

UAV Energy Extraction with Incomplete Atmospheric Data Using MPC

Yuyi Liu, Jos van Schijndel, Stefano Longo and Eric C. Kerrigan

Abstract—A Nonlinear Model Predictive Control strategy is presented for UAV trajectory determination. The objective is to find optimal paths in the atmosphere by maximizing the UAV's energy (kinetic and potential) over a finite but receding horizon. The main assumption is that the updraft distribution is unknown, creating a realistic situation. The updrafts are only estimated online using standard on-board inertial sensors. Real-time implementation of the algorithm is shown to be possible in principle.

Index Terms—Trajectory optimization, UAV, NMPC, Thermal updraft estimation, Thermal energy extraction.

I. NOMENCLATURE

x, y, h	components of horizontal position and height of aircraft's mass center in an inertial frame
γ	flight path angle of aircraft
ψ	heading angle of aircraft
V	aircraft airspeed
α	angle of attack
ϕ	bank angle of aircraft
m	mass UAV
S	wing area UAV
ρ	air density
g	gravitational acceleration
W_x	value of horizontal wind velocity in x direction
W_y	value of horizontal wind velocity in y direction
W_h	value of vertical thermal updraft velocity
h_e	weight-specific energy height
F_L	aerodynamic lift of aircraft
C_L	coefficient of aerodynamic lift
D	aerodynamic drag of aircraft
C_D	coefficient of aerodynamic drag
J	cost function
t_0	initial time control horizon
t_f	terminal time control horizon
$\mathbf{x}(t)$	state vector
$\mathbf{u}(t)$	control vector
E	terminal cost
L_{TSP}	stage cost TSP method
L_{PVH}	stage cost PVH method

β_c	scaling factor stage/terminal cost
A	cost at discretization point
d	distance between locations (PVH)
$\mathcal{P}$	set with penalized locations (PVH)
$\mathcal{H}$	set with predicted locations
Δt_{th}	receding horizon interval
t_{hor}	prediction horizon length
V_{up}	sufficient updraft strength
h_{safe}	safe altitude for heuristic search TSP
t_{search}	heuristic search time TSP
λ	spread PVH function
t_{del}	PVH delay before storing locations
Δt_{PVH}	interval storing PVH locations
n_{ini}	amount a priori neurons (GRNN, PVH)
n_w	weight a priori neurons (GRNN, PVH)
Δt_{GRNN}	interval adding neurons to GRNN
σ	spread GRNN

II. INTRODUCTION

ON the promise of higher endurance for Unmanned Aerial Vehicles (UAVs), techniques to extract energy from the atmosphere are highly beneficial to help loitering of UAVs. One form of energy, coming from thermals, is a largely unexploited resource in the atmosphere and this form of energy is currently only utilized, in an effective way, by soaring birds and glider pilots [1], [2]. There is a great amount of energy that is waiting to be harvested. However, an automatic and, most importantly, *optimal* way to find and use this form of energy without having any prior knowledge of the atmosphere has yet to be developed. Autonomous optimal soaring is therefore an area of keen interest.

The main advantage of using an optimal energy search method is that the endurance of a UAV can be increased. This can reduce fuel consumption or increase battery life. Therefore, the UAVs can have better performance for a diverse set of purposes, such as area surveillance, rescue searching, border patrol and commercial advertising [3]. In the future, they might also be used to harvest the energy from the atmosphere (both from updrafts and horizontal winds) to generate electric power to be injected directly into the electrical grid [4].

Previous work on UAV energy harvesting by autonomous soaring around thermals has been done using classical control methods, e.g. using a PI controller [5]. A variety of search approaches have been employed for UAV control, such as navigating through the environment by using a heuristic search called the A* algorithm [6]. Other research in this field includes optimizing an energy-efficient path through a thermal in 2D [7] and using an adaptive controller [8]. Research has also been conducted on the estimation of atmospheric

Y. Liu is with the Chair of Cognitive Systems, Department of Computer Science, University of Tübingen, Sand 1, 72076, Tübingen, Germany
yuyi.liu@uni-tuebingen.de

J. v. Schijndel is with the Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, 5612 AZ, The Netherlands
j.v.schijndel@student.tue.nl

S. Longo is with the Department of Automotive Engineering, Cranfield University, Cranfield, MK43 0AL, UK
s.longo@cranfield.ac.uk

E. C. Kerrigan is with the Department of Electrical and Electronic Engineering and with the Department of Aeronautics, Imperial College London, London, SW7 2AZ, UK
e.kerrigan@imperial.ac.uk

energy. A set of algorithms for thermal data measurement were flight tested and reported in [9]. The surrounding parameters, such as updraft velocity, radius and position were estimated and reported in [10]. Other studies on soaring UAV control focus on the shear effects of the horizontal wind for long-endurance flights [11]–[13]. The majority of previous research has revolved around thermal centering methods [5], [14], often with a pre-defined thermal model and an estimated updraft distribution over the flight region of the UAV [7]. Thermal centering control was recently flight tested and reported in [15]. Various algorithms, such as Kalman filtering [16] or particle filtering [17] are utilized for the wind speed estimation. Other research on UAV energy extraction also combines a horizontal wind field with updrafts [18], [19]. However, the updraft models in the majority of previous works are static (i.e. with no change throughout the duration of the UAV's flight), hence control for energy harvesting can be unrealistic. Another difference is that no special equipment (i.e. forward sensing infrared thermal cameras [20]) is assumed to be integrated on the UAV; we assume that only Inertial Measurement Unit (IMU) readings are used to measure the UAV position and airspeed is measured via standard on-board sensors.

We propose a Nonlinear Model Predictive Control (NMPC) method to find high energy locations in the atmosphere. The NMPC problem is set up such that the controller continuously computes the trajectory that maximizes the total energy gained by the UAV. For the optimal control problem defined in an NMPC framework, a 3 degrees-of-freedom (DOF) UAV model is used. We use a 3DOF rather than a 6DOF model because 6DOF models require more computational power and have only been validated by simple environmental simulations [5]. Two different exploration strategies are used to search for energy (updrafts) in an environment that has a thermal distribution with a negative average magnitude, where a UAV with a pure random flight will lose energy. First, a heuristic search is proposed, which uses NMPC to follow a trajectory that is generated in conjunction with the solution of a Traveling Salesman Problem (TSP) [21]. The second search method uses NMPC to steer the UAV to high energy locations, while penalizing the UAV for traveling towards points it has already visited; we call this the Penalized Visited History (PVH) method. To obtain a good estimate of the updrafts distribution required by the NMPC scheme, a Generalized Regression Neural Network (GRNN) is used.

The novelty of this paper is the application of NMPC for the UAV to solve the energy harvesting task with no support of vision-based equipment. Only IMU and airspeed sensors are used. We demonstrate the feasibility of the combination of NMPC with online updraft estimation by simulation. The computation time required by our proposed control system is shown to be small enough to be applied in real-time, which is promising for further applications. Additionally, we show the controller's inherent robustness via simulations in a dynamic (time-varying) atmospheric scenario.

The outline of this paper is as follows. In Section III, the problem setup of the NMPC strategy is presented, including the 3DOF mathematical model of the UAV and the constraints used within the NMPC algorithm. Section IV describes the

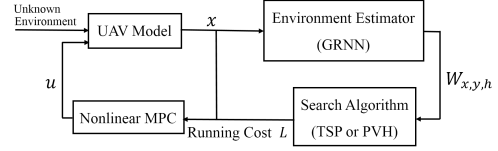


Fig. 1. NMPC framework for soaring UAVs.

cost function definitions and the search strategies employed for the optimal control of the UAV. Section V explains the online atmosphere estimation technique that is used. Section VI presents a simulation study, which introduces the test environment and a short comparison of the computation time of the 3DOF versus 6DOF model in the NMPC algorithm. The heuristic search based on the TSP and a search based on the PVH method, both combined with the online atmosphere estimator, are then presented and compared. Finally, the cross wind influence is introduced for a more realistic and challenging simulation. We present conclusions in Section VII.

III. PROBLEM DESCRIPTION

The main objective of our autonomous problem can be described as: *find an optimal path towards high energy locations such that the total (kinetic and potential) energy gained by the UAV is maximized, while keeping the computation time for the solution of the resulting optimal control problem small to enable real-time implementation.*

This objective is achieved by the following procedure:

- 1) The UAV obtains the current location and velocity via an IMU and airspeed sensor, and computes the updraft velocity on that location (based on an internal model);
- 2) The atmosphere estimator generates the ‘updraft distribution picture’ from the collected updraft velocity data;
- 3) The embedded computer calculates an optimal trajectory by solving the nonlinear, constrained optimal control problem using the UAV dynamical model interconnected with the atmosphere estimator model;
- 4) As in receding horizon control, the first control action is applied to the UAV for the duration of a time step and when this time step is finished, the process is repeated from step 1.

A complete NMPC framework is displayed in the Figure 1, from which readers can intuitively understand how our feedback system works that contains NMPC, GRNN and exploration algorithms.

A. Nonlinear Model Predictive Control

NMPC is an advanced feedback control method that predicts the change in the dependent variables of the modeled system. The controller minimizes a cost function by computing a sequence of optimal control inputs and the resulting optimal trajectory for a UAV model with constraints over a finite horizon. Only the first control input of the sequence is then applied to the UAV. At the next time step, the calculations are repeated starting from the current state, obtaining new control inputs as well as a new predicted trajectory.

Another approach that can be found in the literature is Reinforcement Learning (RL), which is often compared with Dynamic Programming (DP) strategies [22], [23]. NMPC and

RL are similar in the sense that they both provide an optimal input to the system, considering the reward they can obtain (given by a cost function). In our case, the reward is defined as in (1) below and is written as a minimization problem. In a sense, NMPC can be seen as an online DP-like algorithm, which can be seen as a model-based RL algorithm. The 3DOF-model-based NMPC is able to accurately describe the trajectory of the UAV, which can guarantee that the UAV's flight will satisfy the dynamic constraints to be described later.

The equation below is employed as the framework in which the UAV equations of motion, associated constraints and the atmospheric model, are incorporated. The optimal control problem can be stated as

$$\min_{\mathbf{u}(\cdot)} J(\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_0, t_f), \quad (1a)$$

subject to $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$ and

$$\mathbf{x}_L \leq \mathbf{x}(t) \leq \mathbf{x}_U, \quad \mathbf{u}_L \leq \mathbf{u}(t) \leq \mathbf{u}_U, \quad (1b)$$

for all $t \in [t_0, t_f]$, a given t_0 , t_f , and a given initial state $\mathbf{x}(t_0)$, where $J(\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_0, t_f)$ represents the cost function, $f(\mathbf{x}(t), \mathbf{u}(t))$ is the UAV dynamical model, and $\mathbf{x}(t)$ and $\mathbf{u}(t)$ are the state and control vector. Non-available states can be estimated using non-linear estimation techniques (e.g. [24]). We can divide the cost function in (1a) into two functions

$$J(\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_0, t_f) = \int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t)) dt + \beta E(\mathbf{x}(t_f)), \quad (2)$$

where $L(\cdot, \cdot)$ is the running cost, $E(\cdot)$ is the terminal cost, and β [-] is a weight to scale between the importance of the two. The optimal inputs calculated for a given t_0 , t_f , are then defined by

$$\mathbf{u}^*(\cdot) = \arg \min_{\mathbf{u}(\cdot)} J(\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_0, t_f), \quad (3)$$

and the predicted states $\mathbf{x}(t)$ can be calculated by

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \int_{t_0}^t f(\mathbf{x}(s), \mathbf{u}^*(s)) ds, \quad \forall t \in [t_0, t_f]. \quad (4)$$

Note that in the case study, $t_0 = \tau$ [s] and $t_f = \tau + t_{hor}$ [s] are chosen when finding the solution to (1), where τ [s] is the current time, and t_{hor} [s] is the prediction horizon length.

B. Unpowered Aircraft Model

With the goal of achieving real-time control, a rigid 3DOF UAV model is employed for the optimal control problem. The 3DOF point-mass unpowered flight model consists of six nonlinear ordinary differential equations (ODEs), including three navigation and three dynamic equations, represented in the inertial coordinate system, with certain rate constraints included to ensure that its performance is realistic. The scalar representation of a UAV model under both vertical atmospheric

TABLE I
UAV (DG-100 GLIDER) MODEL PARAMETERS.

Parameter [unit]	Value
Mass, m [kg]	300
Wing area, S [m ²]	11
Air density, ρ [kg·m ⁻³]	1.225
Gravitational acceleration, g [m·s ⁻²]	9.81

and horizontal cross wind influence is given by

$$\dot{x} = V \cos \gamma \cos \psi + W_x, \quad (5a)$$

$$\dot{y} = V \cos \gamma \sin \psi + W_y, \quad (5b)$$

$$\dot{h} = V \sin \gamma + W_h, \quad (5c)$$

$$\dot{\gamma} = \frac{1}{mV \cos \gamma} (L \cos \phi - mg \cos \gamma + m\dot{W}_x \sin \gamma \cos \psi + m\dot{W}_y \sin \gamma \sin \psi + m\dot{W}_h \cos \gamma), \quad (5d)$$

$$\dot{\psi} = \frac{1}{mV \cos \gamma} (L \sin \phi - m\dot{W}_x \sin \psi - m\dot{W}_y \cos \psi), \quad (5e)$$

$$\dot{V} = \frac{1}{m} (-D - mg \sin \gamma - m\dot{W}_x \cos \gamma \cos \psi - m\dot{W}_x \cos \gamma \sin \psi + m\dot{W}_h \sin \gamma), \quad (5f)$$

where

$$\begin{aligned} \dot{W}_x &= \frac{dW_x}{dh} \dot{h}, & \dot{W}_y &= \frac{dW_y}{dh} \dot{h}, \\ \dot{W}_h &= \frac{dW_h}{dx} \dot{x} + \frac{dW_h}{dy} \dot{y}, \\ F_L &\triangleq \frac{1}{2} C_L \rho S V^2, & C_L &\triangleq 0.7 \cdot 2\pi\alpha, \\ D &\triangleq \frac{1}{2} C_D \rho S V^2, & C_D &\triangleq 0.015 + 0.02 C_L^2. \end{aligned} \quad (6)$$

The factor of 0.7 on the coefficient of lift is used to discount for the three-dimensional flow effects on a finite-span wing. The zero-lift drag coefficient is assumed to be 0.015 (a typical value is between 0.015 and 0.018). The updraft velocity W_h needs to be estimated. For simplicity, the horizontal wind speed (W_x and W_y) is assumed non-zero only in the x direction ($W_y \equiv 0$, $\dot{W}_y \equiv 0$). The aircraft pitch angle is replaced by the flight path angle and the angle of attack (AOA). We assume that the model has a symmetric structure along the aircraft principal axis, hence the aerodynamic sideslip force is neglected here. Other parameters, corresponding to those of a DG-100 glider, are given in Table I. All other variables are described in the nomenclature section.

In this 3DOF model, the control input vector is defined to be the AOA rate and bank angle rate, i.e.

$$\mathbf{u} \triangleq (\dot{\alpha}, \dot{\phi}). \quad (7)$$

The state vector is defined as the six states in the equations of motion, together with the AOA and bank angle

$$\mathbf{x} \triangleq (x, y, h, \gamma, \psi, V, \alpha, \phi). \quad (8)$$

One could have selected the AOA and bank angle as the inputs. The AOA rate and bank angle rate are selected as inputs instead so that we are able to put constraints on them in our optimization problem, as discussed next.

TABLE II
STATE CONSTRAINTS OF 3DOF UAV MODEL.

States [unit]	Lower bound	Upper bound
x [km]	x_{min}	x_{max}
y [km]	y_{min}	y_{max}
h [km]	0.1	∞
γ [°]	-30	30
ψ [°]	$-\infty$	∞
V [m/s]	15	70
α [°]	0	10
ϕ [°]	-45	45

TABLE III
INPUT CONSTRAINTS OF 3DOF UAV MODEL.

Inputs [unit]	Lower bound	Upper bound
$\dot{\alpha}$ [°/s]	-10	10
$\dot{\phi}$ [°/s]	-30	30

C. Constraints

Table II lists the constraints of the aircraft states employed in the optimal controller, where the bounds of $-\infty$ and ∞ indicate that there is no lower or upper bound, respectively. The area that can be explored is constrained and will be discussed in the simulation study. The angles and the angle rates of the UAV, i.e. AOA, AOA rate, bank angle, bank angle rate and heading angle, are constrained such that the 3DOF model given in (5) is valid [25], [26]. These constraints are also set to give the plane a smooth flight and to prevent undesirable physical phenomena, such as stall, to occur. The height h has a constrained minimum equal to 100 meters. This prevents that the UAV will soar too close near the ground.

Rate constraints are also imposed on input vectors as given in Table III, in order to ensure that the UAV stays within the envelope of a realistic flight.

Note that all constraints shown here are hard constraints. In a non-deterministic setting, e.g. in the real world, there can be a mismatch between the prediction and the actual flight path of the UAV. However, in our case, the UAV model and the controller's UAV model are equal, which means there will be no mismatch. For implementation on a real UAV, one could think of rewriting some of the constraints as soft constraints.

IV. OPTIMAL CONTROL METHODS AND COST FUNCTIONS

We now consider two exploration methods to find high energy locations in the atmosphere. Both methods minimize the cost function previously stated in (1) and (2). We first discuss the cost function that is used when no search method is active, followed by the search method based on the Traveling Salesman Problem (TSP). We then discuss the Penalize Visited History (PVH) method, which penalizes the UAV for traveling to previously visited locations in the atmosphere.

A. UAV Not Searching

When the UAV is not searching, for example after it found a sufficiently strong thermal, the controller aims to maximize the total energy of the UAV at the end of the prediction horizon. Hence, the optimal controller is designed to maximize the amount of atmospheric energy extracted at the end of the

horizon. The stage cost is in this case set equal to zero. The optimal problem is defined as in (1) and (2), with

$$L(\mathbf{x}(t), \mathbf{u}(t)) \triangleq 0, \quad (9a)$$

$$E(\mathbf{x}(t_f)) \triangleq - \left(h(t_f) + \frac{V^2(t_f)}{2g} \right), \quad (9b)$$

and $t_f = \tau + t_{hor}$ [s], where $t_0 = \tau$ [s] is the current (simulation) time, t_{hor} [s] is the prediction horizon length, and $E(\mathbf{x}(t_f))$ [m] is the weight-specific energy height, which represents the summation of flight potential and kinetic energy over its weight [27], with the minus sign ensuring the problem is defined as a minimization problem.

B. UAV with Search Method Active

When the UAV is searching, both methods use the stage cost to either reward or penalize the UAV for going towards certain points in the atmosphere. This means that during search, a non-zero stage cost replaces (9a). For explanatory reasons, the stage cost during search is shown in its discretized form using the trapezoidal rule as

$$\int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t)) dt \approx \sum \Delta t_{rh} A(a), \quad (10)$$

for $a \in \mathcal{H} \cup 0.5 \{\mathcal{H}_1, \mathcal{H}_{N_H}\}$, where N_H is the cardinality of $\mathcal{H}$, and

$$\mathcal{H} \triangleq \{x(i\Delta t_{rh}), y(i\Delta t_{rh}) \mid i \in \mathbb{N}_0, t_0 \leq i\Delta t_{rh} \leq t_f\}, \quad (11)$$

with $t_0 = \tau$ [s] and $t_f = \tau + t_{hor}$ [s], where Δt_{rh} [s] is the receding horizon interval, $A(a)$ is a penalty or reward, which will be discussed later, $x(i\Delta t_{rh})$ [m] and $y(i\Delta t_{rh})$ [m] are the predicted absolute locations of the UAV on the map in x and y direction, respectively, τ [s] is the current (simulation) time, and t_{hor} [s] is the prediction horizon length. Note that we always, for both exploration methods, in both the case of searching or not searching, use (9b) to try to lose the least amount of energy possible during flight.

In the following two subsections, we explain in detail how the exploration algorithms work.

1) *Traveling Salesman Problem Method*: The idea of the TSP method is that the UAV (a salesman) has to visit all different areas (cities) in the atmosphere (on a map) with the shortest possible path [21]. The heuristic search algorithm is determined by the following conditions:

- 1) Only when the UAV is higher than a 'safe altitude' does the UAV start to consider the search of updrafts (otherwise, the UAV tries to gain altitude first);
- 2) A reference direction is generated for the UAV to fly for the given 'heuristic search time'.

As listed in Table IV, the heuristic search parameters are selected to stop searching when the updraft found is strong enough for the UAV to extract a sufficient amount of energy. This number will depend on the topology of the atmosphere considered. Table IV also displays the NMPC setting parameters, such as prediction control interval, prediction horizon length and time discretization density.

The TSP method makes use of an adaptive grid, which varies the size of areas on the map between smaller or larger

TABLE IV
HEURISTIC SEARCH MODEL PARAMETERS.

Description	Parameter [unit]	Value
Receding horizon interval	Δt_{rh} [s]	2
Prediction horizon length	t_{hor} [s]	120
Discretization per prediction horizon	N [-]	361
Safe altitude	h_{safe} [m]	500
Reasonable updraft strength	V_{up} [m/s]	2.5
Heuristic search time	t_{search} [s]	60

squares, depending on the measurements of the atmosphere. The grid space is coarse when the UAV discovers that the updraft velocity did not reduce or increase much over the past period of time (updraft gradient is low or negative). On the contrary, the adaptive grid will use fine grid size elements when the thermal gradient was strong and positive during the past period of time. This way, the time the UAV spends near a downdraft is reduced, while the search time around updrafts is increased. Compared to a random grid procedure, a systematic method for searching is more robust, as can be found in [25]. We refer the reader to [25] for more details on the TSP method.

When the TSP method is searching, a reference path is generated and the stage cost becomes as (10), (11), with

$$A(a) \triangleq \sqrt{(a - x_r)^2 + (a - y_r)^2}, \quad (12)$$

where x_r [m] and y_r [m] are absolute reference locations on the map generated via the TSP method. This works as follows: Each area has a center location, which will be connected by fitting the shortest possible path through them by using the TSP algorithm [25]. When the search starts, the values for x_r [m] and y_r [m] are determined as the center location of the area closest to the UAV. When the UAV nears this center location, the next area in the trajectory determined by TSP method will become the new travel target. In other words, the values for x_r [m] and y_r [m] change to the center location of the next area, which are connected through the solution to the TSP. Note that the terminal cost is always equal to (9b).

2) *Penalized Visited History Method*: The PVH approach will cause the UAV to search for the maximum amount of energy in the atmosphere while penalizing the UAV for traveling to previously visited locations. The stage cost will be used in order to cause the UAV to stay away from certain locations. The PVH method will stop when a sufficiently strong updraft is found, as was the case with the TSP method.

The main idea of the PVH method is to choose a cost function so that the UAV will steer itself towards the desired high energy location(s). Multiple stage costs and terminal costs can be combined and normalized. In this way, the resulting optimal control problem will result in a balanced decision for the trajectory to be flown. However, in this paper, we will only concentrate on minimizing energy and avoiding locations visited in the past. The terminal cost (9b) is used to fulfill the maximizing energy requirement at the end of the horizon.

When the PVH method is searching, we want to make sure that the stage cost is high when the predicted states are placed close to locations visited in the past. The discrete set that holds the locations visited in the past that are penalized is equal to

$$\mathcal{P} \triangleq \{x(i\Delta t_{PVH}), y(i\Delta t_{PVH}) \mid i \in \mathbb{N}_0, 0 \leq i\Delta t_{PVH} \leq \tau - t_{del}\}, \quad (13)$$

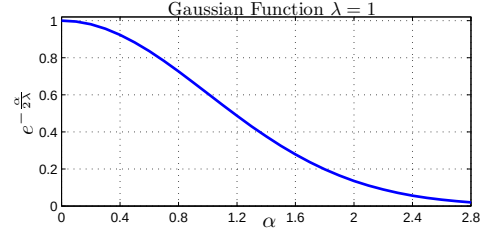


Fig. 2. A Gaussian-type function is used for the stage cost function.

where $x(i\Delta t_{PVH})$ [m] and $y(i\Delta t_{PVH})$ [m] are the absolute location of the UAV on the map in x and y direction, respectively, τ [s] is the current (simulation) time, t_{del} [s] is a delay before a location is considered to be penalized, and Δt_{PVH} [s] is an interval that decides how often locations are added to $\mathcal{P}$. Note that because $i\Delta t_{PVH} \leq \tau - t_{del}$, all $x(i\Delta t_{PVH})$ and $y(i\Delta t_{PVH})$ in the set $\mathcal{P}$ are in the past.

We use a Gaussian function to penalize the distance between the predicted locations in set $\mathcal{H}$ and the visited locations in set $\mathcal{P}$. A Gaussian function is a smooth curve and is shown in Figure 2. The stage cost is as in (10), (11), with

$$A(a) \triangleq \sum_{b \in \mathcal{D}(a)} e^{-\frac{d(a,b)}{2\lambda}}, \quad (14)$$

$$\mathcal{D}(a) \triangleq \{d(a,b) \mid b \in \mathcal{P}\}, \quad (15)$$

$$d(a,b) \triangleq \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}, \quad (16)$$

where $A(a)$ is a penalty for when locations in set $\mathcal{H}$ are close to locations in the set $\mathcal{P}$, λ [m] can increase or decrease the width of the function (14), $\mathcal{D}(a)$ is a set holding the distance of a location in the set $\mathcal{H}$ to all points in the set $\mathcal{P}$, and $d(a,b)$ [m] is the distance between a location a [m] and b [m].

The UAV's flight is a continuous phenomenon, but its path is stored in the discretized set $\mathcal{P}$. One has to specify the interval Δt_{PVH} , which decides how fast points are considered to be within this set. If the interval is too large, the UAV may be able to travel between these locations. If the interval is small, the computation time can become prohibitively high, meaning that the time to solve the optimization problem exceeds the sample time of the receding horizon strategy. A balanced choice has to be made here.

The past flight path of the UAV is, however, not directly penalized. This will cause the UAV to have t_{del} seconds to explore an area of updrafts more carefully before it is pushed away from its location. A downdraft will not be explored, since this will cause the UAV to lose energy. The UAV will search until the UAV finds the sufficiently strong updraft.

A problem that can arise in the case of a time-varying atmosphere is that a downdraft that was previously visited changes into an updraft. The UAV will not be able to find this updraft because the UAV is penalized for traveling towards this area. Under the assumption that the atmosphere changes slowly in time and the (simulation) time of the UAV is short, this causes no problems. In future research, one could think of reducing the penalty on areas visited far in the past (exponentially) against the flight time.

Since the NMPC strategy proposed here is nonlinear, the UAV could get stuck in a local optimum, i.e. an updraft that does not provide enough energy. Therefore, a UAV that uses

TABLE V
PARAMETERS FOR PVH METHOD IMPLEMENTATION.

Description	Parameter [unit]	Value
Receding horizon interval	Δt_{th} [s]	2
Length prediction horizon	t_{hor} [s]	30
Discretization points prediction horizon	N [-]	61
Spread PVH function	λ [m]	0.125
PVH delay before storing locations	t_{del} [s]	70
Interval storing PVH locations	Δt_{PVH} [s]	10
Weight terminal cost	β_c [-]	1
Reasonable updraft strength	V_{up} [m/s]	2.5

the PVH method will not only keep searching new areas by staying away from visited locations, but also get out of local optima that do not provide enough energy.

If the UAV remains at the same position for a longer time, multiple locations in this area will be stored in $\mathcal{P}$. This will cause a heavier penalty for visiting this area than, for example, going to an area in which the UAV visited only once.

A short prediction horizon t_{hor} is taken to reduce computation times. However, we want that a predicted path can cover a sufficiently large distance from the UAV's current position. Also, a too short horizon would prevent the UAV to be able to turn around towards a previously measured updraft. The value for t_{hor} is chosen such that the UAV is just able to turn around.

The chosen parameters for the PVH method can be found in Table V.

V. ENVIRONMENT ESTIMATOR

The velocity of the updrafts in the atmosphere are unknown to the UAV. The terminal cost function defined in the NMPC problem, which should cause the specific height energy to be maximized at the end of the prediction horizon, uses an estimated mapping of these velocities in order to determine a trajectory of the UAV. In other words, the NMPC scheme will only use measured locations to estimate where the high energy areas in the atmosphere can be. In practice, these measurements will be received from an IMU. In the simulations, the updraft velocity will be directly obtained from the atmospheric test model, as discussed later. A method is required to make a regression between updraft measurements. We use a GRNN, which is available in a MATLAB Toolbox. GRNNs can generate a 'picture' of the atmosphere as soon as a few updraft estimates are received from the IMU, and can update the estimation by 'forgetting' old data. The estimator does not identify thermals as separate entities, but makes a weighted average between measured data using gaussian functions.

The way a GRNN is used in this paper is practical, because with each measurement, we add a neuron and instantly get an updated estimation of the atmosphere. We get a rough first impression of where the energy in the atmosphere can be. No new estimation of parameters is required, which means no extra computation time. When the neural network increases in size after a long flight, the computation time can become prohibitively high. However, under the assumption that the atmosphere changes slowly in time and the flight time of the UAV is short, the amount of neurons remains small enough.

In this study, the UAV is restricted to the updraft velocity data measured via sensors along the flight path only, which is

TABLE VI
PARAMETERS FOR ESTIMATION METHOD USING GRNN.

Description	Parameter [unit]	Value
Amount of a priori added neurons (PVH)	n_{ini} [-]	25
Weight of a priori added neurons (PVH)	n_w [m/s]	0
Interval adding neurons to GRNN	Δt_{GRNN} [s]	8
Spread GRNN	σ [-]	160

different from the other studies in which aircrafts are assumed to obtain a whole accurate landscape [28], or in which certain special tools are used such as remote-sensing infrared thermal cameras [20] for scenario detecting. This makes the problem more difficult, but reflective of reality. The inertial position and aircraft velocity are provided by an IMU. The aerodynamic parameters (i.e. AOA and the bank angle) are inferred from the dynamic model. Should the project eventually be implemented on a real aircraft, the ability of the algorithm to work without depending on external-support equipment will be desirable.

The predicted trajectory (4) and the GRNN spread are inputs to the network, where the latter is a constant. The updraft of the atmosphere is estimated at the UAV position with a specific time interval. A measurement consists of the absolute x, y location of the UAV combined with the updraft velocity at that location, which are used as weights in the Radial Basis Layer and the Special Linear Layer, respectively. The interval between two measurements is defined as Δt_{GRNN} seconds. There is again a trade-off between accuracy and computation time. More measurements will increase the accuracy at the cost of higher computation time. If required, external measurements can easily be added in the GRNN as additional neurons.

For the PVH method, one can manipulate the surface by adding neurons to the network before the UAV starts flying. There are two possibilities. The first possibility is to add neurons to represent a positive (upwards) velocity. This way the controller will 'think' that energy is everywhere and therefore it causes the UAV to search the complete atmosphere. When the UAV flies near a priori added neuron, that neuron will eventually become negligible compared to the larger amount of measurements that will be stored by the UAV during flight.

A second possibility is to add neurons to represent a negative (downwards) velocity. Now the controller 'thinks' that the complete atmospheric scenario will only remove potential energy from it. This will cause the UAV to avoid stronger downdrafts than the weight of these a priori-added neurons and the UAV will investigate any area with an upwards velocity better than this weight. In short, the UAV will search near and towards areas with vertical velocities higher than the negative velocity of the a priori-added neurons.

In this study, we employ neurons with a vertical velocity of 0 m/s. The UAV will therefore search and fly towards areas with positive velocity (higher than 0 m/s). This results in a balance between the two possibilities. The parameters of the estimation method using GRNN is listed in Table VI.

VI. SIMULATION STUDY

The Imperial College London Optimal Control Software (ICLOCS) [29] was used to solve the optimal control problem (in conjunction with MATLAB). Assuming there is no

horizontal wind influence in the first set of simulations, we used the 3DOF UAV model with $W_x \equiv 0$ for the optimal controller and simulate the trajectory of the UAV. Hence, the UAV is only under the vertical updraft influence, which remains constant throughout the whole flight duration. The closed-loop NMPC results were obtained from implementing the optimal controller in a receding horizon fashion with a sampling time of 2 s.

First, the computation time of the 3DOF model was compared with the 6DOF model of [26]. Second, a simulation with full knowledge of the atmosphere is run to set a benchmark for comparison with the proposed methods. Third, the TSP and PVH methods were tested in a static environment. Finally, both methods were tested under a horizontal wind disturbance.

A. Atmospheric Test Model

The tested updraft scenario is a modification of MATLAB's `peaks` function, which varies from previous research [5] where a single or multiple updraft cores (distant enough so that each other's influence can be ignored) are considered. The test area is chosen to be a 4-by-4 kilometer square, which defines x_{min} , x_{max} , y_{min} , and y_{max} as in Table II, equal to -2 km, 2 km, -2 km, and 2 km respectively.

The model employed demonstrates a more difficult problem of multiple updraft cores with different magnitudes and radii, which have overlapping regions of influence. Meanwhile, downdrafts are inserted and the average magnitude of the distribution is set to be negative in order to examine the controller performance in a challenging environment, where a UAV with a purely random flight will on average lose energy.

In addition, normal atmospheric conditions exist with the horizontal crosswind velocity close to zero at the earth surface, while the velocity increases nonlinearly when the altitude rises. This crosswind influence will resemble a more realistic dynamic atmospheric scenario. Assuming that the horizontal wind blows along the x -direction and is a function of altitude, we build a wind profile modified from [27]. The horizontal wind speed is peaked at 2.5 m/s, which is one-tenth of the strong wind close to ocean surface. With the definition of

$$G \triangleq \frac{W_{x,max}}{h_0}, \quad (17)$$

a quadratic cross wind model is given as

$$W_x \triangleq \begin{cases} G \left(Oh + \frac{1-O}{h_0} h^2 \right) & \text{if } h \leq h_0, \\ W_{x,max} & \text{if } h > h_0, \end{cases} \quad (18)$$

where the parameters $O = 1.7$, $h_0 = 1000$ m and $G = 0.0025$ are modified from [27] in order to simulate a normal horizontal wind condition on flatlands, as shown in Figure 3.

To take into account the wind influence on the thermal scenario, the updraft/downdraft scenario is assumed to shift itself in entirety in $-x$ direction, with a velocity of $W_{x,max}$, which is shown in Figure 4. At each location on the atmosphere over time, the vertical velocity is independent of height.

B. Computational Performance of 3DOF and 6DOF Models

A pivotal factor of a good controller is the computation time. The application of our 3DOF model with the interaction

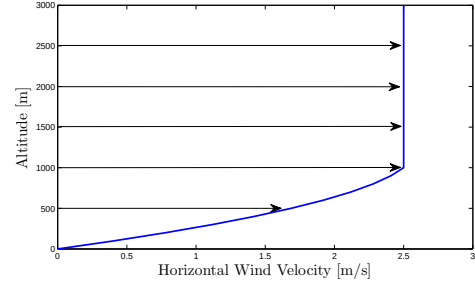


Fig. 3. The horizontal wind velocity profile in the $-x$ direction.

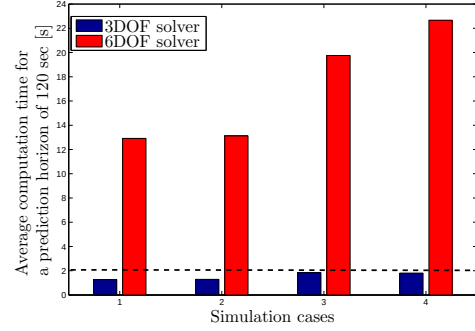


Fig. 5. Comparison for computation time: 3DOF solver vs 6DOF solver. Case 1: with full knowledge in static atmosphere. Case 2: with full knowledge in moving atmosphere. Case 3: without full knowledge in static atmosphere. Case 4: without full knowledge in moving atmosphere.

strategy allows the computation of the optimal solution to be 10 times faster than the situation where a 6DOF model [26] is used in the controller. Without giving the 6DOF model's details, which can be found in [26], we compared the average time cost for a solution with a 3DOF and 6DOF model. The test bench is a 4-core 3.6GHz CPU, 8GB RAM workstation, with MATLAB R2012a 64bit and ICLOCS Ver. 0.2.

Figure 5 shows that for four chosen simulation cases using the 6DOF model, it is not possible to compute a solution in less than 2 s, which is the sampling time we selected for all simulations. Using the 3DOF model significantly speeds up the solver. The less-than-2s average computation time of the 3DOF solver suggests that it is fast and in principle feasible to use this optimal energy-harvesting strategy for real-time UAV control if the complete control system is in real implementation with the help of C++/Matlab compilers.

C. Performance with Full Atmospheric Knowledge

The performance of the optimal control methods is tested via simulations of the UAV trajectory, first with full updraft distribution information in order to check the validity of NMPC. The optimization problem with the cost function in (9) is solved once and all inputs are used (no receding horizon). No online estimation method is employed (not needed) at this stage. The UAV is released nearby the strongest downdraft in the environment space.

When a control horizon of 120 s is employed, the UAV succeeds to converge to the strongest updraft, as in Fig. 6. The UAV first performs an avoidance maneuver and traverses the intermediate updraft core to increase the potential energy. The UAV finally circles around the strongest updraft core and gains energy from it until the end of the simulation.

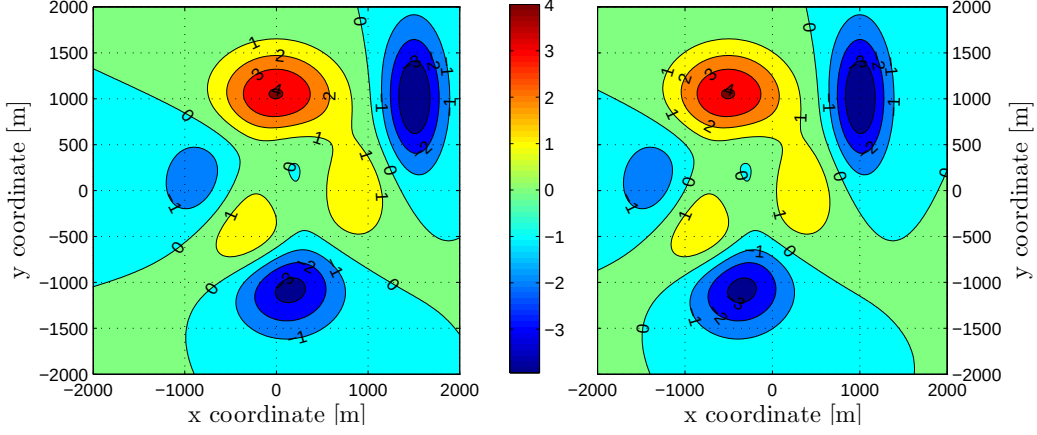


Fig. 4. (Left) Updraft magnitude contour at $t = 0$ s. (Right) Updraft magnitude contour at $t = 200$ s. The atmosphere moves at $W_{x,max}$ m/s in the $-x$ direction. The units of the contour (h direction) are [m/s].

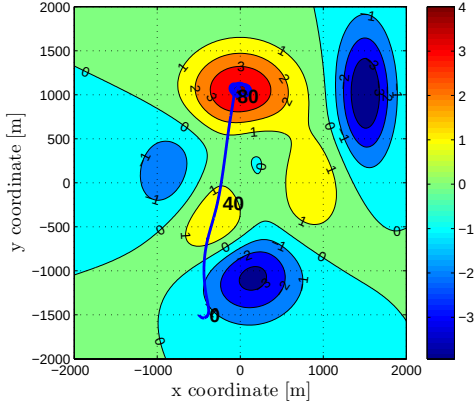


Fig. 6. Trajectory using a control horizon of 120 s. $x(0) = -500, y(0) = -1500, \psi(0) = -\frac{\pi}{4}$.

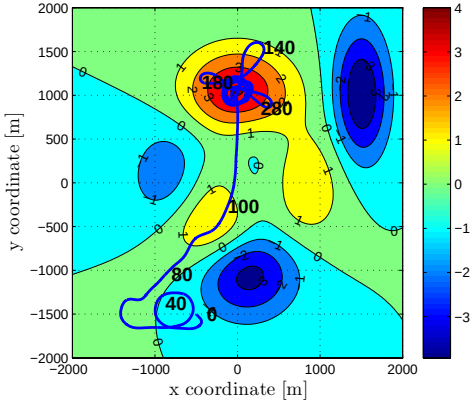


Fig. 7. Trajectory with systematic search and adaptive grid.

D. Performance without Atmospheric Knowledge (with Updraft Estimation)

We will now first use the TSP method to search for the optimal updraft. Following this, the PVH method is used in the same environment. These simulations will be without horizontal wind, in contrast to the simulations done in the next section. Many simulations have been run, but we will only show the few that illustrate the properties of both methods in the best way.

1) *Traveling Salesman Problem Method*: Figure 7 shows

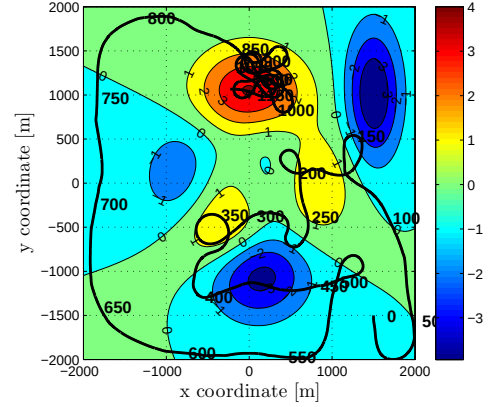


Fig. 8. Interesting case scenario with the PVH method. Initial conditions $x(0) = 1500, y(0) = -1500, \psi(0) = -\frac{\pi}{2}$.

the UAV's trajectory for the same release conditions as Section VI-C (Figure 6). At 130 s, the UAV traversed and experienced an updraft greater than the heuristic search condition of 2.5 m/s. Thereafter, the UAV proceeded to terminate the heuristic search and relied upon its online estimation of the environment to navigate towards the globally optimal updraft.

2) *Penalized Visited History Method*: Figure 8 is an example of a UAV's trajectory to show how the PVH method works. At first the UAV has to avoid the edge of the region. The UAV identifies a possible updraft between 100 and 150 s and therefore turns around to explore this updraft. The UAV then moves towards higher energy locations by turning around each time it heads towards a downdraft. At 400 s, a downdraft is found, causing the UAV to go in a straight line away from this location. When the simulation has passed 450 to 500 s, the UAV turns around, because it is surrounded by the visited history and cannot pass this region. From this moment until an updraft with a minimal velocity of 2.5 m/s is found, the UAV keeps moving with a high velocity without turning around. This is because the atmosphere does not deliver enough energy to the UAV to suggest that there could be a thermal nearby.

A downside of the method is that the UAV can get trapped within the visited history and/or the edge of the atmosphere. One could reduce the weight of the stage cost to allow the UAV to travel through penalized locations with more ease. Another solution could be to reduce the weight of points or

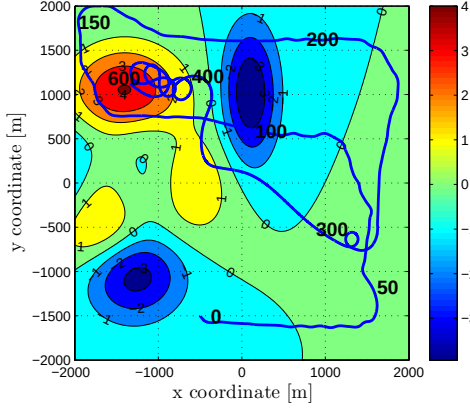


Fig. 9. Trajectory using TSP method at 600 s overlaid upon a time-varying updraft environment. The updraft information is withheld from the UAV. The UAV is at 935 m height. The unit of contour (h direction) is [m/s].

remove points in $\mathcal{P}$ depending on how long ago they were added. A last suggestion could be that the history trajectory can be crossed without cost if the UAV passes a region in a quasi perpendicular direction.

Reducing the weight of points will probably reduce computation time, since the influence of neurons far away from the UAV will be less, making the optimization problem easier. Removing points from $\mathcal{P}$ will reduce computation time, since the distance between the UAV and removed points no longer has to be calculated.

E. Performance with Horizontal Wind

The feasibility of NMPC on the soaring UAV was tested in previous sections with the assumption of no horizontal wind. A crosswind component is a better reflection of a real-world situation. Therefore, we tested the optimal control strategy including unknown cross wind (including as before, an unknown updraft distribution). We compared the performance of the TSP and PVH methods in a 700 s simulation. We again did many tests, but here only show the few that illustrate the properties of both methods in the best way.

1) *Traveling Salesman Problem Method*: The UAV's trajectory using the TSP search method at 600 s can be observed in Figure 9. The UAV started at 2000 m height. After a 400 s search of the strongest updraft, the UAV succeeded in reaching the optimal point and started soaring around the updraft core. However, when the UAV updates the updraft picture estimated after 60 s, the strongest updraft had already moved away, hence the UAV started to chase the updraft. Analysing the estimated updraft distribution, the estimation (Figure 10) is close to the real environment, particularly the moving centre of the strongest updraft core. This suggests that the UAV succeeds in following the environmental motion.

2) *Penalized Visited History Method*: In Figure 11, the UAV's trajectory, which started from 1500 m height, is shown for a crosswind scenario. One can see a similar motion for the no-crosswind scenario. The main difference here is that the strongest updraft is moving its location, which causes the UAV to circle slowly towards the left with the motion of this updraft when near the updraft core.

The estimate shown in Figure 12 indicates that the performance of the PVH method to chase the moving updraft is

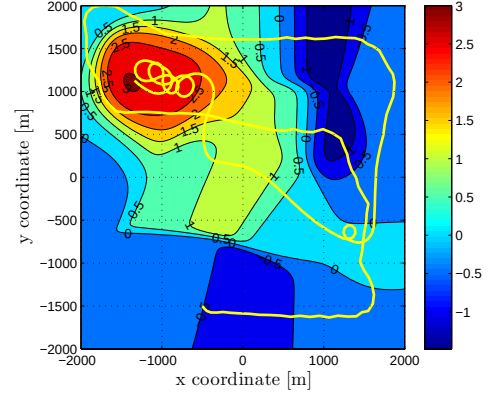


Fig. 10. Trajectory using TSP method at 600 s overlaid upon the estimated updraft environment. The UAV is at 935 m height. The unit of contour (h direction) is [m/s].

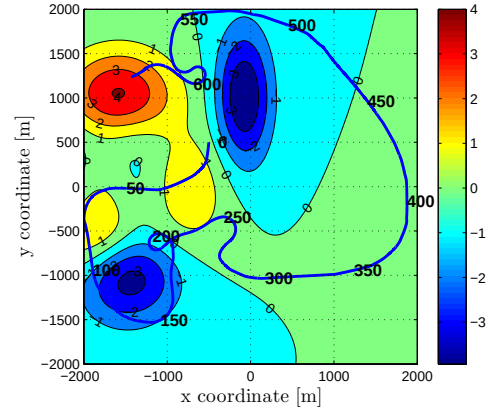


Fig. 11. Trajectory using PVH method at 632 s overlaid upon a time-varying updraft environment. The updraft information is withheld from the UAV. The UAV is at 1066 m height. The unit of contour (h direction) is [m/s].

successful. As already concluded with the TSP method, and since the same strategy is used as in the TSP method, the UAV is able to follow the motion of the atmosphere.

The total and potential energy of the UAV over time for the TSP and PVH exploration methods in the dynamic environment without prior updraft knowledge over a period of 700 s are shown in Figure 13. The results are displayed in the form of weight-specific energy height, as is the terminal cost $E(.,.)$ in (9b). The TSP method attempts to search the atmosphere faster, which causes the UAV to lose energy faster but travel a greater distance. The PVH method attempts to maximize energy, the UAV can therefore fly for a longer period of time, but it is possible that the sufficiently strong updraft is found at a later time than with the TSP method. Finally, from the trajectory contours as well as the energy exchange result graph, one can see that at the end of search, both methods enable the UAV to discover and chase the desired updraft, and therefore extract energy via soaring.

VII. CONCLUSIONS

In this paper, we have shown that the application of constrained nonlinear optimal control techniques on UAVs to extract the energy from atmospheric updrafts can be a promising method to enhance flight endurance, loiter time, reduce fuel consumption and increase battery life. The contribution of the

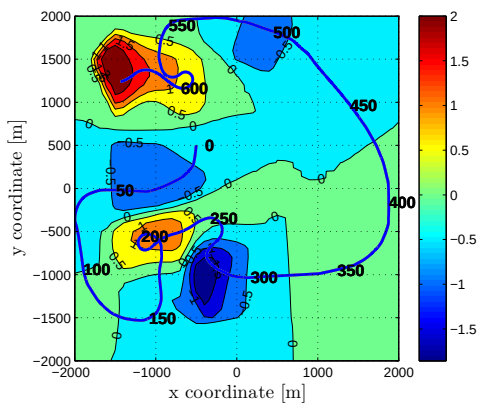


Fig. 12. Trajectory using PVH method at 632 s overlaid upon the estimated updraft environment. The UAV is at 1066 m height. The unit of contour (h direction) is [m/s].

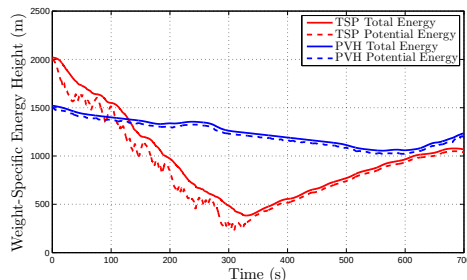


Fig. 13. The energy exchange results of the UAV over flight trajectory using TSP and PVH search method in the unknown dynamic environment.

paper is a solution to the energy harvesting problem in an unknown environment via an NMPC scheme. The solution of such an optimization problem has potential for real-time execution. Furthermore, the problem is practical because the assumption is made that no extra sensors are added to the UAV other than a standard IMU and airspeed sensor. It is shown by simulations that the adaptive grid heuristic search method, based on the TSP and the PVH method, are both effective strategies in the search of a high energy location, even in challenging, realistic and dynamic atmospheric scenarios. The NMPC structure used in this study could be further employed to help the UAV to extract energy also from horizontal wind gradients (dynamic soaring) as well as vertical updrafts. Future work could include an implementation of the proposed method in embedded hardware. The results of this study are also significant as they lend conceptual support to the feasibility of utilizing glider-type aircraft to harvest the naturally-occurring energy present in the atmosphere.

REFERENCES

- [1] Z. Akos, N. M., S. Leven, and T. Vicsek, "Thermal soaring flight of birds and unmanned aerial vehicles," *Bioinspiration & Biomimetics*, vol. 5, no. 4, 2010.
- [2] H. Shannon, G. Young, M. Yates, M. Fuller, and W. Seegar, "Measurements of thermal updraft intensity over complex terrain using american white pelicans and a simple boundary-layer forecast model," *Boundary Layer Meteorology*, vol. 104, pp. 167–199, 2002.
- [3] N. Slegers, J. Kyle, and M. Costello, "Nonlinear model predictive control technique for unmanned air vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 5, pp. 1179–1188, 2006.
- [4] B. Houska and M. Diehl, "Robustness and stability optimization of power generating kite systems in a periodic pumping mode," in *Proc. of the IEEE Multi - Conference on Systems and Control*, Yokohama, Japan, 2010.

- [5] N. Akhtar, A. K. Cooke, and J. F. Whidborne, "Positioning algorithm for autonomous thermal soaring," *AIAA Journal of Aircraft*, vol. 49, no. 2, pp. 472–482, 2012.
- [6] A. Chakrabarty and J. W. Langelaan, "Flight path planning for UAV atmospheric energy harvesting using heuristic search," in *AIAA Guidance, Navigation and Control Conference*, Toronto, Ontario, Canada, August 2010.
- [7] Y. C. Qi and Y. J. Zhao, "Energy-efficient trajectories of unmanned aerial vehicles flying through thermals," *Journal of Aerospace Engineering*, vol. 18, no. 2, pp. 84–92, 2005.
- [8] N. E. Kahveci, P. A. Ioannou, and M. D. Mirmirani, "Adaptive LQ control with anti-windup augmentation to optimize UAV performance in autonomous soaring applications," *IEEE Trans. Control Systems Technology*, vol. 16, no. 4, pp. 691–707, 2008.
- [9] M. J. Allen and V. Lin, "Guidance and control of an autonomous soaring UAV," NASA, TM – 2007 – 214611, Tech. Rep., 2007.
- [10] D. J. Edwards, "Implementation details and flight test results of an autonomous soaring controller," in *Proc. AIAA Guidance, Navigation and Control Conference*, Honolulu, Hawaii, 2008.
- [11] M. Deittert, A. Richards, C. A. Toomer, and A. Pipe, "Engineless unmanned aerial vehicle propulsion by dynamic soaring," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 5, pp. 1446–1457, 2009.
- [12] N. Akhtar, J. F. Whidborne, and A. K. Cooke, "Real-time optimal techniques for unmanned air vehicles fuel saving," *Proc. IMechE Journal of Aerospace Engineering*, vol. 226, no. 10, pp. 1315–1328, 2011.
- [13] J. Elston and E. W. Frew, "Unmanned aircraft guidance for penetration of pre-tornadoic storms," *Journal of Guidance, Control and Dynamics*, vol. 33, no. 1, pp. 99–107, 2010.
- [14] J. Wharington, "Control of a high endurance unmanned air vehicle," International Council of the Aeronautical Sciences and the American Institute of Aeronautics and Astronautics, Tech. Rep., 1998.
- [15] K. Andersson, I. Kaminer, V. Dobrokhodov, and V. Cichella, "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.
- [16] N. T. Depenbusch and J. W. Langelaan, "Coordinated mapping and exploration for autonomous soaring," in *Proc. Infotech @ Aerospace*, St Louis, Missouri, March 2011.
- [17] R. Bencatel, J. Sousa, M. Faied, and A. Girard, "Shear wind estimation," in *Proc. AIAA Guidance, Navigation, and Control Conference*, Portland, Oregon, USA, August 2011.
- [18] C. Gao and H. H. T. Liu, "Static soaring surveillance in the wind field," in *Proc. AIAA Guidance, Navigation and Control Conference*, Portland, Oregon, 2011.
- [19] J. W. Langelaan, "Gust energy extraction for mini and micro uninhabited aerial vehicles," *Journal of Guidance, Control and Dynamics*, vol. 32, no. 2, pp. 463–472, 2009.
- [20] W. B. Kagabo and J. R. Kolodziej, "Trajectory determination for energy efficient autonomous soaring," in *American Control Conference*, San Francisco, CA, USA, 2011.
- [21] L. D. Stone, *Theory of Optimal Search*, ser. Mathematics in Science and Engineering. New York: Academic Press, 1975, vol. 118.
- [22] L. Busoniu, R. Babuska, B. D. Schutter, and D. Ernst, *Reinforcement learning and dynamic programming using function approximators*, ser. Automation and Control Engineering. CRC Press, 2010.
- [23] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *Circuits and Systems Magazine, IEEE*, vol. 9, no. 3, pp. 32–50, 2009.
- [24] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, "Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations," *IEEE Trans. Automatic Control*, vol. 48, no. 2, pp. 246–258, 2003.
- [25] D. Lee, S. Longo, and E. C. Kerrigan, "Predictive control for soaring of unpowered autonomous UAVs," in *Proc. 4th IFAC Nonlinear Model Predictive Control Conference*, Netherlands, 2012.
- [26] Y. Liu, S. Longo, and E. C. Kerrigan, "Nonlinear predictive control of autonomous soaring UAVs using 3DOF models," in *The European Control Conference 13*, Zurich, Switzerland, July 2013.
- [27] Y. J. Zhao, "Optimal patterns of glider dynamic soaring," *Optimal Control Applications and Methods*, vol. 25, pp. 67–89, 2004.
- [28] Z. Kowalczyk and K. E. Olinski, "Gliding strategy design with the use of discrete optimization," in *Proc. of the 17th IFAC World Congress*, 2011, pp. 14 011–14 016.
- [29] P. Falugi, E. C. Kerrigan, and E. Van Wyk, Imperial College London Optimal Control Software (ICLOCS). Online: <http://www.ee.ic.ac.uk/ICLOCS/>.

AUTHOR INFORMATION



Yuyi Liu studied mechanics at Fudan University and mechanical engineering at the University of Birmingham in 2007–2011, where he received a B.Sc. degree and a B.Eng degree. He received his M.Sc. in aeronautical engineering from Imperial College London in 2012. He is currently pursuing a Ph.D. degree in Computer Science at the University of Tuebingen. His research interests include

the application the real implementation of nonlinear control algorithms on mobile robots. He is also active in the real-time applications of model predictive control methods.

Jos, please add your bios and send me a photo.



Stefano Longo received his M.Sc. in Control Systems from the University of Sheffield in 2007 and completed his Ph.D. in Control Systems at the University of Bristol in 2011. His Ph.D. thesis was awarded the Institution of Engineering and Technology (IET) Control and Automation Prize for significant achievements in the area of control engineering.

In November 2010 he was appointed to the position of Research Associate at Imperial College London. He was appointed to the position of lecturer (assistant professor) in Vehicles' Electrical and Electronic Systems at Cranfield University in the summer of 2012.



Eric C. Kerrigan (S'94–M'02) received a PhD from the University of Cambridge in 2001 and has been with Imperial College London since 2006. His research is focused on the development of efficient numerical methods and embedded computing architectures for solving advanced optimization, control and estimation problems in real-time. His work is applied to a variety of problems in the

design of cyber-physical and computing systems that arise in aerospace and renewable energy applications. He is on the IEEE Control Systems Society Conference Editorial Board and is an associate editor of the IEEE Transactions on Control Systems Technology, Control Engineering Practice, and Optimal Control Applications and Methods.