2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Thermal detection and generation of collision-free trajectories for cooperative soaring UAVs

J. A. Cobano[1], D. Alejo[1], S. Sukkarieh[2], G. Heredia[1] and A. Ollero[1]

Abstract— **This paper presents a cooperative system architecture that extends the flight duration of multiple gliding fixed-wing Unmanned Aerial Vehicles (UAVs) for long endurance missions. The missions are defined by a set of Points of Interest (PoI) and UAVs should pass through them. A module to detect and identify thermals is implemented to exploit their energy and extend the flight duration, known as static soaring. A collision-free trajectory planner based on the RRT* (Optimal Rapidly-exploring Random Trees) planning algorithm is implemented. The proposed system allows applications in real time because of its low computational needs. Simulations and experiments carried out in the airfield of La Cartuja (Seville, Spain) show the performance and advantages of the proposed system.**

I. Introduction

Extending the flight duration of multiple UAVs cooperating for long endurance missions is an important problem for many applications. Recently, autonomous soaring has been proposed to extend the flight endurance of a single UAV [1] [2]. Soaring could be defined as flight in which a propulsion system is not used and the energy available in the wind environment is exploited. Although the energy can be extracted from different sources, in this paper atmospheric energy is harvested from vertical air motion, called thermals, and the approach is known as static soaring. Thermals are caused in convective boundary layers of the atmosphere.

This paper addresses long endurance cooperative missions with multiple UAVs. Multiple UAVs should perform a mission visiting a set of Point of Interest (PoI) in an environment where thermals may exist. While the mission is running each UAV should detect and identify the thermals present in the environment in order to exploit their energy and gain altitude. Cooperative missions with multiple UAV allows for faster detection and more efficient exploitation of the thermals, since each UAV transmits to the rest of the teams the location of the thermals it has identified. The guidance and control of an autonomous soaring UAV is not addressed in this paper, but we use the approach in [3]. A collision-free trajectory planner is also needed to efficiently carry out the mission. The planner has to consider constraints such as the the energy available and localization of the thermals which will influence the computation of collision-free trajectories.

A method to assign each PoI to a UAV is used. It is based on the method described in [4]. A Thermal Detector block is added to detect and identify thermals during the mission. A collision-free trajectory planner based on the RRT* (Optimal Rapidly-exploring Random Trees) planning algorithm is implemented by considering the energy of the UAV. Simulations show the performance and advantages of the proposed system. Experiments have been carried out in the airfield of La Cartuja (Seville, Spain) with the gliding fixed-wing UAV shown in Figure 1.

Fig. 1. Gliding fixed-wing UAV used in the experiments.

The paper is organized into seven sections. The state of the art is presented in Section II. The proposed system is described in Section III. Section IV presents the thermal model, the features of the environment considered and the algorithm to detect and identify the thermals. The collision-free trajectory planning algorithm is explained in Section V. The simulations and experiments performed are shown in Section VI. Finally the conclusions are detailed in Section VII.

II. State of the art

The earliest studies on soaring were based on the flight patterns of birds [5]. Soaring UAVs capable of extracting energy from the atmosphere to gain altitude in order to stay aloft have been presented in [6].

First analysis on autonomous thermal soaring were proposed in [7] [1] and [8]. Interesting results of autonomous thermal soaring are reported in [2]. In [9] the static soaring problem is applied to the Vehicle Routing Problem with

*This work was supported by the European Commission FP7 ICT Programme under the International Research Exchange Network ”Multi-UAV Cooperation for long endurance applications ” (SI-0971/2012) and and the CLEAR Project (DPI2011-28937-C02-01) funded by the Ministerio de Ciencia e Innovacion of the Spanish Government. The authors would like to thank Mr. Victor Vega for their unselfish help during the development of the experiments in the airfield of La Cartuja (Seville).

[1]J. A. Cobano, D. Alejo, G. Heredia and A. Ollero are with the Robotics, Vision and Control Group, Engineering School, University of Seville, 41092 Seville, Spain {jacobano,dalejo,guiller,aollero}@cartuja.us.es

[2]S. Sukkarieh is with the Australian Centre for Field Robotics, Faculty of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, NSW, 2006, Australia. salah@acfr.usyd.edu.au

978-1-4673-6357-0/13/$31.00 ©2013 European Union

Time Windows (VRPTW) and develops an exact solution method including preprocessing, route optimization and route validation . The total flight time minimization is achieved and a considerable increase in the level of autonomy of a soaring UAV is attained. Some works focus on the detection and identification of thermals [10] [11]. Thermal models are presented in [3] and [12].

Path planning is also addressed in other studies. A graph-based method for planning energy-efficient trajectories over a set of waypoints is presented in [13]. A method to generate a spatio-temporal map of the wind and a path planning to generate energy-gain paths based only on local observations of the wind is presented in [14]. The trajectory generation for autonomous soaring is also addressed in [15] and [16].

The coordination of multiple UAVs in order to perform a long endurance mission considering the detection of thermals and the computation of collision-free trajectories has been studied in [12]. UAVs communicate to each other the location of potential thermals in the area but the simulation shown only considers one UAV.

[17] describes a new algorithm for maximizing the flight duration of a group of UAVs using thermals located in the area. A simultaneous perturbation stochastic approximation method (SPSA) is used to detect the center of the thermal and the method treats the thermal center drift effectively. It is assumed that there exists a path planning algorithm that keeps the vehicles from colliding with each other. This work considers small unmanned powered glider, so the propulsion system or soaring can be used.

[18] presents a method for distributed mapping of the wind field. The map is discretized and a Kalman filter is used to estimate the vertical wind speed and associated covariance in each cell. Flocks of small UAVs are considered to maximize the endurance. [19] investigates the possible benefits of using a cooperating team of small UAVs to increase the probability of finding thermal lift. A collision-free trajectory planning algorithm is not implemented to optimize the search of thermals in [18] and [19]. These works and [9] consider lifetime and drift of the thermals. Moreover, The proposed system allows applications in real time because of its low computational needs.

III. System description

This section describes the proposed system for long endurance missions with multiple gliding fixed-wing UAVs. It assigns the PoI to the UAVs and computes the collision-free trajectory for each UAV if a collision is detected. During the flight, each UAV should detect and identify unknown thermals. Figure 2 shows the block diagram:

- **Local Path Planner (PP)**. Each UAV has its Local Path Planner. The path planning algorithm is called Bounded Recursive Heuristic Search method (BRHS) presented in [4] and is based on a Depth-First Search algorithm (DFS). It periodically generates a new flight plan taking into account its current location of the UAV (from the Autopilot), time, position of the remaining PoI and Thermal Points (TPs) and thermals in the space and

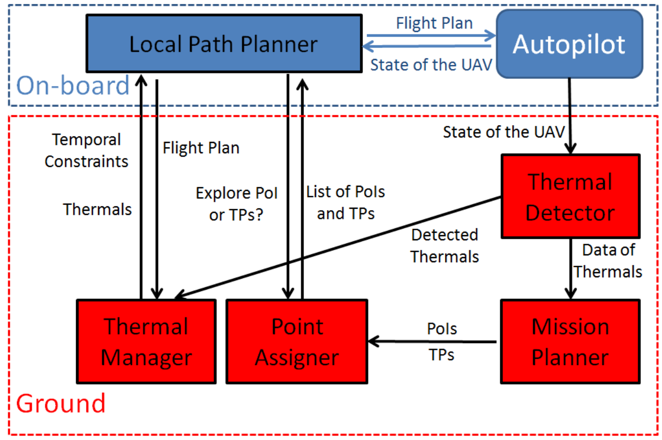

Fig. 2. Block diagram of the system.

minimum altitude of the UAV to fly. The flight plan is computed every second and the time of execution is one millisecond. Then, the PP transmits the new flight plan to the Autopilot. A flight plan can be defined by one of the following three alternatives:

1) Two waypoints: Current location and one PoI. UAV does not need a thermal to gain altitude.
2) Three waypoints: Current location, entry point and exit point of the thermal. The UAV can not reach a PoI and should access to a thermal first.
3) Four waypoints: Current location, entry point and exit point of the thermal, and a PoI. UAV could visit a PoI after gaining altitude in a thermal.

After the flight plan is computed an algorithm to generate a collision-free trajectory is executed if a collision is detected. This algorithm is based on RRT*.

- **Autopilot**. An autopilot is needed to fly a flight plan and gives information about the state of the UAV. The flight plan is received from the corresponding Local Path Planner.
- **Thermal Detector (TD)**. This block receives the state of each UAV in each instant and determines when a potential thermal is detected by one UAV from the changes of energy of the UAV. If a thermal is detected, this block sends to the Mission Planner block the estimated location of the center of the thermal and the UAV trajectory passing through the thermal. The characteristics of the new thermal are also sent to the Thermal Manager block.
- **Mission Planner (MP)**. In this block there are two types of points to visit: PoIs and TPs. Both could be considered as an exploration process: PoI to explore some places and take pictures or images, and TPs to explore potential thermals. PoIs are set by the operator and TPs are generated when the location of a potential thermal is received from TD block. The TPs are computed to ensure that the UAV passes through the center of the thermal and the trajectory is perpendicular to the first one, corresponding to the first pass. Thus,

more exact parameters can be computed to define the thermal.

- **Thermal Manager (TM)**. This block stores the information on the thermals and manages the access to them. It communicates to each Local Path Planner the existing thermals in the space and the temporal constraints to access a thermal if needed. Its inputs are the flight plans proposed by each UAV. It checks whether the flight plan proposed by each UAV to gain altitude is safe to access to a thermal. A flight plan to access to a thermal is safe when the vertical separation between two UAVs within the thermal is larger than a safety margin, d_{safety}. The outputs are temporal constraints to access to the thermal. If a flight plan is not safe, TM sends the UAV the temporal constraints to meet the vertical separation.
- **Point Assigner (PA)**. This block stores the list of remaining PoI and TPs to be visited. Initially the list shows all the PoI defined in the environment. The list is updated every time one PoI is visited or a thermal is detected. Each UAV proposes visiting a PoI or TP when generating its flight plan. PA should assign to a UAV a PoI or TP if its estimated time of arrival (ETA) to it is the lowest one so far. In order to prevent oscillatory behaviours, whenever an UAV_i proposes visiting a PoI_k or TP_k that has been already assigned to a UAV_j, the ETA of UAV_i not only should to be lower than the ETA of UAV_j but also should decrease this time with a given margin, t_{visit}. Otherwise PoI_k or TP_k continues being assigned to UAV_j.

IV. Detection and identification of thermals

This section describes how the wind map of the environment is generated and the potential thermals are detected. The parameters that define a thermal are: center of the thermal (C), vertical wind velocity (w), radius (R), maximum altitude (A) and drift of the thermal, (V_{drift}) .

The environment is divided into uniform two-dimensional cells. A value of the vertical wind velocity is assigned in each cell. Each UAV will estimate the vertical wind velocity of the center of the thermal from the changes of energy. The UAV speed is constant, so changes of altitude are taken into account.

A. Thermal model

The model of the thermal used is based on the model presented in [10]. Some modifications are performed to compute a more realistic initial wind map:

- Each value of ω is corrupted by zero-mean Gaussian noise.
- The lifetime of the thermal is considered. Thus, the vertical wind velocity distribution that defines the thermal decreases with respect to time.
- The drift of the thermal is considered.

In this paper, a test set has been generated to validate the proposed system. An algorithm to randomly generate the wind maps has been implemented. The inputs are: number of thermals at the start, lifetime of each thermal, drift of each thermal, zero-mean Gaussian noise considered, size of the environment and the cells, probability to generate new thermals during the mission and separation between the thermals. Thus, different wind maps can be generated.

B. Thermal detection algorithm

The thermal detection algorithm is implemented in the TD block. Algorithm 1 presents the thermal detection algorithm. Changes of altitude are considered to detect a potential thermal. When a change is positive (see line 4), the origin of a potential thermal (see line 5) or the continuation of the climb is considered (see line 7). If the change is negative, two cases are possible: the descent takes place after a climb (see line 11) or the descent follows (see line 18). In the first case, algorithm decides if a thermal is detected. A thermal is detected if the altitude gained during the climb, h_{gain}, is greater than $H_{threshold}$. Otherwise, a thermal is not detected and values of the thermals are initialized.

Algorithm 1 Thermal detection algorithm

1. Initialize the values $thermal_{origin}$, h_{gain}, h_{final}
2. **for** Each current altitude, h_i, **do**
3.   Compute the change of altitude $\Delta h = h_i - h_{i-1}$
4.   **if** $\Delta h > 0$ **then**
5.   **if** $thermal_{origin} = 0$ **then**
6.   Save position 3D, X_i, and update origin of a potential thermal, $thermal_{origin}$
7.   **else**
8.   Save position 3D, X_i, altitude gained, h_{gain}, and maximum altitude, h_{final}.
9.   **end if**
10.   **else**
11.   **if** $h_i < thermal_{origin}$ or $h_i < h_{final}$ **then**
12.   **if** $h_{gain} > H_{threshold}$ **then**
13.   Thermal Detected. Save data saved on the thermal: $thermal_{origin}$, h_{gain}, h_{final}
14.   Initialize the values $thermal_{origin}$, h_{gain}, h_{final}
15.   **else**
16.   Initialize the values $thermal_{origin}$, h_{gain}, h_{final}
17.   **end if**
18.   **else**
19.   Save position 3D, X_i, and update origin of a potential thermal, $thermal_{origin}$
20.   **end if**
21.   **end if**
22. **end for**

Once a thermal is detected, the parameters of the thermal are estimated. Center of the thermal, vertical wind velocity and radius are estimated as those used by Allen [10]. Drift of the thermal and more precise parameters are estimated when a UAV passes through the thermal again. Figure 3 shows how a thermal is detected when a UAV passes through it. The drift is computed when UAV passes through the thermal again by considering the center estimated in each pass.

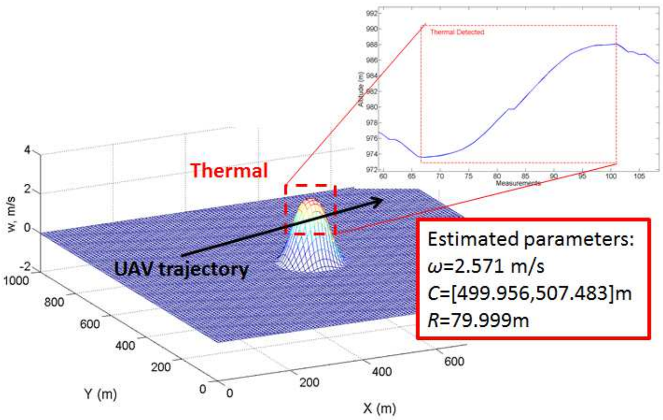

Fig. 3. Detection of a thermal when a UAV passes through it by using Algorithm 1.

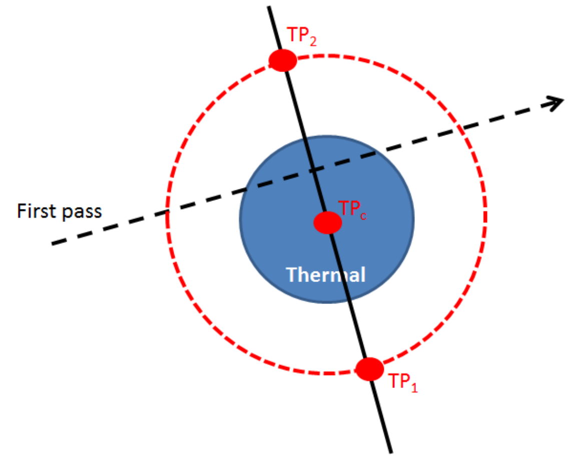

Fig. 4. Computation of the TPs to pass through a thermal again. TP_1, TP_c and TP_2 are computed from the First pass (estimated center of the thermal and UAV trajectory).

C. Computation of the TPs

The computation of the TPs is performed by the MP block from the data of the detected thermal: estimated center of the thermal and direction of the UAV trajectory passing through the thermal. Two more waypoints are computed to ensure that the UAV trajectory will pass through of the center of the thermal with a perpendicular direction to the first one (see Figure 4). The steps followed are:

1) Compute the perpendicular straight line to the first trajectory (dashed black line in Figure 4). This new straight line should pass through the center of the thermal (solid black line in Figure 4).
2) Consider a circle whose center is the estimated center of the thermal and the radius can be set (dashed red circle in Figure 4). The radius will define the distance between the TPs and the center of the thermal.
3) Compute the cross points between the new straight line and the circle. The two points computed, TP_1 and TP_2, along with the center of the thermal , TP_c, will be the set of TPs to explore the thermal.

The TPs computed, the tuple (TP_1, TP_c, TP_2), are sent to the PA block and each UAV can apply for passing through PoI or TPs to improve the computation of the parameters of a thermal.

V. Collision-free trajectory planning algorithm

Whenever multi-UAV systems are used in any application, they should maintain as much as possible a minimum separation among all UAVs for safety. In our system, the separation inside thermals is carried out by the TM block. This system will not let a UAV enter in a thermal if a vertical separation is violated and will also ensure that the horizontal and vertical separation should be satisfied outside the thermals.

A collision-free trajectory planning algorithm is executed when a collision between UAVs is detected. It is based on a RRT* planning algorithm. RRT* makes two main modifications to the original RRT planning algorithm [20].

The detection algorithm is based on cylinders. Each UAV is defined by a cylinder. A collision is detected if the horizontal separation, euclidean distance in the XY plane, or the vertical separation are violated. This technique presents as advantages the low execution time and the need of few parameters to describe the system.

Several improvements of the RRT* algorithm are presented in [21]. The algorithm is adapted to our problem considering the Local Bias to decrease the cost and the node rejection to increase efficiency. With regards to the Local Bias, it would be desirable to make a thermal bias in the first step of the algorithm. Thus, the tree explores the space of the thermals that are close to the segment that unites the start and goal configurations.

A new model based on the one described in [22] has been used in order to generate feasible trajectories. The main modifications to the original model are the constant descent rate, assumption of constant airspeed and the addition of the vertical wind velocity that is retrieved from the wind map. The configuration space of this model is composed by three spatial coordinates (x, y, z) and the heading θ. However, new samples are generated randomly in this space, but coordinates z and θ are calculated in the extend procedure of RRT* taking into account the current heading and the wind map. This ensures that the final trajectories are flyable.

The configuration space is expanded when solving one conflict. For example, one state is defined by $(x_1, y_1, z_1, \theta_1, x_2, y_2, z_2, \theta_2, t)$ with two UAVS. Besides, the time should be included in order to check possible collisions.

As indicated in [20] the RRT* algorithm is only capable to minimize the length of the trajectories. As long as this algorithm is only applied outside the thermals it is a very fair approximation of energy-efficient trajectories.

Figure 5 shows two collision-free trajectories computed to solve a collision detected between two UAVs. Both change their initial trajectory to avoid the collision. The trajectories generated by RRT* should be then smoothed before sending them to the autopilot. The algorithm has been implemented in C++ using the Open Motion Planning Library using the contrib package where an implementation of the RRT* algorithm is available. More than a hundred test cases have been used as bench-test and the average run time was of $2.53s$ with a standard deviation of $1.5s$. This allows the algorithm to be used in the system for real-time applications.

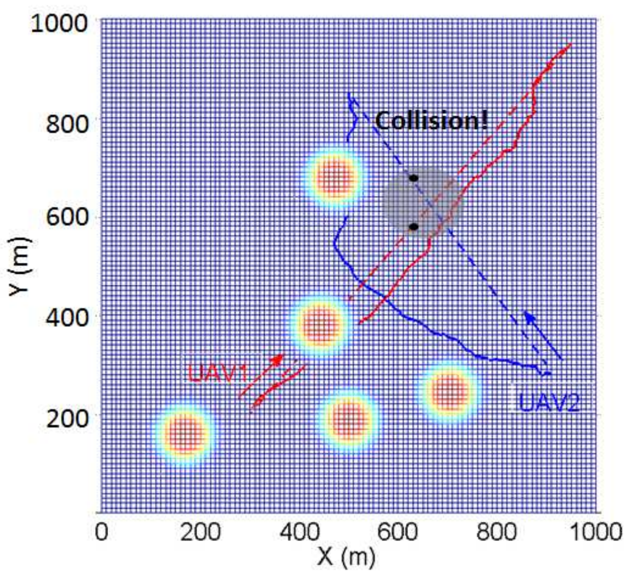

Fig. 5. UAV trajectories computed from the RRT* planning algorithm to solve a collision: solid line is the solution trajectory and dashed line is the initial trajectory.

VI. Simulations and Experiments

A. Simulations

Simulations in different scenarios generated randomly have been performed to show the behaviour of the system and how thermals are identified with cooperative gliding fixed-wing UAV. The gliding UAV model used is based on the model proposed in [22]. The algorithms have been run in a PC with an Intel©Core ™i5-2410M @2.3 GHz processor with 4GB of RAM and 500 GB of hard disk with Windows 7 64 bits installed.

First, a simulation with one UAV is shown. The exploration mission is given by fifty PoI. Initially, twelve thermals are created (thermals 1-12). Five thermals more are created during the mission in different times (thermals N1-N5) (see Figure 6). The UAV does not have any information about their location or strength, so the thermals are unknown to the system. Table I shows the configuration parameters considered.

Figure 6 shows the UAV trajectory and the wind map at the instant $t = 600$ seconds. The mission lasts 24 minutes and 0.016 seconds (from 10:13:24.772 to 10:37:24.788). UAV detects and identifies seven thermals in different times and it exploits one of them. Note that the drift of the thermals.

TABLE I

Configuration parameters used in the simulations.

System block	Parameter	Value
Wind Map	Cell Size	$10m$
	Drift	$0.5m/s$
	Lifetime	$25min$
	Noise std	$0.3m/s$
	Windspeed	$3m/s$
System Configuration	Vertical separation	$25m$
	Horizontal separation	$50m$
	d_{safety}	$25m$
	t_{visit}	$5s$
UAV Model	Descent angle	$0.08rad$
	Windspeed	$13.89m/s$
	Climb rate	$1.9m/s$
	Min alt.	$80m$

Table II and Figure 7 show the information on the detection of the thermals. One thermal is identified when the UAV passes through it twice. After identifying the sixth thermal, the UAV exploits it and gains approximately 140 meters (see Figure 7).

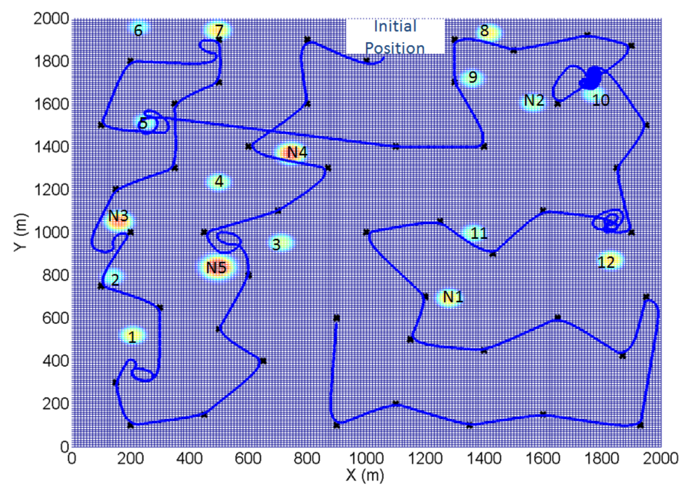

Fig. 6. UAV trajectory to pass through fifty PoI (black points). Thermals 1-12 are created at the start and N1-N5 are generated during the mission. Wind map considered corresponding with t=600 seconds.

TABLE II

Detection and identification of thermals

Thermal	Time (s)	Elapsed time (s)
4	10:15:44.899	140.127
1	10:17:58.299	273.527
N3	10:19:09.420	236.468
7	10:20:53.300	448.528
5	10:22:20.337	535.565
10	10:27:18.412	833.640
12	10:31:34.329	1089.557

The next study analyzes the behaviour of the system with several cooperative UAVs. We consider the same wind map generated randomly. Ten simulations are performed in each case by changing the initial position of each UAV. Table III shows the results obtained and Figure 8 presents one of the simulations to show how the exploration is carried out with several UAVs.

Simulations performed demonstrate the performance of the system. It is important to highlight the following characteristics:

- RRT* planning algorithm is executed when a collision is detected. This algorithm ensures the safety of the system.
- Mission Planner block computes the corresponding TPs to pass through a detected thermal again.
- UAVs take into account the drift estimated to exploit the thermals detected.
- Using cooperative UAVs the mission time is reduced.
- Benefits of using cooperative UAVs to detect thermals are shown.
- The flight duration is extended to carry out the mission while thermals exist in the environment.
- The system could be applied in real time.

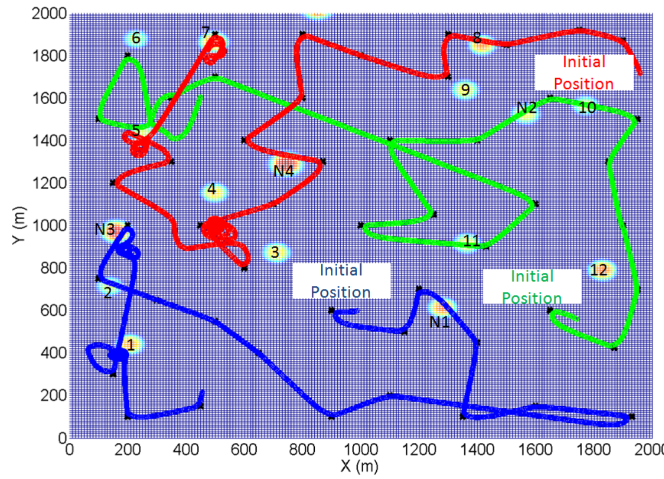

Fig. 8. Mission with three UAVs: UAV trajectories to pass through fifty PoI (black points). Thermals 1-12 are created at the start and N1-N4 are generated during the mission. Wind map considered corresponding with t=450 seconds.

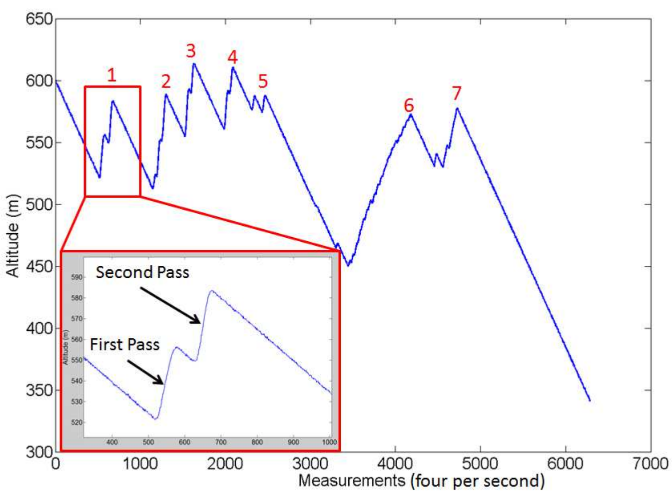

Fig. 7. Vertical profile of the UAV flight. Seven thermals are identified and the UAV passes through each thermal twice to estimate its parameters.

TABLE III
DETECTION AND IDENTIFICATION OF THERMALS CONSIDERING TEN SIMULATIONS.

UAVs	Elapsed Time (s)	Thermals detected
1	1345.31 $\pm$ 57.20	4.63 $\pm$ 1.51
2	759.92 $\pm$ 42.02	4.81 $\pm$ 1.68
3	498.844 $\pm$ 36.01	5.75 $\pm$ 1.92
4	466.24 $\pm$ 44.03	6.89 $\pm$ 0.84
5	344.25 $\pm$ 31.63	5.73 $\pm$ 1.34

B. Experiments

Experiments have been performed with two gliding fixed-wing UAVs in the airfield of La Cartuja (Seville) in order to test the behaviour of the whole system in real time. One of them is a real gliding fixed-wing UAV and the other is simulated. The thermals are emulated; that is, when the real UAV accesses a simulated thermal it simulates the gaining of energy by using the propulsion system to gain altitude. The rest of the flight is carried out without propulsion. The following variables of the UAV were set: gliding angle (0.11rad), airspeed (13.89m/s) and climb rate (1.9m/s) within a thermal.

Figure 9 shows a part of the flight in 2D. The real gliding fixed-wing UAV should pass through PoI1 and PoI3. The simulated fixed-wing UAV should pass through PoI2. The experiment, flight and ground station, can be seen in the video attached. Finally, Figure 10 presents the trajectories in the airfield of La Cartuja (Seville).

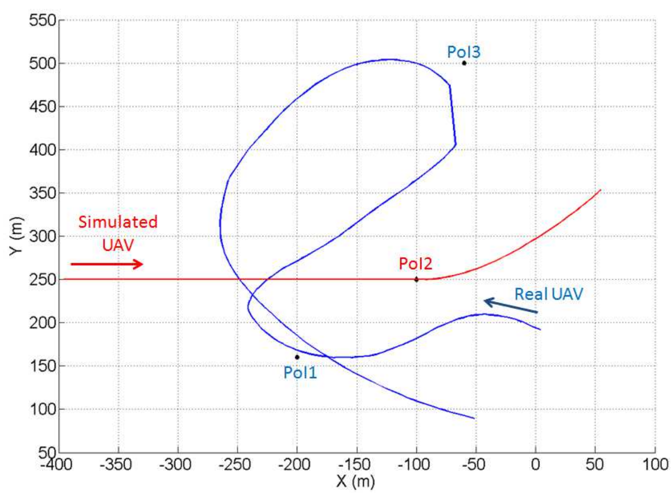

Fig. 9. Real and simulated flight in 2D to explore the environment in the airfield of La Cartuja (Seville). The system assigns the PoI in real-time: the real UAV passes through PoI1 and PoI3, and the simulated UAV passes through PoI2. .

VII. Conclusion

This paper considers long endurance cooperative missions with multiple gliding fixed-wing UAVs. A new system based

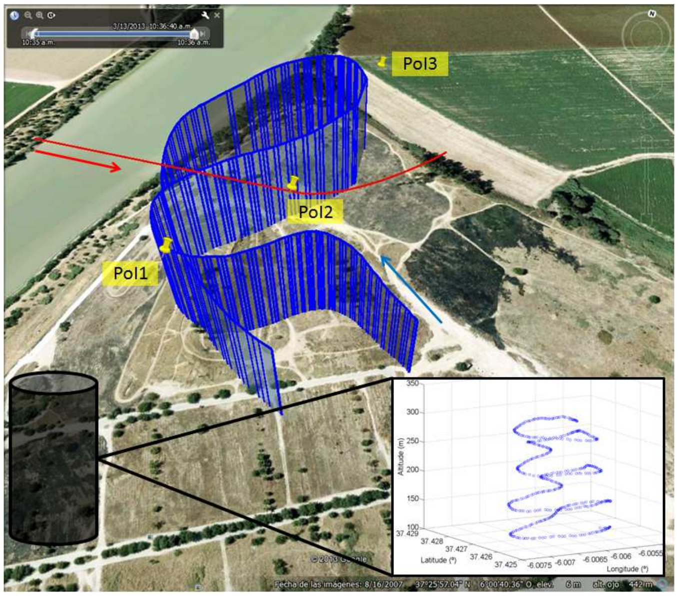

Fig. 10. UAV Trajectories and location of thermal represented in the airfield of La Cartuja (Seville): real gliding fixed-wing UAV (blue), simulated gliding fixed-wing UAV (red) and thermal (black cylinder). The real UAV trajectory described inside of the thermal is shown (blue circles).

on the method described in [4] is proposed to extend the flight duration by harvesting energy that comes from thermals. The goal is to explore an environment without landing, decreasing the time to perform the mission and increasing the probability of detecting unknown thermals by using multiple UAVs. Simulations to study the behaviour of the system and real experiments have been performed.

New blocks and functionalities are added to identify unknown thermals and resolve collisions detected between UAVs. An algorithm is implemented to detect and identify unknown thermals to exploit them. First, a potential thermal is detected from the changes of altitude of a UAV. Then an algorithm computes two extra waypoints to ensure that a UAV will pass through of the center of the thermal with a perpendicular direction to the first one. The thermal parameters are estimated after the second pass and these parameters are sent to the TM block.

Moreover, a collision-free trajectory planning algorithm based on the RRT* is implemented to solve the collisions detected between UAVs. The RRT* planning algorithm presented in [21] has been adapted to this problem by considering the energy. This algorithm is an important difference with respect to the works presented on multiple UAV in autonomous soaring [12] [17] [18] [19].

Simulations performed demonstrate the utility of the system by adapting to changes in the environment and maximizing the endurance. Experimentation on a real platform shows that it can be applied for real time applications because of its low computational needs. This latter presents an important contribution with respect to [18] and [19].

Future work will be focused on the mapping of 3D wind field and generation of energy-efficient trajectories by considering this 3D map of wind.

References

[1] M. J. Allen, "Guidance and control of an autonomous soaring uav," *NASA TM-214611*, February 2007.

[2] D. J. Edwards, "Implementation details and flight test results of an autonomous soaring controller," in *AIAA Guidance, Navigation, and Control Conference, AIAA, Paper 2008-7244*, Reston, VA, August 2008.

[3] M. J. Allen, "Updraft model for development of autonomous soaring uninhabited air vehicles," in *44 th AIAA Aerospace Sciences Meeting and Exhibit, AIAA 2006-1510*, 2006, pp. 9–12.

[4] J. A. Cobano, D. Alejo, S. Vera, G. Heredia, and A. Ollero, "Multiple gliding uav coordination for static soaring in real time applications," in *IEEE International Conference on Robotics and Automation (ICRA2013)*, Karlsruhe, Germany, 6-10 May 2013, pp. 782–787.

[5] L. Rayleigh, "The soaring of birds," *Nature*, no. 27, pp. 534–535, 1883.

[6] C. K. Patel and I. M. Kroo, "Theorical and experimental investigation of energy extraction from atmospheric turbulence," in *Proc. 26th Congress of International Council of the Aeronautical Sciences*, Anchorage, Alaska, USA, 14-19 September 2008.

[7] J. Wharington, "Autonomous control of soaring aircraft by reinforcement learning," Ph.D. dissertation, Royal Melbourne Institute of Technology, Melbourne, Australia, 1998.

[8] M. Hazard, "Unscented kalman filter for thermal parameter identification," in *AIAA Region II Student Conference, AIAA*, Washington, DC, USA, 2009.

[9] N. Kahveci, P. Ioannou, and D. Mirmirani, "Optimal static soaring of uavs using vehicle routing with time windows," in *AIAA Aerospace Sciences Meeting and Exhibit, AIAA-2007-158, AIAA*, Washington, DC, USA, 2007.

[10] M. J. Allen, "Autonomous soaring for improved endurance of a small uninhabited air vehicle," in *Proceedings of the 43rd Aerospace Sciences Meeting, AIAA*, 2005.

[11] D. Metzger and J. Hedrick, "Optimal flight paths for soaring flight," in *Proceedings of the 2nd International Symposium on the Technology and Science of Low Speed and Motorless Flight*, 1974.

[12] A. Klesh, P. Kabamba, and A. Girard, "Optimal cooperative thermalling of unmanned aerial vehicles," *Optimization and Cooperative Control Strategies*, vol. 381, pp. 355–369, 2009.

[13] A. Chakrabarty and J. W. Langelaan, "Energy-based long-range path planning for soaring-capable unmanned aerial vehicles," *Journal of Guidance, Control and Dynamics*, vol. 34, no. 4, pp. 1002–1015, 2011.

[14] N. R. J. Lawrance and S. Sukkarieh, "Path planning for autonomous soaring flight in dynamic wind fields," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA2011*, Shanghai, China, 2011.

[15] W. B. Kagabo and J. R. Kolodziej, "Trajectory determination for energy efficient autonomous soaring," in *2011 American Control Conference*, San Francisco, CA, USA, June 29 - July 01 2011, pp. 4655–4660.

[16] J. H. A. Clarke and W. H. Chen, "Trajectory generation for autonomous soaring uas," *International Journal of Automation and Computing*, pp. 248–256, June 2012.

[17] G. O. Antal C. and L. S., "Adaptive autonomous soaring of multiple uavs using simultaneous perturbation stochastic approximation," in *Proceedings of the 49th IEEE Conference on Decision and Control*, Atlanta, GA, USA, 2010.

[18] N. T. Depenbusch and J. W. Langelaan, "Coordinated mapping and exploration for autonomous soaring," in *AIAA Infotech@Aerospace Conference*, St. Louis, Missouri, USA, Marchl 29-31 2011.

[19] K. Andersson, I. Kaminer, K. Jones, V. Dobrokhodov, and D. Lee, "Cooperating uavs using thermal lift to extend endurance," in *AIAA Unmanned Unlimited Conference*, Seatle, Washington, USA, April 6-9 2009.

[20] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, 2011.

[21] B. Akgun and M. Stilman, "Sampling heuristics for optimal motion planning in high dimensions." in *International Conference on Intelligent Robots and Systems (IROS2011)*, San Francisco, California (USA), 25-30 September 2011, pp. 2640 – 2645.

[22] T. W. McLain, R. W. Beard, and A. Beard, "Coordination variables, coordination functions, and cooperative timing missions," 2003.