HAL
open science

Monitoring the Evolution of Clouds with UAVs

Alessandro Renzaglia, Christophe Reymann, Simon Lacroix

To cite this version:

Alessandro Renzaglia, Christophe Reymann, Simon Lacroix. Monitoring the Evolution of Clouds with UAVs. IEEE International Conference on Robotics and Automation (ICRA), May 2016, Stockholm, Sweden. pp. 278-283. hal-01275324

HAL Id: hal-01275324

https://hal.science/hal-01275324

Submitted on 17 Feb 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Monitoring the Evolution of Clouds with UAVs

Alessandro Renzaglia[1,2], Christophe Reymann[1,3] and Simon Lacroix[1,2]

***Abstract*— We study the problem of monitoring the evolution of atmospheric variables within low-altitude cumulus clouds with a fleet of Unmanned Aerial Vehicles (UAVs). To tackle this challenge, two main problems can be identified: *i)* creating on-line maps of the relevant variables, based on sparse local measurements; *ii)* designing a planning algorithm which exploits the obtained map to generate trajectories that optimize the adaptive data sampling process, minimizing the uncertainty in the map, while steering the vehicles within the air flows to generate energetic-efficient flights. Our approach is based on Gaussian Processes (GP) for the mapping, combined with a stochastic optimization scheme for the trajectories generation. The system is tested in simulations carried out using a realistic three-dimensional current field. Results for a single UAV as well as for a fleet of multiple UAVs, sharing information to cooperatively achieve the mission, are provided.**

I. Introduction

Atmospheric scientists have been early users of UAVs, from which significant scientific results have rapidly been obtained, see *e.g.* [1]. UAVs indeed bring forth several advantages over manned flight to probe atmospheric phenomena: low cost, ease of deployment, possibility to evolve in high turbulences [2], etc. This article depicts on-going work on the development of an approach to probe low-altitude cumulus clouds with UAVs. From an atmospheric science point of view, there remain numerous uncertainties and even unknowns in the cloud micro-physics models that could be alleviated with the acquisition of a variety of data *within and around* the cloud. Wind currents, pressure, temperature, humidity, liquid water content, radiance and aerosols are variables of interest that must be collected with a spatial and temporal resolution of respectively about 10 m and 1 Hz over the cloud lifespan to better understand the phenomena that conduct cumulus formation and evolution.

Exploiting UAVs to explore a cloud is a poorly informed and highly constrained adaptive sampling problem, in which the UAV motions must be defined so as to maximize the amount of gathered information and the mission duration. The work presented here tackles this challenge. It is developed within the context of the SkyScanner project[1], which involves atmosphere and drone scientists. Its goal is to deploy a fleet of UAVs to explore cumulus clouds, so as to be able to synchronously gather data in various areas of the clouds.

A global approach has been defined, which casts the overall problem in a hierarchy of two modeling and decision stages. A macroscopic parametrized model of the cloud (Fig. 1) is built and exploited at the higher level by an atmospheric scientist, which sets information gathering goals. A UAV or a subset of the UAV fleet is allocated to each goal, considering *e.g.* the UAVs current position in the cloud, their on-board energy level, and their sensing capacities.

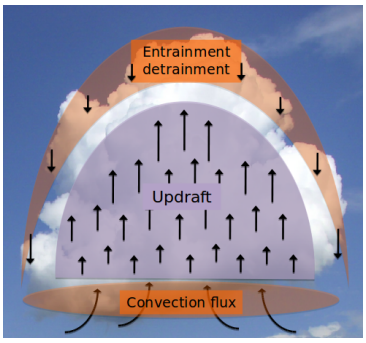

Fig. 1. Coarse schematic representation of a cumulus cloud. The black arrows represent the wind vertical velocities, the orange blobs denote areas where mixing is occurring between the cloud and the surrounding atmosphere. The precise understanding of these phenomena, as well as the definition of laws that relate the cloud dimensions, the inner wind speeds, and the spatial distribution of the various thermodynamic variables is still a matter of research, for which UAVs can bring significant insights. The cloud dimensions can vary from 100m to several hundreds of meters.

These high level goals typically consist of cloud regions to explore, and are handled by the lower level, which autonomously optimizes the selected UAVs trajectories using an on-line updated dense model of the variables of interest. The article focuses on this latter level, the variables of interest being the winds 3D coordinates, making the problem an "explore vs. exploit" one.

Related work: While UAVs have been used for data sampling within clouds, they are usually employed following predefined standard trajectories: adaptive energetically-optimal autonomous missions are still a challenge. In the literature, few recent works have tackled the problem considering realistic models. The possibility of using dynamic soaring to extend the mission duration for sampling in supercell thunderstorms has been presented in [3]. In this case, only the energetic part is analyzed, while the gathered information does not affect the planning. In [4], Lawrance and Sukkarieh present a problem very close to ours, where a glider explores a wind field trying to exploit air flows to augment flight duration. This work presents a hierarchic approach for the planning, where a target point is firstly selected and then a trajectory to reach it is generated for every planning cycle. In a similar scenario, a reinforcement learning algorithm to find a trade-off between energy harvesting and exploration is proposed in [5]. The problem of tracking and mapping

[1]CNRS, LAAS, 7 av du colonel Roche, F-31400 Toulouse, France; {firstname.lastname} at laas.fr
[2]Univ de Toulouse, LAAS, F-31400 Toulouse, France
[3]Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France
[1]https://www.laas.fr/projects/skyscanner/

atmospheric phenomena with a UAV is also studied in [6]. Even though this latter work does not take into account air currents for the navigation, it is worth to remark that it includes experiments with a real platform – contrary to the previous ones. Autonomous soaring has also been studied in different scenarios, as in [7], where a glider has to search for a target on the ground. The goal here is to maximize the probability of detecting the target traveling between thermals with known location. In all the aforementioned work, only the use of a single UAV to achieve the mission is considered, and no cooperative multi-robot strategies are proposed.

Autonomous exploration of current fields is not exclusively related to aerial applications: the use of Autonomous Underwater Vehicles for oceanographic studies has been recently investigated [8], [9].

Outline: Section II depicts how the area to explore is mapped with Gaussian Processes using data gathered locally by the UAVs, focusing in particular on the mapping of the wind, which conditions the UAV motions. The resulting map is exploited to derive a utility measure, so as to drive the exploration process, which is depicted in Section III. The exploration process aims at augmenting the information contained in the map, while optimizing the energy consumption. It is defined for a glider, and exploits a stochastic optimization scheme. Section IV then presents some exploration results, using a realistic atmospheric simulator to generate inputs for the mapping processes, and a flight simulator to steer the UAVs according to the planned motions.

II. Mapping clouds

To plan energy-efficient and informative trajectories, a model that represents both the wind currents and the atmospheric variables to measure is required. The accuracy of this information is of course of utmost importance, as it is the factor that steers the information gathering mission and that conditions the expectation of the path costs estimates.

The considered context raises two main issues: the size of the three-dimensional space, in which UAVs collect very sparse measurements, and the dynamics of the considered atmospheric phenomena. The only way to be able to predict short term atmospheric conditions from the sparse measurements is to make use of the strong spatio-temporal correlations of the atmospheric processes. Considering the little available knowledge about these, there are not many efficient tools to tackle this problem. Among them, Gaussian Processes (GP) can successfully be used to perform on-line spatio-temporal regression in robotics problems. We use a GP mapping framework similar to [4], with the added ability to update hyper-parameters during the mission.

A. Gaussian Process Model

GP is a very general non-parametric framework, where the underlying process is modeled by "a collection of random variables, any finite number of which have a joint Gaussian distribution" [10]. Under this assumption, the process $y = f(x)$ is defined only by its mean and covariance functions:

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x},\mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x})(f(\mathbf{x}') - m(\mathbf{x}'))]\,. \end{aligned} \tag{1}$$

The mean function m is often assumed to be zero, but it can be used to encode prior knowledge. The covariance function k represents similarity between points: given a set of n samples (x, y) and assuming zero mean, the GP is fully defined by the Gram matrix $\Sigma_{X,X} = [k(x_i, x_j)]$ of the covariances between all pairs of sample points. The particularity of the GP model is to provide full predictive distributions over all possible f, whose mean and variance at each point can be interpreted as the process predicted value and associated error. Indeed, with the previous knowledge of a set (X,Y) of examples and assuming a zero mean process, one can infer the marginal values of the mean $\bar{y}_\star$ and variance $\mathbb{V}[y_\star]$ of the process at a location $x_\star$ using:

$$\begin{aligned} \bar{y}_\star &= \Sigma_{\mathbf{x}_\star,\mathbf{X}}[\Sigma_{X,X} + \sigma_n^2 I]^{-1}\mathbf{Y}, \\ \mathbb{V}[y_\star] &= k(\mathbf{x}_\star,\mathbf{x}_\star) - \Sigma_{\mathbf{x}_\star,\mathbf{X}}[\Sigma_{X,X} + \sigma_n^2 I]^{-1}\Sigma_{\mathbf{x}_\star,\mathbf{X}}^\top \end{aligned} \tag{2}$$

where the $\sigma_n^2 I$ perturbation added to the covariance matrix is used to model Gaussian noise on the examples.
The choice of the kernel function k conditions the process distribution as it sets a prior on the process properties such as isotropy, stationarity or smoothness. In practice, a family of kernels parametrized by hyper-parameters has to be chosen. The most used, which we adopt in this article, is the squared exponential kernel:

$$k_{SE}(\mathbf{x},\mathbf{x}') = \sigma_f^2 e^{-\frac{1}{2}|\mathbf{x}-\mathbf{x}'|M|\mathbf{x}-\mathbf{x}'|} \tag{3}$$

with $M = \mathbf{l}^{-2} I$ a diagonal matrix defining characteristic anisotropic length scales $\mathbf{l}$. This kernel is thus anisotropic, stationary and infinitely smooth.

Until recently, the usage of GP models for on-line problems has been prevented by prohibitive inference cost in $\mathcal{O}(n^3)$, due to the inversion of the covariance matrix $\Sigma_{\mathbf{X},\mathbf{X}}$, which must be updated each time new samples are added or each time hyper-parameters vary. Recent algorithmic advances in streaming data and greater computing power spawned contributions that help to solve on-line problems [11]. Nevertheless, the issue of selecting the hyper-parameters θ remains. In most settings, their values are optimized off-line using the log-likelihood criterion:

$$\log p(\mathbf{y}|\mathbf{X},\theta) = -\frac{1}{2}\mathbf{y}^\top\Sigma^{-1}\mathbf{y} - \frac{1}{2}\log|\Sigma| - \frac{n}{2}\log 2\pi \tag{4}$$

This non-convex optimization may not converge to the global minimum and is relatively slow. Furthermore, in our context it cannot be performed off-line: the hyper-parameters indeed depend on the dynamics of the cloud, where different regions have radically different characteristics.

B. Evaluating information utility of trajectories

For a mapping mission, the quantity of the information that is collected along the path of the UAV is the primary concern to decide the trajectories to achieve. The utility of

new measurements is assessed by the information gain they bring to the current model. For this purpose, the variance of the GP as defined by eq. (2), which represents the uncertainty of the current model at each point in space, is a natural candidate to evaluate the utility of new measurements. In [5], Chung *et al.* integrate the variance of the GP over the region to be mapped and derive a measure of the quality of the model. Unfortunately, a closed-from expression of this integral does not exist in general.

The problem of selecting the best possible future measurements to estimate a statistical model has been extensively studied. The idea is to minimize the variance of the estimator using a statistical criterion. The integration criterion defined in [5] is an instance of I-optimality. Other classical criteria directly seek to minimize the covariance matrix:

- D-optimality aims to maximize the differential Shannon entropy of the statistical model, which comes to maximizing the determinant of the information matrix (the inverse of the covariance matrix).
- T-optimality aims to maximize the trace of the information matrix.

To efficiently evaluate the information gain of a new set of measurement points, we define the conditional covariance $\Sigma_{\mathbf{X}_{new}|\mathbf{X}}$ of the set of m new points $\mathbf{X}_{new}$ against $\mathbf{X}$ the ones already included in the regression model:

$$\Sigma_{\mathbf{X}_{new}|\mathbf{X}} = \Sigma_{\mathbf{X}_{new},\mathbf{X}_{new}} - \Sigma_{\mathbf{X}_{new},\mathbf{X}}\Sigma^{-1}_{\mathbf{X},\mathbf{X}}\Sigma^{\top}_{\mathbf{X}_{new},\mathbf{X}} \quad (5)$$

The $\Sigma_{\mathbf{X}_{new}|\mathbf{X}}$ matrix is of fixed size $m \times m$ which is independent of the size of the model, which yield swift computations. The matrix itself is computed in $\mathcal{O}(nm^2 + mn^2)$, subsequent inversion or computation of the determinant are performed in $\mathcal{O}(m^3)$. The values v_D and v_T of the D- and T-optimality criterion are thus defined as:

$$v_T(\Sigma_{\mathbf{X}_{new}|\mathbf{X}}) = tr([\Sigma_{\mathbf{X}_{new}|\mathbf{X}} + \sigma_n^2 I]^{-1}) \quad (6)$$

$$v_D(\Sigma_{\mathbf{X}_{new}|\mathbf{X}}) = \frac{1}{2} log|[\Sigma_{\mathbf{X}_{new}|\mathbf{X}} + \sigma_n^2 I]^{-1}| + \frac{m}{2} log 2\pi e \quad (7)$$

As can be seen in Fig. 2, the proposed way of computing T- and D- criteria is faster than the computation of I-optimality, and thus scales better with the number of samples. Additionally, T- and I-optimality tend to exhibit the same behavior, and T-optimality does not depend on the type of the chosen kernel.

Although there may not exist a minimum value for all criteria, there exists a maximum value: in all case (provided the kernel is stationary), the most informative measurements will have a diagonal covariance matrix $\Sigma_{opt} = (k(0,0) + \sigma_n^2)I$. Indeed the information is maximized when all measurements are independent. The maximal value v_{opt} is thus defined using Σ_{opt} instead of $[\Sigma_{\mathbf{X}_{new}|\mathbf{X}} + \sigma_n^2 I]$ in equation (6) and (7). This bound will be used to normalize the objective function to maximize (section III-B).

III. Optimal Trajectories Planning

The algorithm to guide the UAVs must optimize the two main objectives of the considered mission: minimize the uncertainty on the mapped atmospheric variables, and exploit the air flows to augment the mission duration. The map resulting from the GP provides the information needed to generate the optimal trajectories, which are generated over a sequence of short-time horizons ΔT, defined by the frequency at which the GP hyper-parameters are updated. Before detailing the optimization algorithm, we firstly describe the UAV model and we formalize the optimization criteria to define the objective function.

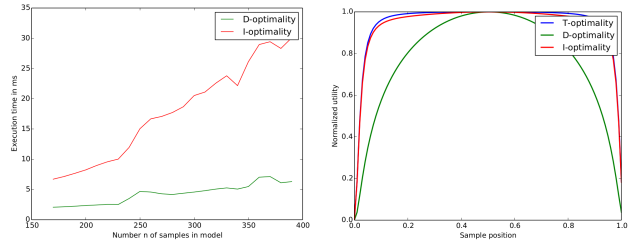

Fig. 2. Comparing D-, T- and I-optimality information criteria. I-optimality has been implemented using the formula described in [5]. Left: execution time depending of number of samples in the model (T-optimality is not shown as it is confounded with D-optimality) Right: normalized utility of a single sample taken in 1D space on a line between samples taken every unit, using k_{se} kernel ($\sigma_f = 1$, $l = 0.5$, $\sigma_n = 1e-3$).

A. UAV model

Even though the project future experiments will exploit moto-gliders, we consider a glider without any means of propulsion to test and validate our approach in this paper. The absence of propulsion does not change the nature of the problem, neither over-simplifies it, and hence remains a valid and realistic test for the optimization algorithms.

Let us consider a classic UAV dynamics model, expressed as follows:

$$\mathbf{p}(t+dt) = \mathbf{p}(t) + (\mathbf{V} + \mathbf{c})dt \quad (8)$$

$$\psi(t+dt) = \psi(t) + \dot{\psi}(t)\,dt \quad (9)$$

where $\mathbf{c}$ is the wind velocity, $\mathbf{p}$ the UAV position, ψ its yaw angle and $\mathbf{V}$ the velocity relative to the air currents, whose components are:

$$v_x = V\cos\theta\cos\psi \quad (10)$$

$$v_y = V\cos\theta\sin\psi \quad (11)$$

$$v_z = V\sin\theta \quad (12)$$

$$\theta = \arctan\left(\frac{1}{L/D\cos\phi}\right) \quad (13)$$

$$\phi = \arctan\left(\frac{\dot{\psi}V}{g}\right) \quad (14)$$

where $V = \|\mathbf{V}\|$ is the airspeed and L/D is the lift-to-drag ratio. In our model we have $V = 15\,\mathrm{ms}^{-1}$ and $L/D = 30$. From the previous equations, we can notice that the entire motion is completely defined by V and ψ. However, the airspeed V depends mainly on the angle of attack and is usually considered as constant during the flight. As a result,

in our model the trajectory is uniquely defined by the yaw angle, and so the only optimization variable is $\dot{\psi}$, which is constrained by $\dot{\psi} \in [\dot{\psi}_{min}, \dot{\psi}_{max}] = \Delta\psi_{max}$. No assumption has been made on the current speed, which can be also greater than V, and hence define unreachable areas.

B. Utility function

The following step to formalize the optimization problem is to define a utility function which incorporates the criteria corresponding to energy and gathered information. As a third criterion, we also included a penalization term to force the UAVs to remain within a given region of interest, which can be considered as a soft constraint to satisfy.

1) Energy: The first term, U_E, reflects the aim of flying and mapping at a given optimal altitude z_{opt}, which is fixed for every planning horizon ΔT. The gliders will thus need to periodically exploit the updraft present into the cloud to regain this altitude, compensating the continuous loss of energy. Note also that the vertical currents are completely unknown at the beginning of the mission and estimated during the exploration by the mapping process. Formally, we define U_E as follows:

$$U_E(\mathrm{dz}^{(j)}) = \left(1 - \frac{|\mathrm{sign}(z_{opt} - z_{in}^{(j)})\mathrm{dz}_{max} - \mathrm{dz}^{(j)}|}{\mathrm{dz}_{max} + |\mathrm{dz}^{(j)}|}\right) \times \times \left(1 - \frac{1}{1 + \frac{(z_{opt} - z_{in}^{(j)})^2}{l_E}}\right) \tag{15}$$

$$U_E(\mathrm{dz}) = \frac{1}{N_r} \sum_{j \in [1..N_r]} U_E(\mathrm{dz}^{(j)}), \tag{16}$$

where $\mathrm{dz}^{(j)}$ and $z_{in}^{(j)}$ are the total difference in altitude and the initial altitude for the j-th glider respectively.

2) Information: The mission goal is to minimize the uncertainty in the built map. To define an objective function, we adopted the D-optimality criterion, as defined in eq. (7), with samples along the trajectories of all UAVs. The resulting value v is then normalized using the theoretical optimal value v_{opt}, as defined in section II-B:

$$U_I(v) = \frac{1}{1 + \frac{(v_{opt} - v)^2}{l_I}}. \tag{17}$$

The information utility is computed for all UAVs at once, and thus favors trajectories that are not too close, as they would gather redundant information.

3) Boundaries: Finally, to force the gliders to remain in a predefined region of interest, we include a term, U_B, to penalize paths that partially exit this area:

$$U_B(n_{out}) = \frac{1}{1 + \frac{n_{out}^2}{l_B}}, \tag{18}$$

where n_{out} is the number of total way-points external to the region of interest.

All the previous terms are normalized against a known maximum value to be $U_x \in [0, 1]$ and the parameters l_x are fixed scale factors which were manually tuned. Combining these three utility terms, the final utility function we aim to maximize along the path is:

$$U = \frac{1}{3}(U_E + U_I + U_B). \tag{19}$$

In future work it is our intention to explore also different methods to tackle this multi-criteria optimization problems, e.g using Multi Criteria Decision Making approaches [12].

C. Trajectory generation

To generate optimal UAVs trajectories, for each planning horizon we consider m sections of duration dt in which the UAVs yaw angles are constant. As a result, the trajectory for the robot j during ΔT is described by the sequence $\psi_i^{(j)}$, with $i \in \{1, ..., m\}$, and a given initial condition $\psi_0^{(j)}$.

For every ΔT, we can now formulate the trajectory generation problem, which consists in maximizing eq. (19) subject to the constraints:

$$|\psi_i^{(j)} - \psi_{i-1}^{(j)}| \leq \Delta\psi_{max} \quad \forall i, j. \tag{20}$$

To tackle this optimization, we propose a centralized[2] two-step approach: a first phase based on a blind random search in order to have a good trajectories initialization, followed by a gradient ascent algorithm to optimize them. To perform the gradient ascent we adopted a constrained version of the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm [13], [14]. This algorithm ensures a faster convergence to a local maximum with respect to classic gradient approximation algorithms, as for example the Finite Difference Stochastic Approximation, where the optimization variables are varied one at a time instead of simultaneously. At every algorithm iteration k, the optimization variables $\boldsymbol{\psi}$ are then updated as follows:

$$\boldsymbol{\psi}_{k+1} = \Pi(\boldsymbol{\psi}_k + a_k \hat{g}(\boldsymbol{\psi}_k)), \tag{21}$$

where Π is a projection operator to force $\boldsymbol{\psi}$ to stay in the feasible space, and $\hat{g}$ is the gradient approximation, for which we used the one-sided version:

$$\hat{g}_k(\boldsymbol{\psi}_k) = \begin{bmatrix} \frac{U(\boldsymbol{\psi}_k + c_k\Delta_k) - U(\boldsymbol{\psi}_k)}{c_k\Delta_{k1}} \\ \vdots \\ \frac{U(\boldsymbol{\psi}_k + c_k\Delta_k) - U(\boldsymbol{\psi}_k)}{c_k\Delta_{kN}} \end{bmatrix} \tag{22}$$

To ensure the convergence of the algorithm, a simple and popular distribution for the random perturbation vector Δ_k is the symmetric Bernoulli ± 1 distribution, and the standard conditions on the gain sequences a_k, c_k are:

$$a_k > 0,\ c_k > 0,\ a_k \to 0,\ c_k \to 0, \quad \sum_{k=0}^{\infty} a_k = \infty, \quad \sum_{k=0}^{\infty} \frac{a_k^2}{c_k^2} < \infty. \tag{23}$$

[2]Note that the optimization is achieved across the joint space of all UAVs. Even though this would scale poorly with the number of robots, in our application we do not deal with large swarms, but only small fleets of typically 3-5 aircrafts.

The first phase of our optimization process, based on a blind random search, is instead achieved creating a set of feasible trajectories obtained by a constrained random sampling of directions angles ψ_i, and exploiting the approximated field generated by the GP regression. The trajectories are then evaluated using the utility function U and the best set of N_r trajectories is the initial configuration for the gradient ascent phase. The presence of the first sampling step is due to the dependence of the gradient-based solution on the initial configuration. In this way, even though we only have local convergence guarantees, the probability of getting stuck in local maxima far from the global optimum is reduced.

Figure 3 shows some trajectories obtained for a simple two-dimensional case where fictitious current field and utility maps has been defined to make the results easy to visualize and understand. Here the optimization function is given by the sum of the utility collected along the trajectory.

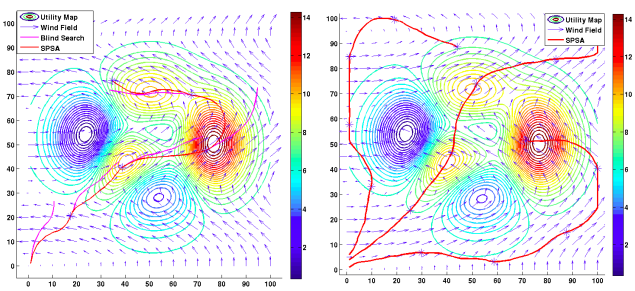

Fig. 3. Left: one UAV is moving in a 2D environment where a scalar utility map and a wind field are defined. The trajectories initialized by a blind search at every planning-horizon ΔT are shown in magenta, and the final trajectories provided by the SPSA algorithm are in red. Right: 3 UAVs are steered in the same environment to maximize the total utility (only the final trajectories are shown).

IV. Results

Working with meteorologists, we use realistic meso-scale atmospheric simulations as settings of our experiments. These simulations are provided by the CNRM-GAME lab[3] and generated using the Meso-NH[4] atmospheric model. These micro-physics simulations use preset initial conditions and let field of clouds evolve over the course of a day. The simulations provided have a 10 m spatial resolution and 1 s time resolution, over a field of $4 \times 4\,\mathrm{km}^2$, and up to an altitude of 3 km. A typical cumulus cloud field as generated by the Meso-NH simulation is shown in Fig. 4. Each time-frame occupies about 700 Mb, one hour over 2.5 Tb. Due to technical restrictions, we currently use only frozen time-frame (one single second) to test our models.

The 3D wind produced by the simulation is the "ground truth". It is used to simulate data acquisition for the mapping process, and as an input to the flight simulator.

To test our mapping and optimization scheme, we firstly simulate one single UAV flying in the vector field generated by a Meso-NH simulation at a given time, *i.e.* it is not varying

[3]http://www.cnrm.meteo.fr/?lang=en

[4]http://mesonh.aero.obs-mip.fr

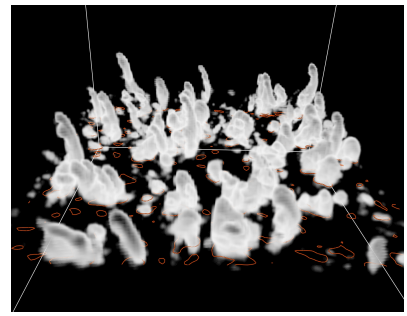

Fig. 4. View from above of a cumulus cloud field generated by a Meso-NH simulation (vertical axis stretched). Clouds are identified by the presence of liquid water (concentration in shades of gray), isometric curves of vertical $0.5\,\mathrm{ms}^{-1}$ upwind on the 1 km altitude plane are drawn in orange.

in time. The simulated flight starts from an altitude of 900 m already in an area that contains vertical currents. The optimal altitude z_{opt} for the exploration phase is set at 1000 m. Fig. 5 shows the 3D trajectory of the glider and the behavior of its height of flight as a function of time. The glider remains into the updraft until the optimal altitude is reached, it then starts to explore more distant areas, before coming back again so as to regain the required altitude and pursue the exploration. The result of the GP regression is presented in Fig. 6. The map of the true vertical wind is compared with the predicted one obtained using measurements taken along the trajectory. Additionally, absolute error and predicted variance are also shown: they show the ability of the model to estimate well the wind around the explored area.

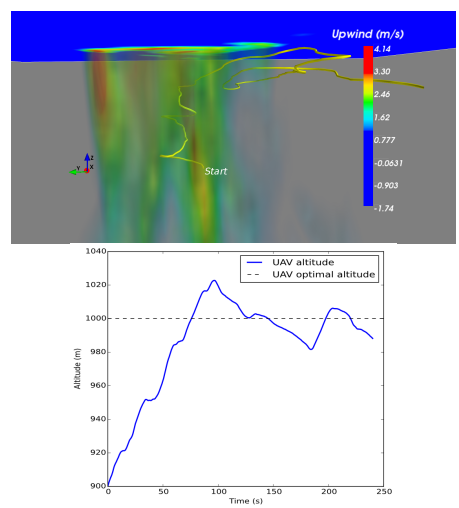

Fig. 5. Top: trajectory executed by a single UAV in a 3D currents field. Only the vertical component of the wind is visualized (but the flight is also affected by horizontal winds). Bottom: evolution of the UAV altitude during time.

Fig. 7 presents the results for a scenario that involves 3 UAVs, with the same initial conditions as for the single UAV scenario (the 3 UAVs start from the same location). The UAVs exploit the vertical currents to reach the required altitude, where they stabilize. The maximization of the information gathered by the fleet lead also to a spread of the UAVs

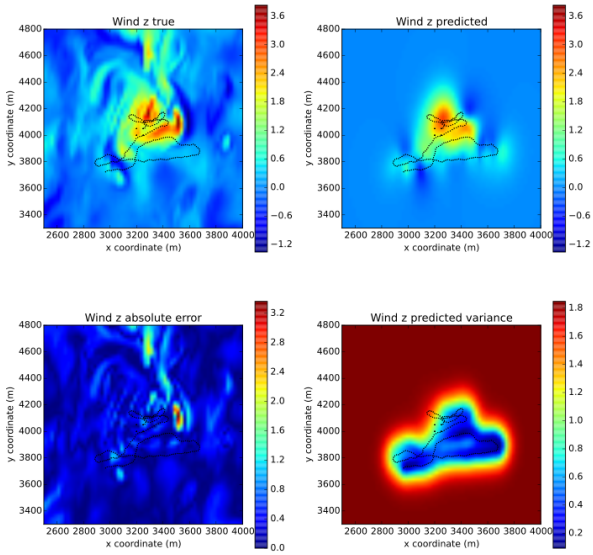

Fig. 6. Maps showing the true and predicted vertical wind, the absolute error and its predicted variance. Black dots are measurements taken into account in the model for the prediction.

over the area of interest to avoid overlapping information[5].

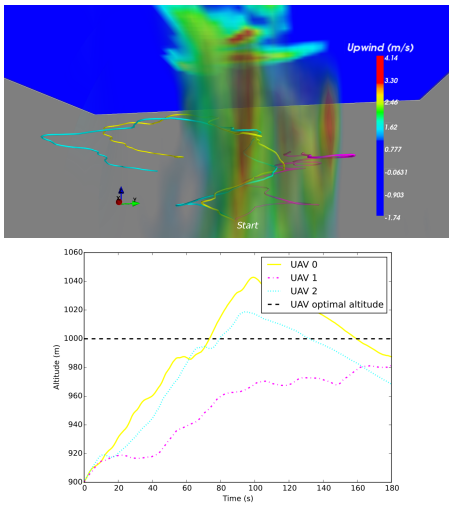

Fig. 7. Top: same scenario as in Fig. 5, but with three UAVs exploring the environment, all with the same required optimal altitude.

V. Conclusions and Future Work

This paper presented first results of a challenging project whose objective is to navigate a fleet of UAVs within low-altitude clouds to optimally gather atmospheric data, while exploiting air flows to achieve enduring flights. An approach to map atmospheric variables based on local measurements is integrated with a planning algorithm to generate trajectories which aims at simultaneously maximizing the accuracy on the map and optimize an energetic criterion. The results obtained in simulation show the effectiveness of the approach for both a single UAV and a multi-UAV system.

We aim to extend this work along several directions. Firstly, considering a moto-glider rather than a glider, including a new energetic consumption model, and time-varying currents will better match the future experimental phase. Then, beyond the local planning presented in this paper, the design of a global mission which combines the coarse and local cloud models will be crucial to reach the project's goals. A global planner will also include total energy information to ensure that a safe return of the UAVs to ground is possible at every moment. The mapping approach will also benefit from the integration of a global cloud model, and will consider other variables than wind. Finally, we will consider other planning requirements during the data sampling mission, such as synchronized observations in distant areas or formation specifications within the fleet.

[5]No collision avoidance check has been implemented so far, this will of course be included for future experimental tests

Acknowledgments

This work is made in the context of the SkyScanner project, supported by the STAE foundation. The authors want to thank Fayçal Lamraoui, from CNRM-GAME lab, to have provided the Meso-NH simulation results used in this work.

References

[1] V. Ramanathan, M. V. Ramana, G. Roberts, D. Kim, C. Corrigan, C. Chung, and D. Winker, "Warming trends in Asia amplified by brown cloud solar absorption," *Nature*, vol. 448, no. 7153, 2007.

[2] J. S. Elston, J. Roadman, M. Stachura, B. Argrow, A. Houston, and E. Frew, "The tempest unmanned aircraft system for in situ observations of tornadic supercells: design and VORTEX2 flight results," *Journal of Field Robotics*, vol. 28, no. 4, pp. 461–483, 2011.

[3] J. Elston and B. Argrow, "Energy efficient UAS flight planning for characterizing features of supercell thunderstorms," in *IEEE ICRA*, 2014.

[4] N. R. Lawrance and S. Sukkarieh, "Autonomous exploration of a wind field with a gliding aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 3, pp. 719–733, 2011.

[5] J. J. Chung, N. R. Lawrance, and S. Sukkarieh, "Learning to soar: Resource-constrained exploration in reinforcement learning," *IJRR*, vol. 34, no. 2, 2015.

[6] S. Ravela, T. Vigil, and I. Sleder, "Tracking and Mapping Coherent Structures," in *Int. Conf. on Computational Science*, 2013.

[7] J. Nguyen, N. Lawrance, R. Fitch, and S. Sukkarieh, "Energy-constrained motion planning for information gathering with autonomous aerial soaring," in *ICRA*, 2013.

[8] J. Das, J. Harvey, F. Py, H. Vathsangam, R. Graham, K. Rajan, and G. Sukhatme, "Hierarchical probabilistic regression for AUV-based adaptive sampling of marine phenomena," in *IEEE ICRA*, 2013.

[9] M. Michini, M. A. Hsieh, E. Forgoston, and I. B. Schwartz, "Robotic tracking of coherent structures in flows," *Robotics, IEEE Transactions on*, vol. 30, no. 3, pp. 593–603, 2014.

[10] C. E. Rasmussen and C. K. Williams, "Gaussian processes for machine learning," *the MIT Press*, 2006.

[11] A. Ranganathan, M.-H. Yang, and J. Ho, "Online sparse Gaussian process regression and its applications," *Image Processing, IEEE Transactions on*, vol. 20, no. 2, pp. 391–404, 2011.

[12] F. Amigoni and N. Basilico, "A decision-theoretic framework to select effective observation locations in robotic search and rescue scenarios," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) - Workshop on Search and Pursuit/Evasion in the Physical World: Efficiency, Scalability, and Guarantees*, 2010.

[13] J. C. Spall, *Introduction to stochastic search and optimization: estimation, simulation, and control*. John Wiley & Sons, 2005, vol. 65.

[14] P. Sadegh, "Constrained optimization via stochastic approximation with a simultaneous perturbation gradient approximation," *Automatica*, vol. 33, no. 5, pp. 889–892, 1997.