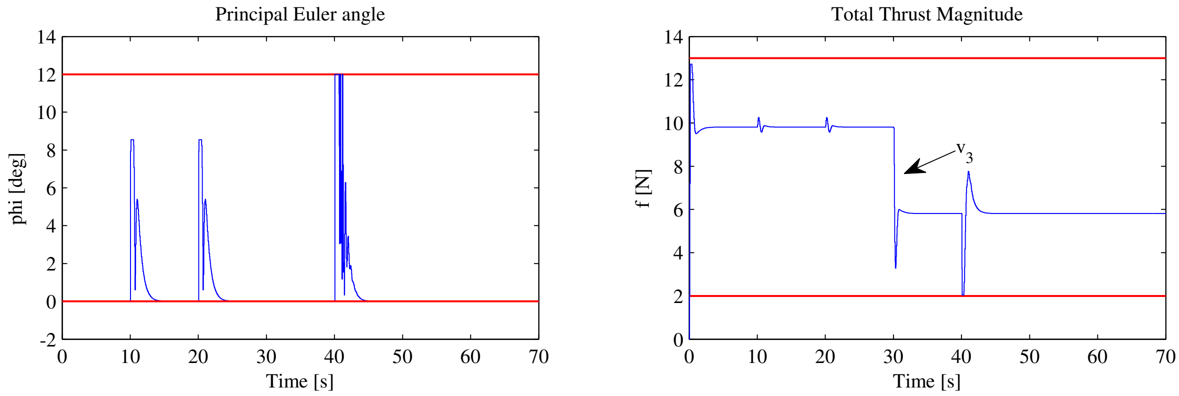

Figure 9. Abrupt variations on ϕ and f.

vehicle performed an inclination to accelerate in the X_{R} direction the controller had to increase the total trust magnitude f to avoid loss of lift. The same analysis can be realized at $t = 20$ s when was applied a step command in the Y_{R} axis direction. At $t = 30$ s an external disturbance $\mathrm{v}_3 = 4$ N was applied on the vehicle. To reject this disturbance, in the f graphic, one can see that the controller acts decreasing the total thrust magnitude. At $t = 40$ s, however, the reference signals were changed in the three variable at the same time, forcing the controller to saturating ϕ and f in their upper and lower bound, respectively. Nevertheless, the controller was able to stabilize the vehicle respecting their constraints.

5. CONCLUSIONS

This paper presented a new control strategy to solve the multirotor position tracking problem. The method considers a conic constraint on the total thrust vector. The results have presented an effective tracking of spiral and abrupt trajectories with a smooth response and without violate the constraints imposed for ϕ and f even in the present of external disturbances. Moreover, experimental tests are under preparation.

6. ACKNOWLEDGMENTS

The authors would like to thank FAPEAM/Brazil by research grants 028/2011.

7. REFERENCES

Alexis, K., Papachristos, C., Nikolakopoulos, G. and Tzes, A., 2011. “Model predictive quadrotor indoor position control”. In *Proceedings of the 19th Mediterranean Conference on Control and Automation*. Corfu, Greece.

Bouabdallar, S. and Siegwart, R., 2005. “Backstepping and sliding-mode techniques applied to an indoor micro quadrotor”. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. Barcelona, Spain.

Castillo, P., Dzul, A. and Lozano, R., 2004. “Real-time stabilization and tracking of a four-rotor mini rotorcraft”. *IEEE Transactions on Control Systems Technology*, Vol. 12, pp. 510–516.

Hoffman, G.M., Waslander, S.L. and Tomlin, C.J., 2008. “Quadrotor helicopter trajectory tracking control”. In *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*. Honolulu, Hawaii.

Lopes, R.V., Santana, P.H.R.Q.A., Borges, G.A. and Ishihara, J.Y., 2011. “Model predictive controle applied to tracking and attitude stabilization of a vtol quadrotor aircraft”. In *Proceedings of the 21st International Congress of Mechanical Engineering - COBEM*. Natal, Brazil.

Mistler, V., Benallegue, A. and M’Sirdi, N.K., 2001. “Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback”. In *Proceedings of the 2001 IEEE International Workshop on Robot and Human Interactive Communication*. Corfu, Greece, pp. 586–593.

Raffo, G.V., Ortega, M.G. and Rubio, F.R., 2010. “An integral predictive/nonlinear $\mathcal{H}_\infty$ control structure for a quadrotor helicopter”. *Automatica*, Vol. 46, pp. 29–39.

8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.