A STRATEGY FOR UAV PERFORMANCE ENHANCEMENT THROUGH ATMOSPHERIC ENERGY EXTRACTION USING MPC TECHNIQUES

Grégori Pogorzelski* , **Flávio José Silvestre****
***CENIC Engenharia** , ****ITA – Instituto Tecnológico de Aeronáutica**

Keywords: *Autonomous soaring; Model–Based Predictive Control; Thermal estimation; Nonlinear programming; UAV performance*

Abstract

The need for efficient propulsion systems allied to increasingly more challenging fixed–wing UAV mission requirements has led to recent research on the autonomous thermal soaring (ATS) field with promising results. As part of that effort, the feasibility and advantages of modal predictive control (MPC) based guidance and control algorithms capable of extracting energy from updrafts have already been demonstrated numerically. However, given the nature of the dominant phenomena, a nonlinear optimal control problem results. Since it may be of large order, implementation and real–time operation difficulties may arise. Knowing that, an alternative MPC–based ATS controller designed to yield a simple and small nonlinear programming problem to be solved online is presented herein. In order to accomplish that, linear prediction schemes are employed to impose the differential constraints, thus no extra variables are added to the problem and only linear bound restrictions result. For capturing the governing nonlinear effects during the climb phase, a simplified representation of the aircraft kinematics with quasi–steady corrections is used by the controller internal model. Simulation results using a 3 degree–of–freedom model subjected to a randomly generated time varying thermal environment show that the aircraft is able to locate and exploit updrafts, suggesting that the proposed algorithm is a feasible MPC strategy.

1 Introduction

For a long time the atmospheric energy potential in the form of small–scale convective air currents has been explored by glider pilots and soaring birds. Having as basic requirement the ground exposition to the solar radiation, thermals occur over different areas, from flatlands to mountains, throughout all the seasons of the year [1]. However, only recently the phenomenon has started to receive serious attention from the aeronautical engineering community as a way of enhancing the performance of powered fixed–wing aircraft. The widespread usage of unmanned aerial vehicles (UAV), that are frequently equipped with customizable digital flight control hardware, makes them natural candidates to incorporate autonomous energy harvesting capabilities, providing endurance and range gains.

A controller must run specific algorithms for dealing with the two typical flying phases associated to soaring (climbing and searching), switching between them when appropriate. Remote thermal recognition methods could be used to locate rising air zones, but they would require complex equipments to be onboard. Therefore, to allow wider and less expensive applications, the algorithms presented herein assume the UAV is equipped only with a minimal set of sensors, namely, a Pitot–static system, a Global Navigation Satellite System (GNSS) receiver and an Inertial Measurement Unit (IMU).

Rising air columns of different shapes and intensities can be found in a single mission and those parameters may rapidly change with respect to time and altitude [1]. Furthermore, the pursuit of optimal climb rates typically results in the airplane flying close to its operational limitations with large amplitude of commands to manoeuvre accordingly. Those aspects, inherent to soaring, may somehow limit the application of classical control techniques. Nevertheless the MPC approach seems to be an adequate tool for dealing with them [2], since: 1) It relies on an internal dynamic model whose parameters can be iteratively changed in real–time to describe different atmospheric scenarios; 2) The commands to be applied result from the solution of an optimal control problem which can be written to mathematically express the energy gain goal; 3) MPC algorithms can naturally incorporate constraints in such a way that they are respected by the controller during the predictions.

The conceptual feasibility of ATS controllers based on nonlinear MPC schemes has already been demonstrated. Thus, the present work proposes an alternative quasi–linear MPC approach. It leads to a minimum order nonlinear programming (NLP) optimization problem with linear inequalities only, which tends to be simpler to solve, requiring less computational resources.

1.1 Previous Works

Allen [3] proposed and successfully flight tested the first practical ATS system on a $4.27m$ span motor–glider. Total energy rate and acceleration, derived from measurements, are the basic inputs to the algorithm. For thermaling, a reference radius and corresponding steady–state turn rate are defined heuristically. The computed turn rate commands are further modified in order to respond to changes in total energy acceleration. The work by Allen [3] was extended by a series of studies [4, 5, 6, 7] that proposed algorithm updates or the inclusion and validation of new features, leading to robust and reliable designs. Autonomous flights with range and endurance of about $100km$ and $5h$ have been reported [6].

While the previously cited works employed classical control techniques, the studies by Lee, Longo and Kerrigan [8] and Liu et al. [9] investigated the ATS problem in a more formal way, writing it as an optimal control problem to be solved using a nonlinear MPC–based scheme. State and control constraints were present and different cost functions were tested for the search mode, while in climb mode the maximization of the total specific energy was the goal. The results of numerical simulations indicated that the controller was able to explore a non–homogeneous thermal environment, avoid sinking zones, identify and exploit updraft cores.

1.2 Present Work Contribution

A potential drawback of the methodology introduced by Lee, Longo and Kerrigan [8] and Liu et al. [9] is the relatively high computational costs that could hamper real–time usage. Hence, the present work aims to contribute by proposing a simpler yet practical MPC controller which could potentially lead to faster and more efficient computations allied to straightforward implementation and easier integration on flight hardware.

The ATS optimal control problem is inherently nonlinear because the updraft intensity distribution is itself nonlinear (in space and time) and significant variations on the aircraft states usually take place when climb or search manoeuvres are executed. Under those circumstances, the direct use of a linear model for prediction will lead to poor results with significant discrepancies after few seconds. Lee, Longo and Kerrigan [8] and Liu et al. [9] have used fully nonlinear 3 degree–of–freedom (DOF) point–mass equations of motion (EOM) for prediction. In order to implement the NLP solution in this framework, normally the differential constraints have to be translated to algebraic nonlinear constraints and consequently more variables (associated to the states) need to be introduced, leading to a larger problem. Furthermore, regular bound constraints become nonlinear restrictions too. As a simpler alternative, the proposed work relies on using linear prediction models as much as possi-

ble, while applying pertinent corrections to take into account some of the major phenomena governing soaring flight. The differential constraints are then imposed as straightforward matrix operations, therefore all the restrictions remain linear and the order of the problem is reduced, because only the controls are treated as variables. That leads to a smaller and less complex NLP problem, allowing simpler numerical algorithms to be used for its solution.

2 PROBLEM FORMULATION

The simulations are based on the rigid body 3 DOF point–mass equations taking into account vertical wind effects. In summarized form, they are written as Eq. (1), being $\mathbf{u}$ the control vector, whose elements are the bank angle (ϕ) and lift coefficient (C_L). The state vector ($\mathbf{x}_{cart}$) is composed by the airspeed (V), flight path angle (γ), heading angle (ψ) and Cartesian coordinates (x_I, y_I, z_I). When the climb or scan modes are latched a polar coordinate system is used and the x_I and y_I state variables are then replaced by r and θ, which compose the alternate state vector ($\mathbf{x}_{pol}$). Aerodynamical, geometrical and inertial characteristics of a typical manned $15m$ span club class glider (Table 1) are employed in the simulations.

$$\begin{aligned} \mathbf{x}_{cart} &= \begin{bmatrix} V & \gamma & \psi & x_I & y_I & z_I \end{bmatrix}^T \\ \mathbf{x}_{pol} &= \begin{bmatrix} V & \gamma & \psi & r & \theta & z_I \end{bmatrix}^T \\ \mathbf{u} &= \begin{bmatrix} C_L & \phi \end{bmatrix}^T \\ \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\mathbf{u}) \end{aligned} \tag{1}$$

Table 1 Parameters of the aircraft model.

Mass $\mathbf{m}[\mathbf{kg}]$	Speed $\mathbf{V}[\mathbf{km/h}]$	Gld. Ratio $\left(\frac{\mathbf{L}}{\mathbf{D}}\right)_{\mathbf{max}}$	Min. Sink $\mathbf{V_{z,min}}[\mathbf{m/s}]$
340	$[65,220]$	32.7^1	-0.68^2

[1] At $81km/h$. [2] At $71km/h$.

The aircraft is subjected to the vertical motion of the air mass, whose velocity is assumed to be an arbitrary scalar field function of the Cartesian coordinates and time, i.e., $V_{w,z} = V_{w,z}(x_I, y_I, t)$. The controller incorporates a modified Gedeon [10] thermal model used for the predictions. Two different reference radii are introduced, $R_{t,x}$ and $R_{t,y}$, thus the thermal can assume an elliptical format with maximum intensity $V_{t,max}$, according to Eq. (2). Once those basic parameters are chosen, the estimated wind vertical speed ($V_{t,z}$) at any location (x_I, y_I) can be obtained. Note that the updraft centre position $(x_{t,0}, y_{t,0})$ and the rotation angle (η) are adjustable parameters too.

$$\begin{aligned} V_{t,z} &= V_{t,max} e^{-\chi^2}\left(1-\chi^2\right) \\ \chi &= \sqrt{\left(\frac{x_t}{R_{t,x}}\right)^2 + \left(\frac{y_t}{R_{t,y}}\right)^2} \\ x_t &= (x_I - x_{t,0})\,cos(\eta) + (y_I - y_{t,0})\,sin(\eta) \\ y_t &= -(x_I - x_{t,0})\,sin(\eta) + (y_I - y_{t,0})\,cos(\eta) \end{aligned} \tag{2}$$

No attempt to run a global thermal mapping algorithm, capable of simultaneously representing multiple updraft and downdraft zones in a wide area, is made herein, because the usefulness of such a global methodology (adopted by Lee, Longo and Kerrigan [8] and Liu et al. [9]) in fast time varying atmospheric scenarios is questionable. Instead, the present work proposes the online environment estimation to be performed locally through a predefined shape model (Eq. (2)), and only when the aircraft is manoeuvring for centring a thermal core. Furthermore, this local approach does not need a long record of measurements to be kept and allows the usage of shorter prediction horizons, since the controller is not "concerned" with the more distant zones.

If one considers an axisymmetric updraft and a steady atmosphere, then it is possible for a sailplane to remain in an equilibrium condition with constant velocity (V_{ss}) and bank angle (ϕ_{ss}), circling at a certain turn rate ($\dot{\psi}_{ss}$) and radius (r_{ss}) relative to the thermal core. Given a specified turn rate and velocity, Eq. (1) can be numerically solved for $C_{L,ss}$, ϕ_{ss}, r_{ss} and γ_{ss} using Scilab *fsolve* subroutine. The resulting sink rate (in calm atmosphere) is then computed as $V_{ss,z} = -V_{ss} sin(\gamma_{ss})$, while the inclusion of the updraft contribution ($V_{t,z}$, Eq. (2)) yields the final climb/sink rate:

$$\dot{z}_{I,ss} = V_{ss,z} + V_{t,z}(r_{ss}) \tag{3}$$

Two important aspects have to be emphasized: 1) the tighter the curve, the greater the sink rake, $V_{ss,z}$ (for a constant airspeed); 2) the closer to the thermal core the aircraft circles, the greater the vertical wind velocity it encounters. Those opposed effects play a major role in the proposed ATS algorithm and, in order to consider them, a database is computed offline. More specifically, the permanent flight conditions are calculated for a rectangular grid of $(V_{ss}, \dot{\psi}_{ss})$ pairs, yielding corresponding $V_{ss,z}$ values. After a bi–spline based regression procedure is applied, the aircraft sink rate in calm atmosphere is expressed as a function of airspeed and turn rate (Eq. (4)).

$$V_{ss,z} = V_{ss,z}(V_{ss}, \dot{\psi}_{ss}) \tag{4}$$

The soaring flight objectives can be transcribed to the following optimal control problem,

$$\begin{aligned} &\min_{\mathbf{u}(t)} J = \Omega\left(\mathbf{x}(t_f)\right) + \int_{t_0}^{t_f} \Gamma\left(\mathbf{x}(t), \mathbf{u}(t), t\right) dt \\ &\text{s.t., } \dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t), t\right), \quad t \in \left[t_0, t_f\right], \\ &\mathbf{u}_{lo} \leq \mathbf{u}(t) \leq \mathbf{u}_{up}, \quad t \in \left[t_0, t_f\right], \\ &\mathbf{x}_{lo} \leq \mathbf{x}(t) \leq \mathbf{x}_{up}, \quad t \in \left[t_0, t_f\right], \end{aligned} \tag{5}$$

whose interpretation is: suppose the aircraft at a given instant of time t_0 and corresponding state $\mathbf{x}(t_0)$. The objective is to obtain the control and state trajectories $(\mathbf{u}(t), \mathbf{x}(t))$, from t_0 to a fixed t_f, that minimize the cost J (composed by terminal (Ω) and stage (Γ) terms associated to energy gain or saving), while respecting differential restrictions and lower and upper bounds. The former are the equations of motion and the latter represent the aircraft inherent physical limitations or mission related constraints. A typical solution approach involves the discretization of the state and control trajectories along time so that they are parametrized by a certain number of scalar coefficients. The continuous time Eq. (5) is then rewritten as a NLP problem[1], whose solution is a key element of the MPC scheme.

At a certain instant of time (t_0) the current state vector $(\mathbf{x}(t_0))$ is estimated from sensor readings and the discretized version of Eq. (5) is solved within a predefined prediction horizon length $(t_f - t_0)$. The obtained controls $(\mathbf{u}(t))$ are then applied to the vehicle during a sampling time interval (ΔT), while the entire calculation process is repeated, so that at the next instant of time $(t = t_0 + \Delta T)$ a new optimal control sequence is available. This procedure is successively repeated using a constant prediction horizon length.

3 ALGORITHM DESCRIPTION

The proposed algorithm operates in three different modes, namely, climb, search and scan.

3.1 Climb Mode

The climb mode controller works in two levels. Initially, a higher level approach assumes the aircraft dynamics can be transcribed to the following set of equations:

$$\begin{aligned} \dot{V} &= a & \dot{x}_I &= V cos(\psi) \\ \dot{\psi} &= \omega & \dot{y}_I &= V sin(\psi) \\ \dot{z}_I &= V_{t,z}(x_I, y_I) + V_{ss,z}(V, \omega) \end{aligned} \tag{6}$$

They result from Eq. (1) after a few simplifications are imposed. Most notably, γ is assumed to remain small and its dynamics is neglected, while the acceleration (a) and angular velocity (ω) control variables are introduced ($\hat{\mathbf{u}} = [\hat{a} \quad \hat{\omega}]^T$). In fact, Eq. (6) describes the kinematics of the airplane motion to which two quasi–steady corrections are applied: the sink rate versus curve radius relationship (Eq. (4)) and the air mass vertical velocity contribution (Eq. (2)).

Once the sampling time (ΔT), the number of prediction (N) and control (M) steps are selected, the predictions are initially performed using only the $\dot{V}$ and $\dot{\psi}$ (linear) equations of the system (6), being $\hat{\mathbf{y}} = [\hat{V} \quad \hat{\psi}]^T$ the predicted output vector. The corresponding linear discrete equations, obtained by means of a zero–order–hold scheme [2], are successively applied, enabling one to evaluate

[1] See *http://plato.asu.edu/sub/nlores.html#general* for a comprehensive list of NLP solvers.

the effect of a sequence of control variations, $\Delta\hat{\mathbf{U}} = \left[\Delta\hat{\mathbf{u}}(k|k)^T \ \ldots \ \Delta\hat{\mathbf{u}}(k+M-1|k)^T\right]^T$, on the predicted outputs, $\hat{\mathbf{Y}} = \left[\hat{\mathbf{y}}(k+1|k)^T \ \ldots \ \hat{\mathbf{y}}(k+N|k)^T\right]^T$, at each time step (k). Equation (7) results, where $\mathbf{G}$ is a constant matrix and $\mathbf{f_{fr}}$ stands for the free response vector. After $\hat{\mathbf{Y}}$ is computed, the remaining equations of the system (6) can be integrated via a trapezoidal rule, leading to the aircraft position at each prediction step, $\hat{\mathbf{Y}}_{\mathbf{traj}} = [\hat{x}_I(k+1|k) \ \ \hat{y}_I(k+1|k) \ \ \hat{z}_I(k+1|k) \ \ \ldots \ \ \hat{y}_I(k+N|k) \ \ \hat{z}_I(k+N|k)]^T$.

$$\hat{\mathbf{Y}} = \mathbf{G}\Delta\hat{\mathbf{U}} + \mathbf{f_{fr}} \tag{7}$$

Note that, given a sequence of control actions $\left(\Delta\hat{\mathbf{U}}\right)$, the entire predicted output history $\left(\hat{\mathbf{Y}}\right.$ and $\left.\hat{\mathbf{Y}}_{\mathbf{traj}}\right)$ is obtained explicitly, via straightforward operations. Moreover, bounds imposed in V, a, ω, $\dot{a}$ and $\dot{\omega}$ yield linear constraints only, since Eq. (7) is itself linear. The problem is now ready to be posed in NLP format, according to Eq. (8).

$$\begin{aligned} &\min_{\Delta\hat{\mathbf{U}}} J = -\left(\frac{\hat{\mathbf{Y}}^2(2N-1)}{2g} - \hat{\mathbf{Y}}_{traj}(3N)\right) \\ &\text{where,} \\ &\hat{\mathbf{Y}} = \mathbf{G}\Delta\hat{\mathbf{U}} + \mathbf{f_{fr}}; \ \ \hat{\mathbf{Y}}_{\mathbf{traj}} = \mathbf{w}(\Delta T, \hat{\mathbf{Y}}, \Delta\hat{\mathbf{U}}) \\ &\text{s.t.,} \ \ \mathbf{S}\Delta\hat{\mathbf{U}} \leq \mathbf{b} \end{aligned} \tag{8}$$

The cost function J, written in terms of the total specific energy, denotes the objective of reaching the maximum energy state at the end of the prediction horizon. $\mathbf{S}$ and $\mathbf{b}$ are a constant matrix and vector respectively that express the state and control constraints in terms of $\Delta\hat{\mathbf{U}}$. The function $\mathbf{w}$ represents the explicit integration procedure applied to the $\dot{x}_I$, $\dot{y}_I$ and $\dot{z}_I$ equations of the system (6). Since the differential restrictions are contained into the prediction scheme, no extra variables nor nonlinear constraints are introduced and the size of the NLP problem is dictated by the control vector size only, i.e., the size of $\Delta\hat{\mathbf{U}}$ $(2M)$. For solving the NLP problem, the LINCOA optimization algorithm [11][2] is employed. It was selected because only the objective function values need to be supplied, which is coherent with the minimum input proposal of the present methodology.

The vehicle's data acquisition system is assumed to record the instantaneous vertical wind speed $\left(V'_{w,z}\right)$ and corresponding spatial coordinates every $\Delta T_{te} = 0.2s$. Perfect measurement and estimation are assumed and the data is stored in limited size $(N_{te} = 225)$ vectors using first-in-first-out queues. A nonlinear least squares problem for fitting the thermal model (Eq. (2)) to the acquired data is then written, being the Scilab *leastsq* subroutine employed to obtain the parameters $\beta_{te} = [V_{t,max} \ \ R_{t,x} \ \ R_{t,y} \ \ x_{t,0} \ \ y_{t,0} \ \ \eta]^T$. This numerical procedure is performed whenever the high level controller is called, providing an updated thermal model to be used on its predictions.

Airspeed and radius values, extracted from the optimal solution of the problem given by Eq. (8), compose the set–point vector $\mathbf{Y_r} = [V_r(k+1) \ \ r_r(k+1) \ \ \ldots \ \ V_r(k+N) \ \ r_r(k+N)]^T$ which is tracked by a low level MPC controller. Its internal model is a linearized version of Eq. (1) with the kinematics written in polar coordinates and disregarding atmospheric motion effects. Basic parameters must be set $(\Delta T, N, M)$ and, again, assuming a zero–order–hold strategy, a linear discrete time prediction equation similar to Eq. (7) is derived, but now the output vector is $\hat{\mathbf{y}} = \left[\hat{V} \ \ \hat{r}\right]^T$. Bound constraints on airspeed (V), controls (C_L, ϕ) and control rates $(\dot{C}_L, \dot{\phi})$ can also be put in linear format. The cost function (J) is written in discrete linear–quadratic (LQ) format for tracking $\mathbf{Y_r}$ throughout the prediction horizon while the control energy is also penalised. Thanks to the linear prediction scheme (Eq. (7)), the low level optimization problem can be transcribed to a standard quadratic–programming (QP) format which is function of $\Delta\hat{\mathbf{U}}$ only [2],

$$\begin{aligned} &\min_{\Delta\hat{\mathbf{U}}} J = \frac{1}{2}\Delta\hat{\mathbf{U}}^T\mathcal{H}\Delta\hat{\mathbf{U}} + \mathbf{c}^T\Delta\hat{\mathbf{U}} \\ &\text{s.t.,} \ \ \mathbf{S}\Delta\hat{\mathbf{U}} \leq \mathbf{b} \end{aligned} \tag{9}$$

where $\mathcal{H}$ and $\mathbf{c}$ are a real constant matrix and vector respectively. At the *k-th* time step the

[2] Code available at *http://mat.uc.pt/~zhang/software.html*

Scilab *qpsolve* subroutine is used to solve Eq. (9) and the control moves are extracted from the solution vector, i.e., $\Delta\mathbf{u}(k) = [\Delta\hat{\mathbf{U}}^*(1) \quad \Delta\hat{\mathbf{U}}^*(2)]^T$. After ΔT seconds, the plant response to those controls is read and used to derive the updated state vector, which is taken as reference for the linearization, prediction and solution processes at the following time step $(k+1)$. Hence, the internal model is constantly relinearized and realigned, increasing the fidelity of the prediction scheme. Note that an analogous realignment approach is also adopted by the high level controller.

3.2 Search and Scan Modes

The search mode is used for cruising and intends to explore the environment in order to find new thermal regions. Target waypoints are successively obtained as the solution of a nonlinear least squares problem set to maximize the distance from recently visited zones and to avoid the airspace boundaries. The reference airspeed is derived from the MacCready's speed to fly rule [12], that gives the autopilot the heuristic ability to speed up when flying into downdraft regions and slow down if updrafts are found. A constrained linear MPC scheme, analogous to the one employed by the low level climb controller (section 3.1), is responsible for tracking the constantly updated heading (ψ_r) and airspeed (V_r) set–points. Therefore, only a simple QP problem needs to be solved every time step. This is an alternative to the heavier search algorithms proposed by Lee, Longo and Kerrigan [8] and Liu et al. [9] which involve more complex nonlinear optimization problems, whose online solutions, depending on some of the related parameters, may become too costly [9].

Intended to run before the climb mode, the scan mode performs an initial survey of the potential updraft zone, acquiring data for the first thermal estimation procedure. It specifies an '8 shaped' reference path in order to explore the four quadrants. A system identical to the climb mode low level controller (section 3.1) tracks the predefined set–point vector composed by constant radius $(r_r = r_{scan} = 120m)$ and airspeed $(V_r = V_{scan} = 110km/h)$ values. Once the first turn is completed, a second turn is requested to the opposite direction.

3.3 Logic of Operation

Figure 1 illustrates the logic of operation of the proposed ATS algorithm.

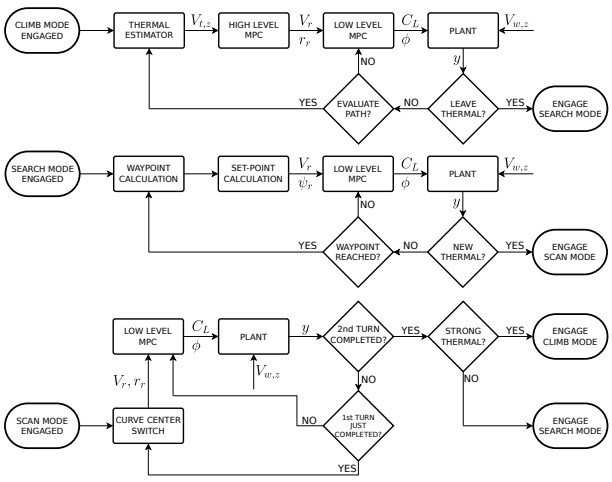

Fig. 1 Proposed ATS algorithm flowchart.

The following decision steps play a major role in dictating the system overall behaviour:

→ *Evaluate Path*: Every $\Delta T_{ref} = 10s$ a new thermal estimation followed by a run of the high level MPC controller is executed, otherwise the last calculated reference trajectory is kept as set–point to the low level MPC controller;

→ *Leave Thermal*: the thermal is abandoned if the height gain during the last $\Delta T_{clb} = 120s$ is smaller than $\Delta h_{clb} = 0m$;

→ *New Thermal*: If at a given time step $\dot{e}'(k)$ is greater than a predefined threshold $(\dot{e}_{src} = 0m/s)$ and, at the same time, it is less than its previous value, $\dot{e}'(k-1)$, the algorithm understands that a potentially usable updraft area was reached. Note that $\dot{e}'$ is the total specific energy rate (Eq. 10), derived from measurements;

$$\dot{e}' = \frac{V'\dot{V}'}{g} - \dot{z}'_I \tag{10}$$

→ *Strong Thermal*: a good thermal is assumed to exist if $\dot{e}'$ has remained above a given threshold $(\dot{e}_{scan} = 0.5m/s)$ for at least $\alpha_{scan}\Delta T_{scan}$

seconds, where ΔT_{scan} is the scan mode run time and $\alpha_{scan} = 0.2$.

4 SIMULATION RESULTS

Several numerical simulations have been executed in an integrated Scilab[3]/Fortran framework, but only the results related to a single baseline scenario are presented herein. The unpowered UAV is allowed to fly over a $9km^2$ quadrilateral area without altitude limitations, relying on the ATS algorithm to remain aloft. Randomly generated non–homogeneous clusters of thermals that can merge with each other and are surrounded by downdraft zones compose the environment. Each cluster is assumed to have a finite lifespan (from 10 to 20 minutes [1]) during which its intensity varies according to a sinusoidal law around the peak value. This explains why the medium at the end is completely different from the one found at the beginning of the run (Fig. 2 versus Fig. 3).

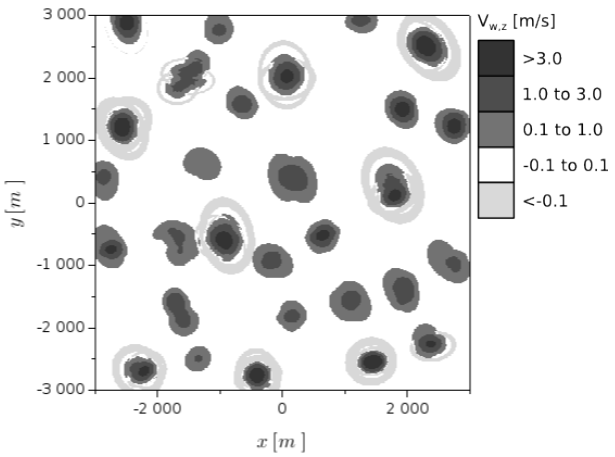

Fig. 2 Atmospheric environment at $t = 0s$.

Table 2 summarizes the main controller parameters. Some of them reflect airframe limitations (e.g, stall speed and control bounds), while others were selected according to practical considerations or tuned during the simulations. Note that the climb mode high level controller works with a $\Delta T = 2s$ sampling time, updating the reference signals every $\Delta T_{ref} = 10s$. The related prediction horizon of $50s$ ($N\Delta T$) has been found to

[3] Available at *https://www.scilab.org/en*

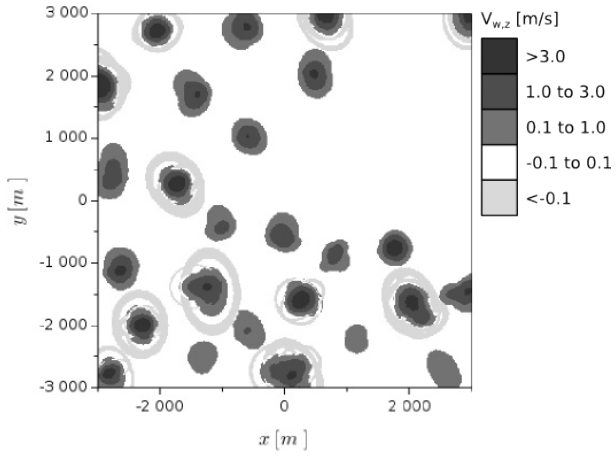

Fig. 3 Atmospheric environment at $t = 3600s$.

be a good trade–off between desired closed loop behaviour and computational effort. Regarding the lower level controllers, a $\Delta T = 0.2s$ sampling time and a shorter horizon ($4s$) have proven to yield accurate tracking of the set–points.

Table 2 Main controller parameters.

Low Level Controllers[1]			
$\Delta T[s]$	N	M	$C_{L,lo}$
0.2	20	5	0.1
$C_{L,up}$	$\dot{C}_{L,lo}[s^{-1}]$	$\dot{C}_{L,up}[s^{-1}]$	$\phi_{lo}[^\circ]$
1.4	-0.1	0.1	-70.0
$\phi_{up}[^\circ]$	$\dot{\phi}_{lo}[^\circ/s]$	$\dot{\phi}_{up}[^\circ/s]$	$V_{lo}[km/h]$
70.0	-9.0	9.0	67.0
High Level Controller[1]			
$\Delta T[s]$	N	M	$a_{lo}[m/s^2]$
2.0	25	25	-0.9
$a_{up}[m/s^2]$	$\dot{a}_{lo}[m/s^3]$	$\dot{a}_{up}[m/s^3]$	$\omega_{lo}[^\circ/s]$
0.9	-0.2	0.2	-30.0
$\omega_{up}[^\circ/s]$	$\dot{\omega}_{lo}[^\circ/s^2]$	$\dot{\omega}_{up}[^\circ/s^2]$	$V_{lo}[km/h]$
30.0	-3.0	3.0	75.0

[1]The lo and up subscripts stand for lower and upper bound respectively.

The simulation started with the aircraft over coordinates $(-2100m, -2100m)$ at a height of $1000m$. Figures 4 and 5 present the resultant trajectory in 2D and 3D format respectively, indicating that the aircraft has scanned a great part of

the airspace, thanks to the search mode strategy. It was able to keep safe altitudes during the entire flight (minimum of 887m, see Fig. 6).

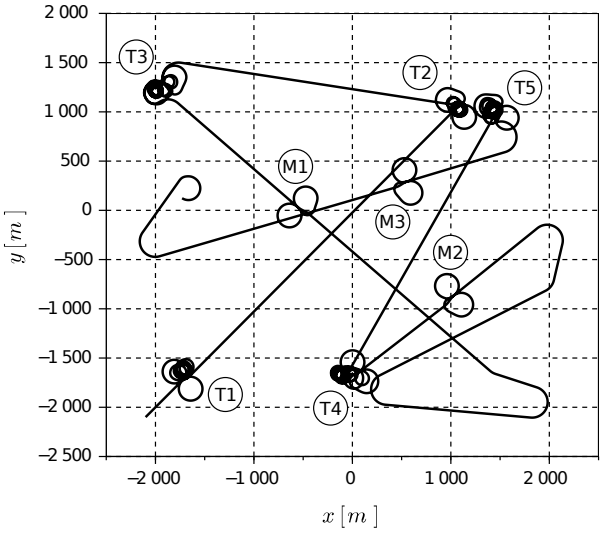

Fig. 4 Flight path after 1 hour – 2D view.

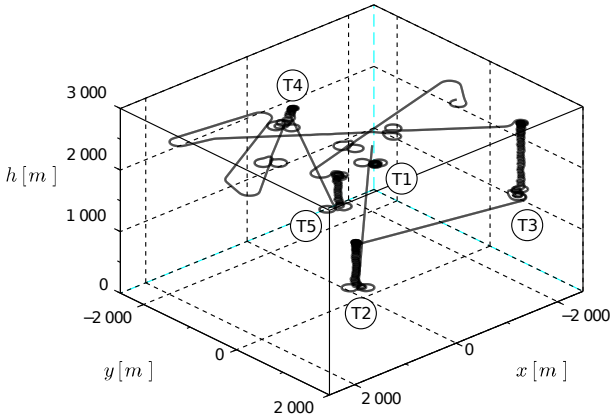

Fig. 5 Flight path after 1 hour – 3D view.

The operation mode was switched 22 times and the climb mode engaged in five occasions (labels $T1$-$T5$ in Fig. 4). In three opportunities (labels $M1$-$M3$), a updraft was found but the algorithm understood it was too weak and no circling was attempted. The flight terminated with a height gain of 1684m, while the mean rate of climb in thermals varied from 0.1 to 1.8m/s with the average turn radius ranging from 59 to 75m.

A detailed view of the flight path immediately before the climb mode latching as result of the encounter with the second thermal is presented

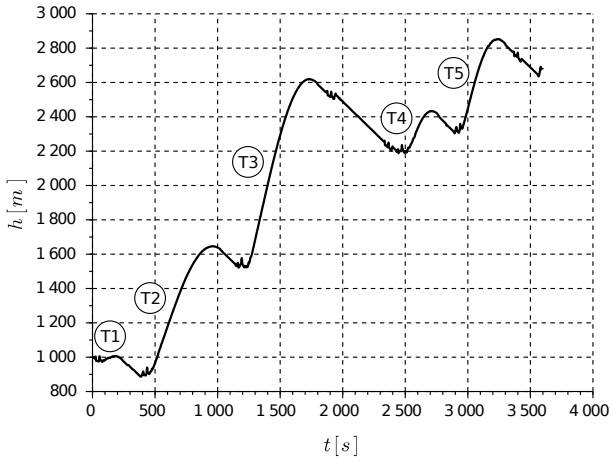

Fig. 6 Height variation.

in Fig. 7. After the '8 shaped' exploratory orbit imposed by the scan mode, it converged to a circular trajectory around the core, as indicated in Fig. 8. About 6.5 minutes after the climb mode engagement (the instant of time shown in Fig. 8), the updraft intensity was considerably lower, indicating that it would soon be abandoned.

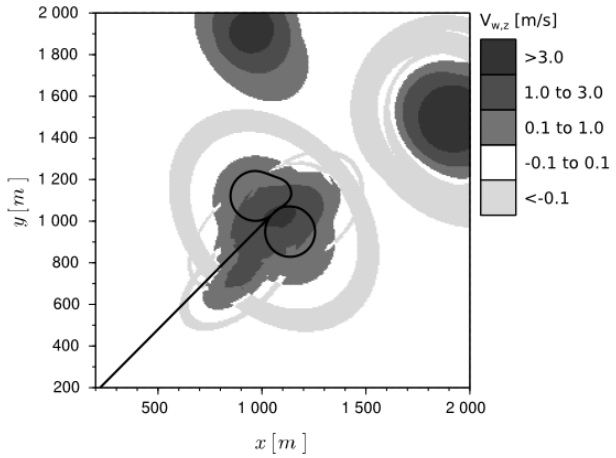

Fig. 7 Flight path 55s after the encounter with T2.

Figures 9 and 10 reveal how the airspeed and radius behaved during the exploration of the second thermal. The reference signals are also indicated, even though the two sets of curves are almost indistinguishable on the scale of the figures, except for a temporary low level controller tracking deviation around 435s, which corresponds to the instant when the curve direction change was commanded by the scan mode. At approximately $t = 460s$ the climb mode is latched and from that

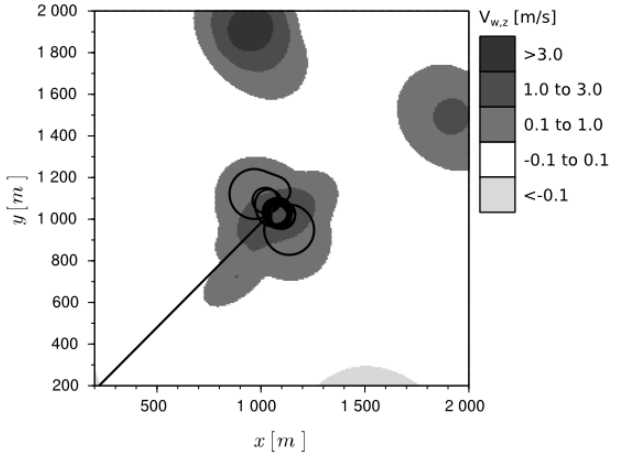

Fig. 8 Flight path 445s after the encounter with T2.

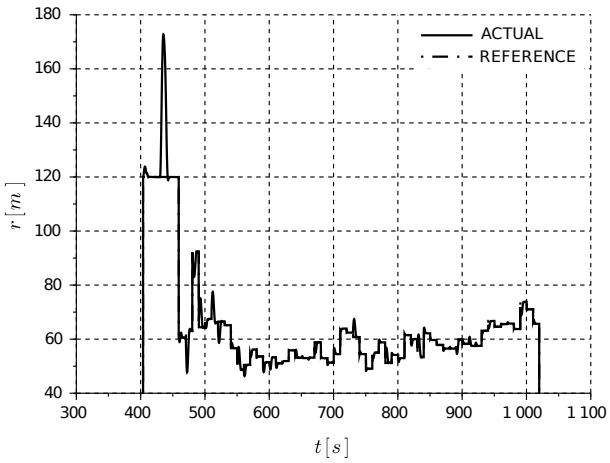

Fig. 10 Radius during the encounter with T2.

moment on the set–points were updated by the high level MPC controller every 10s, explaining the small steps seen in the responses.

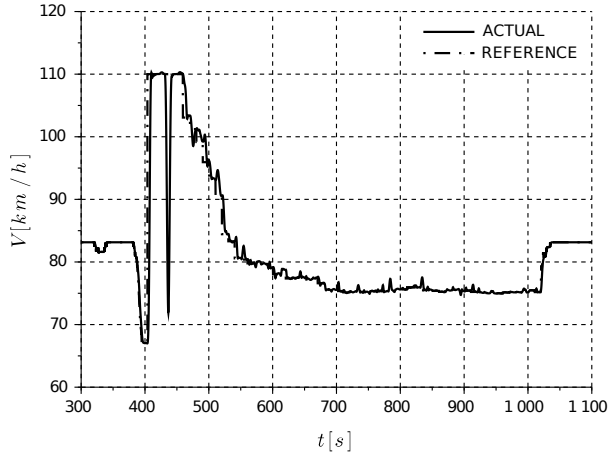

Fig. 9 Airspeed during the encounter with T2.

As soon as the climb mode was engaged, the ATS algorithm started to reduce the airspeed until a value slightly superior to the aircraft minimum sink velocity (see Table 1) was reached and maintained. At the same time the radius decreased to values between 45 and 65m approximately. Possibly flying faster at the beginning was preferred because reaching the updraft central region was more important than saving height. Moreover, the initial thermal environment evaluations tend to be less accurate, forcing the airplane to seek every new estimated core. Note that the airspeed lower bound (67km/h, Table 2) was activated at nearly 400s, but not crossed, illustrating the natural way constraints are treated by MPC schemes.

Table 3 presents a summary of the algorithm computational performance. It refers to the proposed test case run on a 1.80GHz quad–core CPU, 16 GB RAM personal computer executing Scilab version 5.5.0 under Ubuntu Linux 64 bits. The displayed time intervals encompass the entire set of operations needed for a single step.

Table 3 Required computation time per step.

	Full Step[1]	**Regular Step**
Average [s]	0.61	0.05
Maximum [s]	1.04	0.12

[1]Includes the thermal estimation and the high level MPC run (see figure 1).

The maximum recorded computation time when only the low level control actions are carried out is roughly half the sampling time ($\Delta T = 0.2s$), which is a promising result for a future real–time usage on flight hardware. However, possible complications arise when the full computations have to be performed (every $\Delta T_{ref} = 10s$ when the climb mode is latched), since calculation times greater than the elementary 0.2s sampling time were registered (up to 1.04s). Fortunately, there is a straightforward way to deal with that problem. It consists in keeping the previously computed reference path (given by the

high level MPC controller) until the current calculation is completed. In fact, to emulate that, the simulation whose results were presented throughout this section has been run with a fixed $1.2s$ lag in the high level controller response. This approach can also serve as a tool for handling infeasibility problems, since the previous reference can be used until the high level prediction horizon elapses. The NLP problem solution is responsible for about 60% of the calculation time per step. Although the LINCOA code worked successfully, it is not indicated for very large numbers of variables. Hence, the adoption of codes which are more suitable to high dimensional NLP problems could further improve the performance.

5 CONCLUSION

The performed 3 DOF numerical simulations are the first step to demonstrate the feasibility of the proposed ATS strategy. They indicate that energy from the convective phenomena which take place on the lower atmosphere could be successfully harvested without violating the aircraft performance limitations. Critical aspects of the methodology have been tested. Most notably, the simplified kinematic relations with quasi–steady sink corrections (the algorithm's core) have proven to yield sufficiently accurate predictions. Its capability of producing simpler and smaller nonlinear optimization problems has also been demonstrated. For instance, the high level controller used in the test case of section 4 requires a NLP problem with only 50 variables and 150 linear inequalities to be solved at each step, covering a $50s$ prediction horizon. If the differential constraints had to be explicitly imposed, for example by directly using the 3 DOF nonlinear EOM for prediction, the number of variables would be much larger (by a factor of four approximately) and extra nonlinear constraints would have to be considered. Hence, one concludes that the present approach has the potential to facilitate the implementation of less complex, faster and more efficient codes. Indeed, the obtained computation times and adopted implementation strategies point to viable real–time operation.

6 Contact Author Email Address

For contacting the authors:
mailto: gpogor@gmail.com

References

[1] World Meteorological Organization. Weather forecasting for soaring flight. Technical Note No. 203, WMO-No. 1038, 2009.

[2] Maciejowsky J M. *Predictive Control With Constraints*. Pearson Education – Prentice Hall, Harlow, England, 2002.

[3] Allen M J. Guidance and control of an autonomous soaring UAV. NASA/TM-2007-214611, 2007.

[4] Edwards D J and Silverberg L M. Autonomous soaring: the Montague Cross–Country Challenge. *Journal of Aircraft*, Vol. 47, No. 5, pp 1763-1769, 2010.

[5] Andersson K, Kaminer I, Dobrokhodov V and Cichella V. Thermal centering control for autonomous soaring; stability analysis and flight test results. *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 3, pp 963-975, 2012.

[6] Edwards D J. Autonomous Locator of Thermals (ALOFT) autonomous soaring algorithm. NRL/FR/5712–15-10, 272, 2015.

[7] Kahn A D. Atmospheric thermal location estimation. *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 9, pp 2363-2369, 2017.

[8] Lee D, Longo S and Kerrigan E C. Predictive control for soaring of unpowered autonomous UAVs. *4th IFAC NMPC*, Noordwijkerhout, NL, 2012.

[9] Liu Y, Schijndel J V, Longo S and Kerrigan E C. UAV energy extraction with incomplete atmospheric data using MPC. *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 51, No. 2, pp 1203-1215, 2015.

[10] Gedeon J. Dynamic analysis of dolphin–style thermal cross–country flight. *Technical Soaring*, Vol. III, No. 1, pp 9-19, 1973.

[11] Powell M J D. On fast trust region methods for quadratic models with linear constraints. *Math. Prog. Comp.*, Vol. 7, No. 3, pp 237-267, 2015.

[12] Reichmann H. *Cross Country Soaring*. Soaring Society of America, Inc., Hobbs, NM, 1993.

Copyright Statement