Obstacle Avoidance for an Unmanned Aerial Vehicle Model using Model Predictive Control

S. Bertrand⋆, H. Piet-Lahanier⋆, T. Hamel⋆⋆ and C. Le Tallec⋆
⋆ONERA, DPRS
29 avenue de la Division Leclerc, 92322 Châtillon, France
⋆⋆I3S-UNSA-CNRS
Les Algorithmes, Bât. Euclide B, 2000 route des Lucioles, BP 121, 06903 Sophia Antipolis, France

Abstract

This paper presents the design of guidance and control laws for a small rotorcraft-based Unmanned Aerial Vehicle model in uncertain environment with unknown obstacles. A contractive Model Predictive Control scheme is used to simultaneously handle obstacle avoidance and position and attitude stabilization for trajectory tracking. Obstacle avoidance is achieved by considering successively two methods: penalty terms in the cost function or state constraints. Simulation results are presented for these two cases.

1. Introduction

Miniature Unmanned Aerial Vehicles (UAV) are new flying platforms that are bound to be useful for numerous civil missions such as video supervision of road traffic, victim localization after natural disasters or inspection of buildings for maintenance. Military missions can also be imagined such as the surveillance of potentially dangerous areas.

The objective of the European project MAVDEM (Mini Aerial Vehicle DEMonstrator)[1] is to study, design and test a demonstrator of such a miniature aerial vehicle, capable of both stationary and economic fast cruise flight. Since this vehicle must be carriable by a single man, strong weight and dimensional design constraints are taken into account. As a part of a complete Unmanned Aerial System (UAS), a lightweight ground station is also to be included. To allow the operator to only focus on the supervision of the UAV mission, guidance and control laws must be designed to stabilize the vehicle.

As part of a theoretical background for studies on miniatures UAVs and MAVDEM project, this paper deals with the design of guidance and control laws for miniature Vertical Take Off and Landing (VTOL) UAVs in uncertain environments, where the location of possible obstacles may be unknown. Classically, a hierarchical decision and control system is used for autonomous vehicles: a path planner generates way points and/or reference trajectories to be tracked by a guidance and control layer [4, 8] . Trajectory planning algorithms can be designed, for instance, by using constrained optimization [5] or flatness-based methods [13]. If the UAV flies in an environment where the locations of possible obstacles are well known, trajectory planning can be achieved offline, before starting the mission. In the case of an uncertain environment, where the location of obstacles may be unknown or dynamically varying, the UAV must be equipped with a set of sensors (vision based sensors, ultrasonic telemeters, laser scanners, etc.) that can be used for real time detection. Given the positions of the newly detected obstacles, the trajectory of the UAV must then be modified with respect to the initial reference.

A first solution may consist in generating a new collision-free reference trajectory. Some approaches based on Model Predictive Control (MPC) have been proposed to generate local portions of that reference in a receding horizon way [12]. However, the real-time generation of a reference trajectory may be prohibitive. A preferred solution is to introduce a "reactive layer" between the path planner and the tracking layers, to modify the reference by taking into account the detected location of obstacles. This approach is used in [10] where the local reactive layer combines properties of attraction to a single goal point and repulsion from obstacles. A MPC layer is used in [11] to modify the reference trajectory, using terms that penalize the proximity to obstacles in the Model Predictive Control formulation.

In all these approaches, obstacle avoidance and trajectory tracking are achieved separately. Using its faculty to consider new information as it becomes available, MPC can be used to simultaneously deal with obstacle avoidance and trajectory tracking.

[1] www.mavdem-project.org

In this paper, a contractive Model Predictive Control strategy is used to solve both obstacle avoidance and control problems for a six degrees of freedom model of a VTOL UAV. Based on the formulation of [3], a stabilizing constraint is introduced in the optimization problem used to compute the control, imposing a norm contraction on prediction of the state of the system, and hence guaranteeing closed loop stability for trajectory tracking [2]. Additional state constraints and/or penalty terms in the cost function are then defined and activated online to represent the set of positions of the new detected obstacles that the UAV must avoid.
In section 2, the considered model is presented as well as the control strategy. The design of guidance and control laws for trajectory tracking is detailed in section 3. Modifications of the position controller to handle obstacle avoidance is presented in section 4 as well as simulation results. The last part of this paper is devoted to some concluding remarks.

2. Model of UAV and control strategy

2.1 Model of VTOL UAV

The VTOL UAV is represented by a rigid body of mass m and of tensor of inertia I. Two reference frames are used to parameterize its motion (cf Figure 1): an inertial reference frame $(\mathcal{I}) = (e_1, e_2, e_3)$ and a body frame $(\mathcal{B}) = (e_1^b, e_2^b, e_3^b)$ attached to the UAV. We respectively denote by $\xi = [x\ y\ z]^T$ and $v = \left[v_x\ v_y\ v_z\right]^T$ the position and the linear velocity of the UAV in $(\mathcal{I})$. The orientation of the UAV is given by the orientation matrix $R \in SO(3)$ from $(\mathcal{I})$ to $(\mathcal{B})$, and usually parameterized by Euler's pseudo angles ψ, θ, ϕ (yaw-pitch-roll):

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

with the trigonometric shorthand notations $c_\alpha = cos(\alpha)$, $s_\alpha = sin(\alpha)$, $\forall \alpha \in \mathbb{R}$. The angular velocity of the UAV is denoted in $(\mathcal{B})$ by $\Omega = [p\ q\ r]^T$.

Let F be the translational force applied to the UAV, combining thrust, drag, lift and gravity components. For quasi-stationary flight, where the lift force predominates the other components, we can assume that the aerodynamic forces are always in the direction e_3^b [6]. Therefore, the force F can be written as $F = -\mathcal{T} Re_3 + mge_3$, where $\mathcal{T}$ represents the magnitude of the external forces applied to the UAV in direction e_3^b. We denote by $\Gamma = [\Gamma_1\, \Gamma_2\, \Gamma_3]^T$ the torque applied to the UAV in $(\mathcal{B})$.
For a given vector $a \in \mathbb{R}^3$, we denote by $a_\times$ its skew symmetric matrix, which satisfies $\forall b \in \mathbb{R}^3$, $a_\times b = a \times b$, for the vector cross product $\times$. The VTOL UAV dynamics are represented by:

$$\begin{cases} \dot{\xi} = v \\ m\dot{v} = -\mathcal{T}\, Re_3 + mge_3 \\ \dot{R} = R\,\Omega_\times \\ I\dot{\Omega} = -\Omega \times I\Omega + \Gamma \end{cases} \tag{2}$$

where the control inputs to be considered are $\mathcal{T}$ and Γ.

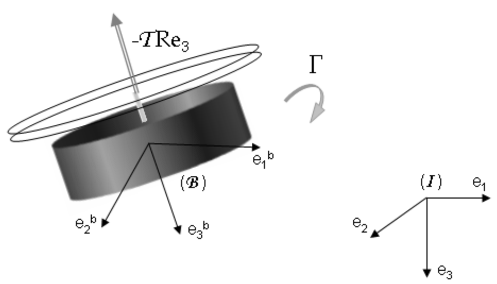

Figure 1: Reference frames.

2.2 Control strategy

We assume that a reference trajectory is given for the UAV in terms of a desired position (ξ^d), yaw (ψ^d), and the time derivatives $v^d = \dot{\xi}^d$ and $\dot{v}^d = \ddot{\xi}^d$.
The controller is designed according to the cascaded approach presented in [2]: separate controllers are developed for the translational dynamics and for the orientation dynamics, leading to a time-scale separation between these two connected subsystems.
For the translational dynamics, the full vectorial term $\mathcal{T}Re_3$ will be defined as the position control input ($\mathcal{T}Re_3 = f(\xi, v, \xi^d, v^d, \dot{v}^d)$). It is split into its magnitude $\mathcal{T} = \left\|f(\xi, v, \xi^d, v^d, \dot{v}^d)\right\|$ representing the first control input, and its direction $R^d e_3 = \frac{1}{\mathcal{T}} f(\xi, v, \xi^d, v^d, \dot{v}^d)$, representing the desired orientation. The desired orientation R^d can then be deduced from the given direction $R^d e_3$, solving for (ψ, θ, ϕ) for a given specified yaw trajectory ψ^d [7].
For the orientation dynamics, we will assign the control torque Γ such that the orientation R of the UAV converges to the desired orientation R^d, and such that the angular velocity Ω converges to Ω^d defined by:

$$\dot{R}^d = R^d \Omega^d_\times \tag{3}$$

3. Guidance and control laws for trajectory tracking

The contractive Model Predictive Control scheme presented in [3] is applied to design each of the two controllers. Since the orientation controller will not converge instantaneously, the translational dynamics may be reformulated to introduce an orientation error term:

$$\begin{cases} \dot{\xi} = v \\ m\dot{v} = -\mathcal{T} R^d e_3 + mge_3 - \mathcal{T}(R - R^d)e_3 \\ \dot{R} = R\,\Omega_\times \\ I\dot{\Omega} = -\Omega \times I\Omega + \Gamma \end{cases} \tag{4}$$

If the orientation controller is asymptotically stabilizing and the input $\mathcal{T}$ is bounded, the orientation error term $-\mathcal{T}(R - R^d)e_3$ may be considered as an additive decaying perturbation and can be handled by the contractive MPC approach with guaranteed stability.

3.1 Position controller

Consider the translational dynamics of (4). For trajectory tracking, let us introduce the error variables

$$\epsilon_1 = \xi - \xi^d, \quad \epsilon_2 = v - v^d, \quad \delta_1 = \frac{m}{k_1}\epsilon_2 + \epsilon_1 \tag{5}$$

with $k_1 > 0$. The translational dynamics become:

$$\begin{cases} \dot{\epsilon}_1 = -\frac{k_1}{m}\epsilon_1 + \frac{k_1}{m}\delta_1 \\ \dot{\delta}_1 = -\frac{k_1}{m}\epsilon_1 + \frac{k_1}{m}\delta_1 - \frac{\mathcal{T}}{k_1}R^d e_3 + \frac{m}{k_1}ge_3 - \frac{m}{k_1}\dot{v}^d - \frac{\mathcal{T}}{k_1}(R - R^d)e_3 \end{cases} \tag{6}$$

Defining

$$X = \left[\epsilon_1^T \; \delta_1^T\right]^T, \quad u = -\frac{\mathcal{T}}{k_1}R^d e_3 + \frac{m}{k_1}ge_3 - \frac{m}{k_1}\dot{v}^d, \quad d = -\frac{\mathcal{T}}{k_1}(R - R^d)e_3 \tag{7}$$

the dynamics (6) can be written as:

$$\dot{X} = f_1(X, u) + d \tag{8}$$

Using the notations of [3], we define by P_p the length of the horizon of prediction and by T_s the sampling time. For $k \in \mathbb{N}$ we define $t_k = t_0 + kP_pT_s$ and divide the interval $[t_k,\ t_{k+1}]$ in P_p parts as follows:

$$\begin{aligned} t_k^j &= t_k + jT_s\,,\ \forall j \in \left\{0,\ \dots,\ P_p\right\} \\ t_k^0 &= t_k \\ t_k^{P_p} &= t_{k+1} \end{aligned}$$

We assume that the control u is piecewise-constant over the prediction horizon and introduce the following control sequence:

$$u_k^j(.) = \left\{u(t_k^j|t_k^j),\ u(t_k^{j+1}|t_k^j),\ \ldots,\ u(t_k^{j+P_p-1}|t_k^j)\right\}$$

Let us define $X_k^j = X(t_k^j)$. At time t_k^j, the control is computed by solving the optimization problem:

$$\begin{aligned} &\min_{u_k^j(.)} \int_{t_k^j}^{t_k^{j+P_p}} \left\{\left\|\bar{X}_k^j(\tau)\right\|_{Q_1}^2 + \left\|u_k^j(\tau)\right\|_{S_1}^2\right\} d\tau \\ &\text{s.t. } \mathcal{L}_1(\check{X}_k^j(t_{k+1})) \leq \alpha_1 \mathcal{L}_1(X_k) \\ &\quad u_{min} \leq u(t_k^{j+i}|t_k^j) \leq u_{max}\ \forall i = 0, \ldots, P_p - 1 \end{aligned} \tag{9}$$

with Q_1 and S_1 two symmetric positive definite weighting matrices, and with $0 \leq \alpha_1 < 1$. The positive definite function $\mathcal{L}_1$ used to evaluate the contractive constraint is defined by:

$$\mathcal{L}_1 = \frac{1}{2}\epsilon_1^T \epsilon_1 + \frac{1}{2}\delta_1^T \delta_1 \tag{10}$$

The predictions $\bar{X}$ are computed using the following model

$$\begin{cases} \dot{\bar{X}}_k^j(t) = f_1(\bar{X}_k^j(t), u_k^j(t)) \\ u_k^j(t) = u(t_k^{j+i}|t_k^j) \\ \bar{X}_k^j = X_k^j \\ \forall t \in \left[t_k^{j+i},\ t_k^{j+i+1}\right],\ \forall i \in \left\{0,\ \ldots,\ P_p - 1\right\} \end{cases} \tag{11}$$

whereas the value $\check{X}$, used in the contractive constraint, is computed as follows:

$$\begin{cases} \dot{\check{X}}_k^j(t) = f_1(\check{X}_k^j(t), u_k^j(t)) \\ u_k^j(t) = u(t_k^{j+i}|t_k^j) \\ \check{X}_k^j = \begin{cases} \check{X}_k^{j-1}(t_k^j),\ \forall j \geq 1 \\ X_k,\ \text{for } j = 0 \end{cases} \\ \forall t \in \left[t_k^{j+i},\ t_k^{j+i+1}\right],\ \forall i \in \left\{0,\ \ldots,\ P_p - 1\right\} \end{cases} \tag{12}$$

Only the first component $u^*(t_k^j|t_k^j)$ of the optimal solution $u_k^{*j}(.)$ is used to calculate the input

$$\mathcal{T}^*(t_k^j) = \left\|k_1 u^*(t_k^j|t_k^j) - mge_3 + m\dot{v}^d(t_k^j)\right\| \tag{13}$$

applied to the translational dynamics over $\left[t_k^j,\ t_k^{j+1}\right]$. As discussed in section 2.2, the desired value of the orientation can be computed using

$$R^d(t_k^j)e3 = \frac{1}{\mathcal{T}^*(t_k^j)}(-k_1 u^*(t_k^j|t_k^j) + mge_3 - m\dot{v}^d(t_k^j)) \tag{14}$$

The whole procedure is repeated at the next sampled instant to compute the following control value at t_k^{j+1}.

The input bounds u_{min} and u_{max} in the optimization problem formulation may be chosen to ensure boundedness properties for $\mathcal{T}^*$ as presented in [2] Lemma 1 and ensure that the relation (14) used for the computation of the desired orientation is well defined.

3.2 Attitude controller

Let us consider the orientation dynamics given by the last two equations of (4). The deviation between R and R^d can be defined by introducing

$$\tilde{R} = (R^d)^T R \tag{15}$$

Using $\tilde{R}\tilde{R}^T = I_d$ (where I_d is the identity matrix of $\mathbb{R}^{3\times3}$) and $\tilde{R}^T \Omega_\times^d \tilde{R} = (\tilde{R}^T \Omega^d)_\times$, the time derivative of (15) may be written as

$$\dot{\tilde{R}} = \tilde{R}\tilde{\Omega}_\times \tag{16}$$

with

$$\tilde{\Omega} = \Omega - \tilde{R}^T \Omega^d \tag{17}$$

The orientation dynamics become:

$$\begin{cases} \dot{\tilde{R}} = \tilde{R}\tilde{\Omega}_\times \\ I\dot{\tilde{\Omega}} = -(\tilde{\Omega} + \tilde{R}^T\Omega^d)_\times I(\tilde{\Omega} + \tilde{R}^T\Omega^d) + \Gamma + I\tilde{\Omega}_\times \tilde{R}^T\Omega^d - I\tilde{R}^T\dot{\Omega}^d \end{cases} \tag{18}$$

The equilibrium ($\tilde{R} = I_d$, $\Omega = \Omega^d$) is achieved for $\Gamma = \Omega^d \times I\Omega^d + I\dot{\Omega}^d$. Let us define $\gamma = \Gamma - \Omega^d \times I\Omega^d - I\dot{\Omega}^d$. We denote by P_r the length of the horizon of prediction. For $k' \in \mathbb{N}$ we define $t_{k'} = t_0 + k'P_rT_s$ and divide the interval $[t_{k'},\ t_{k'+1}]$ in P_r parts as follows:

$$\begin{aligned} t_{k'}^j &= t_{k'} + jT_s\ ,\ \forall j \in \{0,\ \ldots,\ P_r\} \\ t_{k'}^0 &= t_{k'} \\ t_{k'}^{P_r} &= t_{k'+1} \end{aligned}$$

Similarly to the position controller design, we assume that γ is piecewise-constant over the prediction horizon and introduce the following control sequence:

$$\gamma_{k'}^j(.) = \left\{\gamma(t_{k'}^j|t_{k'}^j),\ \gamma(t_{k'}^{j+1}|t_{k'}^j),\ \ldots,\ \gamma(t_{k'}^{j+P_r-1}|t_{k'}^j)\right\}$$

At time $t_{k'}^j$ the control is computed by solving the optimization problem:

$$\begin{aligned} \min_{\gamma_{k'}^j(.)} &\int_{t_{k'}^j}^{t_{k'}^{j+P_r}} \{\ \lambda \mathrm{tr}(I_d - \bar{\tilde{R}}_{k'}^j(\tau)) + \left\|\bar{\tilde{\Omega}}_{k'}^j(\tau)\right\|_{Q_2}^2 + \left\|\gamma_{k'}^j(\tau)\right\|_{S_2}^2\ \}\, d\tau \\ \text{s.t.} \quad & \mathcal{L}_2(\check{\tilde{R}}_{k'}^j(t_{k'+1}), \check{\tilde{\Omega}}_{k'}^j(t_{k'+1})) \leq \alpha_2 \mathcal{L}_2(\tilde{R}_{k'}, \tilde{\Omega}_{k'}) \\ & \gamma_{min} \leq \gamma(t_{k'}^{j+i}) \leq \gamma_{max} \ \ \forall i = 0, \ldots, P_r - 1 \end{aligned} \tag{19}$$

with Q_2 and S_2 two symmetric definite positive weighting matrices, and with $\lambda > 0$ and $0 \leq \alpha_2 < 1$. The positive definite function $\mathcal{L}_2$ used to evaluate the contractive constraint is defined by:

$$\mathcal{L}_2 = \frac{1}{2}\mathrm{tr}(I_d - \tilde{R}) + \frac{1}{2}\tilde{\Omega}^T I\tilde{\Omega} \tag{20}$$

The predictions and the quantities used for the evaluation of the contractive constraint are both computed using equations of (18) with the same formalism as introduced for the position controller design.
The first component $\gamma^*(t_{k'}^j|t_{k'}^j)$ of the optimal solution $\gamma^{*j}_{k'}(.)$ is used to compute the input

$$\Gamma^*(t_{k'}^j) = \gamma^*(t_{k'}^j|t_{k'}^j) + \Omega^d(t_{k'}^j) \times I\Omega^d(t_{k'}^j) + I\dot{\Omega}^d(t_{k'}^j) \tag{21}$$

which is applied to the orientation dynamics during a sampling period T_s. The whole procedure is repeated at the next sampled instant to compute the value of the control torque at $t_{k'}^{j+1}$.

3.3 Closed loop stability for trajectory tracking

The system (4) controlled by the proposed position and attitude controllers is asymptotically stable for trajectory tracking. Elements of the proof are based on results of [3] and can be found in [2]. The predictive nature of the proposed control scheme has been shown to be well suitable for trajectory tracking, in the sense that changes in the reference trajectory are anticipated and hence leads to good tracking performances.
That predictive nature of MPC can also be used to anticipate possible future collisions with the detected obstacles and hence modify the control action for obstacle avoidance.

4. Obstacle avoidance

For obstacle avoidance, we consider the optimization problem (9) used to define the position controller. Since that problem has to be solved at each sampling instant, its formulation can be modified online as soon as new informations on the location of detected obstacles become available.
We assume that a navigation algorithm is used to provide, from sensor measurements, the number N^{obs} of detected

obstacles to be considered, and their respective locations. We also focuse on convex obstacles.

Two methods are considered: penalty terms in the cost function and state constraints. Simulation results are provided for each case. For simulation, the reference trajectory to be tracked is defined by a segment at constant altitude starting from the initial position $\xi^d(0) = [0\ 0\ 2]^T$. The desired yaw $\psi^d(t)$ is chosen to be zero. We consider the case N^{obs}=2 with a spherical obstacle centered in $\xi_1^{obs} = [2\ 2\ 2]^T$ with radius $r_1^{obs} = 0.5$, and a vertical cylinder with infinite height and radius $r_2^{obs} = 0.5$ and the axis of which intersects the reference trajectory in $\xi_2^{obs} = [6\ 6\ 2]^T$.

4.1 Penalty terms in the cost function

The obstacles are represented by the points $\xi_i^{obs} = [x_i\ y_i\ z_i]^T$ with $i \in \left\{1, \ldots, N^{obs}\right\}$. A term is added in the cost function of (9) to penalize proximity to the obstacles. The optimization problem used to compute the position control is reformulated as

$$\begin{aligned} \min_{u_k^j(.)} & \int_{t_k^j}^{t_k^{j+P_p}} \left\{ \left\|\bar{X}_k^j(\tau)\right\|_{Q_1}^2 + \left\|u_k^j(\tau)\right\|_{S_1}^2 + \sum_{i=1}^{N^{obs}} \frac{K_i}{\epsilon + \left\|\bar{\xi}(\tau) - \xi_i^{obs}(\tau)\right\|_{Q_i^{obs}}^2} \right\} d\tau \\ \text{s.t.} \ & \mathcal{L}_1(\check{X}_k^j(t_{k+1})) \leq \alpha_1 \mathcal{L}_1(X_k) \\ & u_{min} \leq u(t_k^{j+i}|t_k^j) \leq u_{max}\ \forall i = 0, \ldots, P_p - 1 \end{aligned} \tag{22}$$

with $K_i \in \mathbb{R}^+$ and where ϵ is a small strictly positive constant introduced to avoid singularities [11]. The symmetric positive definite weighting matrix Q_i^{obs} can be used to define preferred directions of avoidance: small coefficient for the z-axis, for example, will lead to obstacle avoidance in the *xy*-plan.
The simulation parameters are $K_1 = K_2 = 15$, $Q_1^{obs} = \text{diag}(1, 1, 4)$ and $Q_2^{obs} = \text{diag}(1, 1, 0)$. Controller parameters and UAV model parameters are presented in table 1. The values of Q_1^{obs} and Q_2^{obs} have been chosen to respectively achieve vertical avoidance and avoidance in the *xy*-plan.

With that representation, the shape of the obstacles is not taken into account. A good tuning of the parameters K_i and Q_i^{obs} may lead to obstacle avoidance but without guarantee that the UAV trajectory will not intersect the volumes defining the obstacles. Indeed, as can be seen in figure 2, the closed loop trajectory of the UAV is deviated from the reference trajectory but is not collision-free, since it intersects the first obstacle.

Using penalty terms, obstacle avoidance may be achieved by representing, at each instant, a given obstacle by the coordinates of its point which is the closest to the UAV. Another solution consists in representing the obstacle by a set of points defining its shape. In that case, a representation based on state constraints can be used.

4.2 State constraints

The obstacles are represented by ellipsoids. For a given obstacle i centered in $\xi_i^{obs} = [x_i\ y_i\ z_i]^T$, the set of coordinates that the UAV must avoid is represented by a quadratic constraint

$$(\bar{\xi}(\tau) - \xi_i^{obs})^T P_i^{obs} (\bar{\xi}(\tau) - \xi_i^{obs}) > r_i^{obs} \quad \forall \tau \in \left[t_k^j, t_k^{j+P_p}\right] \quad \forall i \in \left\{1, \ldots, N^{obs}\right\} \tag{23}$$

which is added to the optimization problem (9).

The symmetric positive definite weighting matrices P_i^{obs} chosen for simulation are $P_1^{obs} = \text{diag}(1, 1, 1)$ and $P_2^{obs} = \text{diag}(1, 1, 0)$.

Table 1: Controller and UAV parameters

Position Controller	Attitude Controller	UAV model
$k_1 = 3.2$	$\lambda = 1$	
$P_p T_s = 3$ s	$P_r T_s = 1$ s	$m = 8$ kg
$Q_1 = \text{diag}(2, 2, 2, 5, 5, 5)$	$Q_2 = I_d$	$I = \text{diag}(0.32, 0.32, 0.52)$ kg.m^2
$S_1 = I_d$	$S_2 = I_d$	$g = 9.81$ m.s^{-2}
$\alpha_1 = 0.5$	$\alpha_2 = 0.1$	

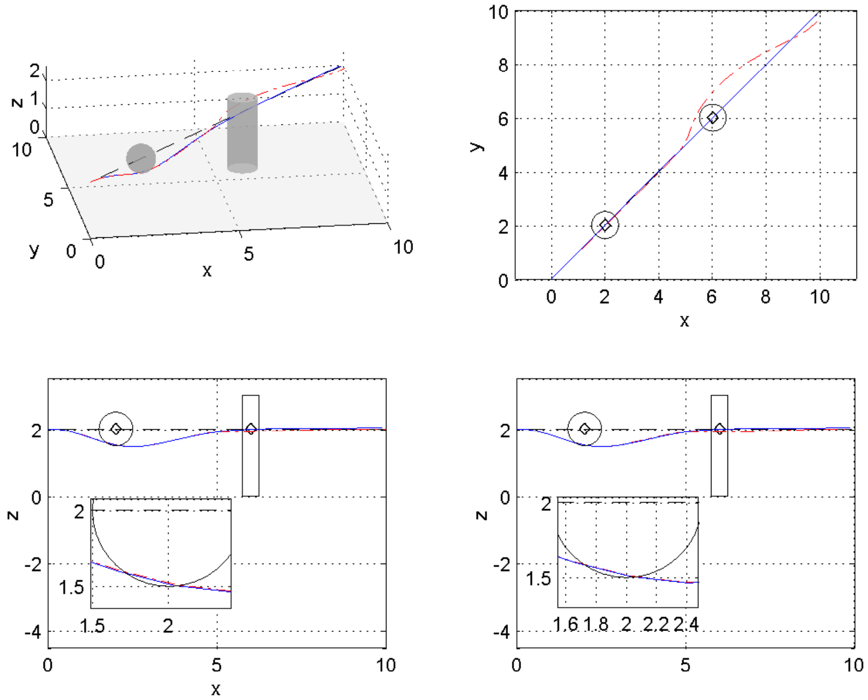

Figure 2: Obstacle avoidance using penalty terms
Reference trajectory (dashed line), closed loop trajectories for avoidance
of obstacle 1 only (solid line) or obstacles 1 and 2 (dash-dot line)

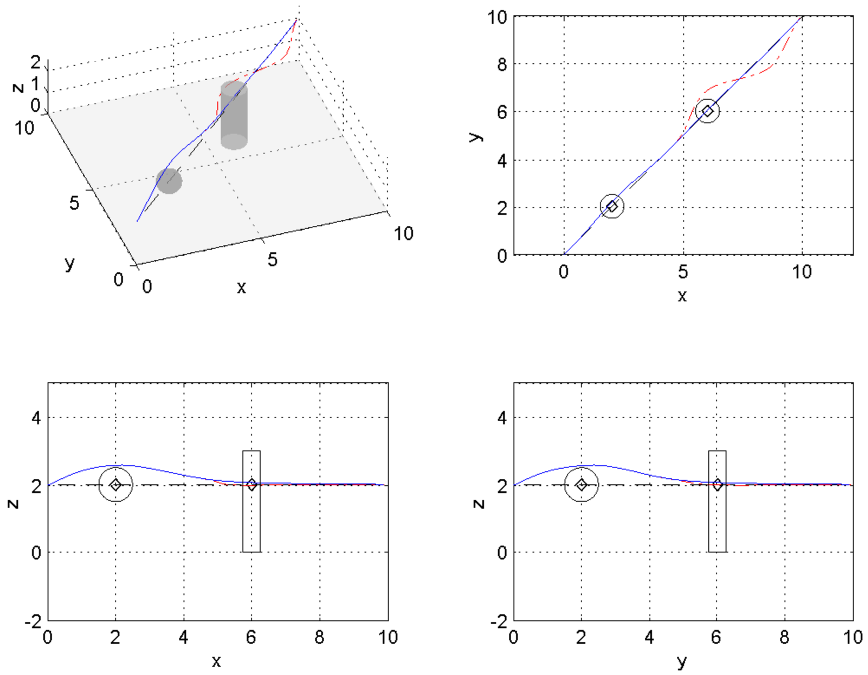

Figure 3: Obstacle avoidance using state constraints
Reference trajectory (dashed line), closed loop trajectories for avoidance
of obstacle 1 only (solid line) or obstacles 1 and 2 (dash-dot line)

The closed loop trajectories obtained in the case of presence of the first obstacle only, and in the case of the two obstacles are plotted in figure 3. Avoidance is well achieved, using privilegiate directions for each kind of obstacle: avoidance of the sphere by vertical evasion, and avoidance of the cylinder by avoidance maneuver in the *xy*-plan.

5. Conclusion

We have presented the design of guidance and control laws for trajectory tracking of a Vertical Take Off and Landing Unmanned Aerial Vehicle model in an uncertain environment, where the location of obstacles may be a priori unknown. In that case, the pre-planned reference trajectory may lead to collision and the vehicle must be capable of obstacle avoidance. Based on a contractive Model Predictive Control scheme, a cascaded controller has been designed, composed of a position and an attitude controller. The position controller is hence modified to take into account the location of detected obstacles and achieve both obstacle avoidance and trajectory tracking. Two modifications have been presented: introduction of penalty terms in the cost function and state constraints. Both methods allow to take into account the locations of obstacles detected online, but collision free avoidance is only guaranteed by the state constraints approach. Simulation results have been presented for the two cases.

References

[1] Bertrand S., Hamel T. and Piet-Lahanier H. Stabilization of a Small Unmanned Aerial Vehicle Model without Velocity Measurement. In *2007 IEEE International Conference on Robotics and Automation*, Roma, Italy.

[2] Bertrand S., Piet-Lahanier H. and Hamel T. Contractive Model Predictive Control of an Unmanned Aerial Vehicle. In *17th IFAC Symposium on Automatic Control in Aerospace*, Toulouse, France.

[3] De Oliveira Kothare S. L. and Morari M. Contractive Model Predictive Control for Constrained Nonlinear Systems. In *IEEE Transactions on Automatic Control*, 45:6, pp1053-1071.

[4] Fabiani P., Fuertes V., Piquereau A., Mampey R., Teichteil-Konigsbuch F. Autonomous Flight and Navigation of VTOL UAVs: from Autonomy Demonstrations to Out-of-Sight Flights. In *Aerospace Science and Technology*, 11 (2007), pp183-193.

[5] Geiger B. R., Horn J. F., DeLullo A. M. and Long L. N. Optimal Path Planning of UAVs using Direct Collocation with Nonlinear Programming. In *2006 AIAA Guidance, Navigation and Control Conference*, Keyston, USA.

[6] Hamel T. and Mahony R. Pure 2D Visual Control for a Class of Under-Actuated Dynamic Systems. In *2004 IEEE International Conference on Robotics and Automation*, New Orleans, USA.

[7] Hamel T., Mahony R., Lozano R. and Ostrowski J. Dynamic Modeling and Configuration Stabilization for an X4-Flyer. In *Proceedings of the 2002th IFAC Symposium*, Barcelona, Spain.

[8] Kim H. J. and Shim D. H. A Flight Control System for Aerial Robots: Algorithms and Experiments. In *Control Engineering Practice*, 11 (2003), pp1389-1400.

[9] Pflimlin J. M., Hamel T., Souères P. and Mahony R. A Hierarchical Control Strategy for the Autonomous Navigation of a ducted fan VTOL UAV. In *2006 IEEE International Conference on Robotics and Automation*, Orlando, USA.

[10] Scherer S., Singh S., Chamberlain L., Saripalli S. Flying Fast and Low Among Obstacles. In *2007 IEEE International Conference on Robotics and Automation*, Roma, Italy.

[11] Shim D. H. Autonomous Exploration in Unknown Urban Environments for Unmanned Aerial Vehicles. In *2nd International Symposium on Innovative Aerial/Space Flyer Systems*, Tokyo, Japan, 2005.

[12] Singh L. and Fuller J. Trajectory Generation for a UAV in Urban Terrain, using Nonlinear MPC. In *Proceedings of the 2001 American Control Conference*, 3 (2001), pp2301-2308.

[13] Van Nieuwstadt M. J. and Murray R. M. Real-Time Trajectory Generation for Differentially Flat Systems. In *International Journal of Robust and Nonlinear Control*, 8:11, pp995-1020.

This page has been purposedly left blank