Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-8 (2019) 283–288

A spatially wind aware quadcopter (UAV) path planning approach

Georgios A. Thanellas*,Vassilis C. Moulianitis**
Nikos A. Aspragathos***

** Department of Mechanical Engineering and Aeronautics, University of Patras, Greece, (e-mail: geothanellas@gmail.com).*
*** Department of Product and Systems Design Engineering, University of Aegean, Greece, (e-mail: moulianitis@syros.aegean.gr)*
**** Department of Mechanical Engineering and Aeronautics, University of Patras, Greece, (e-mail: asprag@mech.upatras.gr)*

Abstract: Wind aware path planning is essential towards the development of autonomous quadcopters (UAVs) which will be able to fly efficiently in outdoor environments. In this paper, the wind formation and quadcopter kinematics are presented so as to provide cost functions for A* based on distance and wind information. Furthermore, a collision checking method is proposed, which is incorporated to a flight scenario in an outdoor terrain under various wind formations. A computational comparison of a uniform graph and a PRM based graph is finally evaluated. Experiments on artificial wind fields, through illustrative simulations, highlight the benefit of wind aware A* searching algorithm.

Keywords: quadcopter, wind, path planning, collision avoidance, minimum time, minimum distance.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) were developed initially for military applications, however they currently became extremely popular in civil applications such as agriculture, environmental protection, aerial photography, public safety and traffic flow control (Mohammed et al, 2014). Moulianitis et al. (2018) concluded that UAVs constitute a cost-effective solution early forest fire detection compared with other state of the art relevant mechatronic systems. Among commercially available UAVs (fixed wing, single rotor and multirotor), multi-rotors offer greater flexibility in control and maneuverability with vertical take-off and landing capabilities. Even though, multi-rotors have a limited flight time, whose energy is spent mainly on fighting gravity and stabilizing in the air. Lawrance et al. (2014) suggested two approaches for the energy autonomy problem, either aerial recharging or ongoing harvesting of energy from the vehicle's surrounding environment. The latter approach can be proved very efficient in wind path planning which could provide paths with reduced energy consumption as well as shorter flight time.

The concept of exploiting the energy from the environmental forces has been applied for marine as well as aerial vehicles (Liu and Sukhatme, 2018; Mahmoud.Zadeh et al. (2016); Kularatne et al., 2016; Techy and Woolsey, 2009; Al-Sabban et al., 2012). Path planning based on Markov Decision processes under ocean currents for underwater robots or under winds for UAV have been introduced (Liu and Sukhatme, 2018; Techy and Woolsey, 2009; Al-Sabban et al., 2012). Mahmoud.Zadeh et al. (2016) implemented a Differential Evolution path planner for navigation in ocean environments considering obstacles. Kularatne et al. (2016) extracted cost functions based on time and energy to be minimized along the determined path.

According to Langelaan (2018), the vertical air motion (thermal instabilities, orographic lift or wave), the spatial wind gradients such as horizontal shear layers and the temporal gradients like gusts are the main sources of energy available in the atmosphere. Gusts have short duration and are stochastic in nature, while the vertical air motion have long duration, while the main source of the wind field is the wind forecasting services based on Numerical Weather Predictions (NWP), (*Windfinder.com, 2018).*

Usually, UAVs are largely limited to primitively follow user-defined waypoints in outdoor environments, which are not always energy efficient and time optimal under wind. In this paper, we propose a path planning procedure, considering a collision avoidance technique using A* with new alternative cost functions, which are compared with distance cost functions of standard A*. To reduce computational searching cost, we introduce a graph, based on surface veil, using either a uniform or a PRM node distribution in a simulated terrain.

The remainder of this paper is organized as follows: Section 2 presents the formation of wind as well as the quadcopter kinematics. Section 3 describes the form of a veil-graph in an unstructured terrain as well as a collision avoidance technique in a triangular mesh. In addition, two path planning approaches are introduced and evaluated with a comparison metric based on time. Section 4 presents the experimental evaluation of path planning approaches. Finally, section 5 presents the main conclusions and future work.

10.1016/j.ifacol.2019.08.084

2. PROBLEM FORMULATION

Path planning efficiency in UAVs depends from distance criteria, especially under obstacle existence, while in outdoor environments, time index can improve flight endurance under windy conditions regard to the limitation of the possible on board (fuel/battery) which can be carried by the UAV. In the following a brief description of an outdoor windy environment as well as a quadcopter kinematics model are presented.

2.1 Windy environment

The success of the efficient wind navigation model depends on the density of measurements and the wind prediction map accuracy. The wind forecast maps provided present high uncertainty and the wind distribution is really stochastic, however a known wind distribution is considered in this paper to show the effectiveness of the proposed method. In real conditions navigation the stochastic nature should be considered or on-line wind measurement will be provided. In this paper, the wind field is considered as a two-dimensional time-invariant vector field, representing velocity at each point of a map:

$$\vec{u}_w(x, y, z) = \langle v_{wx}, v_{wy}, 0 \rangle, \quad (1)$$

where $u_{wx} = u_w cos\theta_w$ and $u_{wy} = u_w sin\theta_w$ define the wind components of $\vec{v}_w$ in x-axis and y-axis respectively referenced to the inertial frame. Fig 1 illustrates a wind map with a mean magnitude $(u_w = 5m/s)$ and direction (θ_w) in each point (x, y, z) of the map. Various formations of artificial wind fields based on wind data formation given by weather agencies, are presented in section 4.

Fig. 1. Snapshot of a wind map (Windfinder.com, 2018).

2.2 Quadcopter kinematics

The quadcopter is an underactuated system since four actuators are used to control 6 DoF of its body. In this paper, the rotational DoF are excluded and the state of the quadcopter is expressed by its position coordinates. The path planning problem is simplified by considering the quadcopter as a free flying object with velocity $\vec{u}_d$:

$$\vec{u}_d = u_{dx}\hat{\imath} + u_{dy}\hat{\jmath} + u_{dz}\hat{k}, \quad (2)$$

and given the wind velocity $\vec{v}_w$ with respect to the ground (fig 2) the total ground velocity of the UAV is the following:

$$\vec{u}_t = \vec{u}_d + \vec{u}_w \Rightarrow \begin{bmatrix} u_{dx} \\ u_{dy} \\ u_{dz} \end{bmatrix} + \begin{bmatrix} u_{wx} \\ u_{wy} \\ 0 \end{bmatrix}. \quad (3)$$

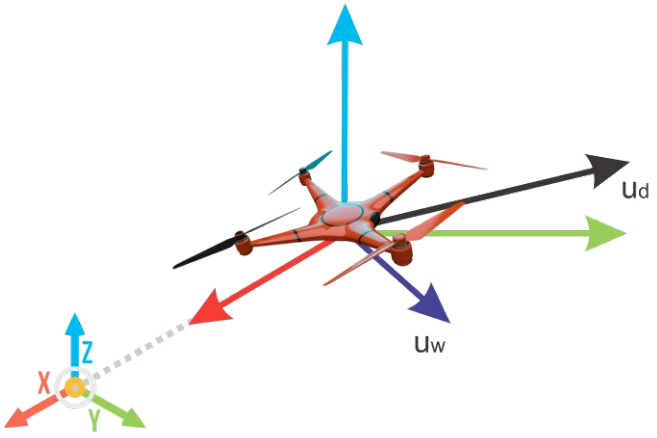

Fig 2. Analysis of UAV related velocities

$\vec{u}_t$ is used to determine the flight time from a start position to a goal position taking into account the wind field as well as obstacle avoidance. The UAV speed limit is considered since the path duration is calculated with the maximum velocity. The acceleration limit is not considered, however this could be taken into account in a smoothing stage of the determined polyline type path.

3. PATH PLANNING

In this work, an outdoor unstructured terrain is modelled with triangular meshes whose edges (defined by vertices) are stored in a circular list that is traversed in counterclockwise order with respect to the outward normal vector of the face. A grid is adapted to the terrain surface like a veil in order to minimize the computational cost of path planning among nodes. In the aftermath, a collision avoidance technique is employed based on computational geometry between path and triangles. For the path planning, a graph searching algorithm A* is used, with cost functions based either on wind deviation from UAV speed direction or distance criteria. The evaluation of their cost functions relies on the determined total flight time, taken into account the wind velocity.

Graph Search-based Trajectory Planning

A discrete graph G=(V,E) is used to represent the workspace in a flight area (W). The number of vertices (nodes) are either centroids of cells in a uniform square discretization or randomly generated nodes in workspace, based on PRM (Kavraki et. al, 1994). The latter one offers an optimization in computational cost due to reduced nodes in 2D. To keep the computational cost low in 3D, the concept of veil above the surface area is adopted, discarding multiple vertices in z-axis. Vertices which lie inside obstacles are elevated above them in

order to keep as much as possible edges between neighboring nodes. Fig.3 shows a 3D view map of a triangular surface plot, which contains a veil with uniform and random generated nodes.

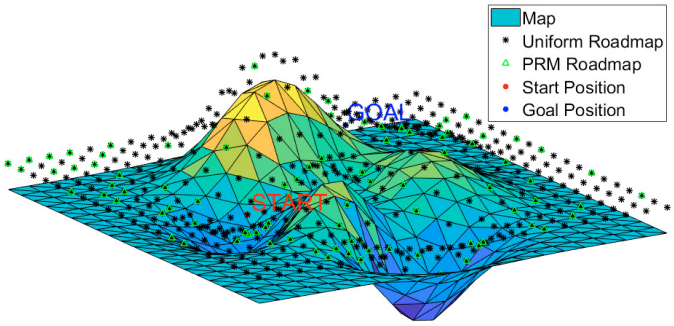

Fig. 3 3D view of simulated outdoor area includes nodes above the surface. Among them there are a start node and a goal node.

A wind aware A* searching algorithm is proposed to determine the path between the start and the goal node. At each iteration of its main loop, A* needs to determine which of its paths to be extended. Based on the cost $g(n)$ between start and current node and an estimation of the cost $h(n)$ between current node and goal node, A* selects the path that minimizes $c(n) = g(n) + h(n)$. Each step of A* includes the following:

Possible States ($\boldsymbol{s}$): the finite set of possible states which are the graph nodes. Each state is described by the Cartesian coordinates and is written as $s_i = [x_i \;\; y_i \;\; z_i]^T$, where x_i, y_i and z_i denote the cartesian coordinates of the UAV position in the graph.

Possible Actions ($\boldsymbol{a_s}$): It is the finite set of actions, in terms of motion, available from state s_i. Depending on graph node density, a spatial radius r_s is defined, which ensures connectivity of s_i with its neighbors s_j, where $i \neq j$. A node s_j is considered neighbor of s_i, if and only if:

$$(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2 \leq r_s \quad (4)$$

So, actions represent all possible directions along each edge of UAV to neighboring states, which are interior of a predefined sphere with r_s radius and are dependent from cost functions and obstacle presence.

Collision Avoidance

Valid edges of the graph are considered those which don't collide with obstacles, ensuring a collision free route between two neighboring nodes. Given that, the simulated terrain consists of triangle meshes and nodes of the graph, we are looking for edges which connect nodes and are always above the surface. To achieve that, we keep a subset of terrain triangles $\vec{v}_l$ which lie between the nodes s_i and s_j, as follows:

$$\begin{cases} s_{i,x} \leq \vec{v}_{l,x} \leq s_{j,x} \\ s_{i,y} \leq \vec{v}_{l,y} \leq s_{j,y} \\ s_{i,z} \leq \vec{v}_{l,z} \leq s_{j,z} \end{cases} \quad (5)$$

An edge is given by:

$$\vec{e} = \vec{s_i} + \lambda \vec{b}, \quad (6)$$

where λ is a scalar quantity and $\vec{b}$ is the unit vector of the line defined by the nodes $[s_i, s_j]$ given by:

$$\vec{b} = \frac{\vec{s}_j - \vec{s}_i}{\|\vec{s}_j - \vec{s}_i\|}, \quad (7)$$

The collision detection method rejects edges which have a common point with a triangle of the mesh surface. This achieved by comparing the sum of areas of an intermediate point $\vec{p}_k$ of the edge $\vec{e}$ and the vertices of triangles $\vec{v}_{l,(1,2,3)}$ with the area $A_{\vec{v}_l}$ formed by the vertices of triangle $\vec{v}_l$. Therefore, if:

$$\begin{cases} A_{\vec{v}_l} = A^{\vec{p}_k}_{\vec{v}_{l,12}} + A^{\vec{p}_k}_{\vec{v}_{l,13}} + A^{\vec{p}_k}_{\vec{v}_{l,23}}, & Collision \\ A_{\vec{v}_l} \neq A^{\vec{p}_k}_{\vec{v}_{l,12}} + A^{\vec{p}_k}_{\vec{v}_{l,13}} + A^{\vec{p}_k}_{\vec{v}_{l,23}}, & No\ Collision \end{cases} \quad (8)$$

where $A^{\vec{p}_k}_{\vec{v}_{l,12}}, A^{\vec{p}_k}_{\vec{v}_{l,13}}, A^{\vec{p}_k}_{\vec{v}_{l,23}}$ are the areas, formed by $(\vec{p}_k, \vec{v}_1, \vec{v}_2)$, $(\vec{p}_k, \vec{v}_1, \vec{v}_3)$ and $(\vec{p}_k, \vec{v}_2, \vec{v}_3)$ respectively. Fig 4 illustrates a point $\vec{p}_k$ of edge vector $\vec{e}$, which forms 3 triangles with vertices $\{v_1, v_2, v_3\}$ of an l triangle between nodes s_i and s_j. According to equation (8), the vector segment between s_i and $\vec{p}_k$ is collision free in contrast to $\vec{p}_{k+1}$, whose sum of areas is the area of triangle.

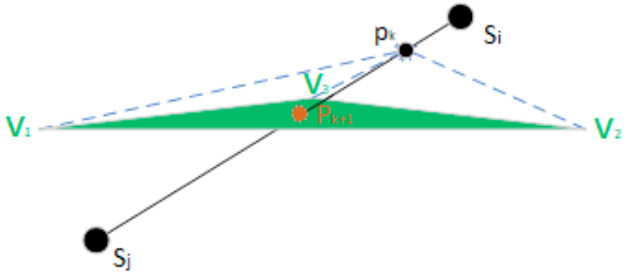

Fig 4. The collision checking method based on computational geometry.

Cost functions

The formulation of A* cost functions is based on edge attributes among nodes. An edge has orientation and Euclidean distance values, while the velocity $\vec{u}_d$ of the UAV coincides with the orientation of path. In this work it is assumed that the winds are instant time-invariant and the cost functions are the following:

Distance minimizing cost: the Euclidean distance is determined between states and rewards the nodes that minimize the path length. The UAV moves to the nearest nodes, until it reaches its goal (s_{goal}). In this way, the calculation of path cost in each A* iteration, defines the state s_j with distance (d) from state s_i, resulting a cumulative cost $g_d(n)$ to current node (n) and computes its remaining distance to target by the heuristic cost $h_d(n)$. The cost functions in each iteration are the following:

$$\begin{cases} g_d(n) = \sum_{i=2}^{n} d(s_i, s_{i-1}) \\ h_d(n) = d(s_n, s_{\text{goal}}) \end{cases}, (9)$$

The above cost functions of A* result a total cost $c_d(n)$, based on Euclidean distance without taking into account the wind field.

Minimum angle cost: This cost formula computes the minimum angle between wind vector and UAV speed vector and keeps the node with minimum angle, resulting a cost $g_\theta(n)$. For example, in Fig 5, from the current position of UAV in state s_i the state s_{j+1} is chosen which has the smallest angle between the UAV direction of motion and the wind direction.

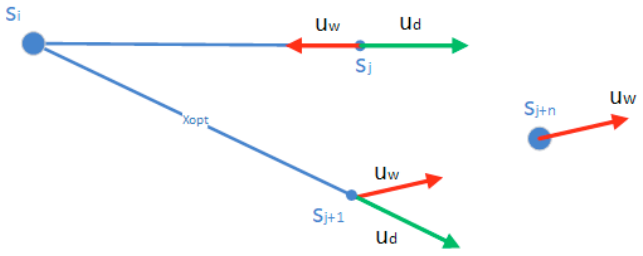

Fig 5. Procedure for determination of optimal path $\boldsymbol{x_{opt}}$ among states.

Regarding the heuristic cost, we compute the mean angle of wind vectors between the current and the target node, resulting a cost $h_\theta(n)$. This assumption for the computation of heuristic function deals with wind direction fluctuation. These cost functions are given by:

$$\begin{cases} g_\theta(n) = \sum_{i=2}^{n} \arctan\left(\frac{\left\|\vec{u}_{\text{d},i-1} \times \vec{u}_{\text{w,i}}\right\|}{\vec{u}_{\text{d},i-1} \cdot \vec{u}_{\text{w,i}}}\right) \\ h_\theta(n) = \frac{\sum_{i=n}^{goal} \theta_i}{\sum_{i=n}^{goal} \text{i}} \end{cases}, \quad (10)$$

where u_d expresses UAV velocity that coincides with path orientation and $\vec{u}_w$ is the wind vector in state s_j. The above cost functions of A* result a minimum angle total cost $c_\theta(n)$, which ensures an optimal edge x_{opt}, which will be traversed by the UAV with maximum possible total velocity $(\vec{u}_t)$, resulting to minimum time.

Hybrid angle cost: This cost formula computes the angle between wind and UAV with $g_\theta(n)$,as minimum angle cost function does and Euclidean distance between current node and target with $h_d(\text{n})$, as distance minimizing cost does. In order to homogenize the quantities of $g_\theta(n)$ and $h_d(\text{n})$ both values are normalized in the range [0,1]:

$$\text{c(n)} = w_1 \cdot \frac{g_\theta(n)}{\pi} + w_2 \cdot \frac{h_d(\text{n})}{\min(h_d(\text{n}), d(s_{start}, s_{\text{goal}}))} \quad (11)$$

where $\text{w}_\text{i}, \text{i} = 1,2$ are the weights which determine the priority of A* in convergence; accordingly, to angle and distance respectively, which synthesize the minimum cost of path between the start and goal position.

Metric of costs

In order to compare A* cost function variants, we examine the execution time of path. For the computation of local flight speed, it is assumed that the drone follows an edge without deviations:

$$u_t^i = u_d^i + u_w^i \cdot cos\varphi \quad (12)$$

Where φ is the angle between the drone velocity and the wind velocity, u_d^i is the local drone speed in each segment and $u_w^i \cdot cos\varphi$ is the wind component along path's segment. $u_w^i \cdot sin\varphi$ component of wind is assumed that it is counterbalanced by the controller. So, the quadcopter will navigate along each segment, resulting a flight time under wind impact, regardless the cost function, as follows:

$$\text{time} = \sum_{i=2}^{n} \frac{d(s_i, s_{i-1})}{u_t^{i-1}}, (13)$$

4. APPLICATION AND DISCUSSION OF RESULTS

In this section artificial vector fields are used to simulate wind fields, which are applied in a discrete simulated flight area x∈[-2,2] and y∈[-2,2], with measurement units expressed in mesh resolution (mr), in which are juxtaposed the two A* s with the proposed cost functions compared with distance based A*. In each simulated case, the execution time is determined, considering that the drone speed $\|u_d\| = 2mr/s$ and the average speed of wind $\|u_w\| = 1mr/s$ are predefined. Finally, uniform and random graph based on PRM time duration of flight are compared.

Three cases of simulated wind fields are considered:

Case 1, Tailwinds. In this field, the wind has constant direction with varied magnitude:

$$\overrightarrow{u_w}(\text{x}, \text{y}) = \frac{(1 - \text{y}^2)\hat{\imath}}{\sqrt{(1 - \text{y}^2)^2}} \quad \forall \text{ y} \in |\text{y}| \leq 1, (14)$$

The start and end position of the UAV are located so that the field will always provide a tailwind component during the motion.

Case 2, Headwinds and Tailwinds. This field provide both headwinds and tailwinds with different magnitudes during the motion of the UAV from the start to end location.

$$\overrightarrow{u_w}(\text{x}, \text{y}) = \frac{(x - 2)\hat{\imath} + (\text{x} + 1)\hat{\jmath}}{\sqrt{(x - 2)^2 + (\text{x} + 1)^2}}, (15)$$

Case 3, Rotated wind field. It is used to validate the algorithm:

$$\overrightarrow{u_w}(x, y) = \frac{-y\hat{i} + x\hat{j}}{\sqrt{y^2 + x^2}}, (16)$$

For each case, the simulation results are shown in Fig 6 (a-c). Each simulation case includes two paths. The first one (red) is extracted with minimum distance cost and the second (green) is extracted with minimum angle cost, proving a wind optimal path.

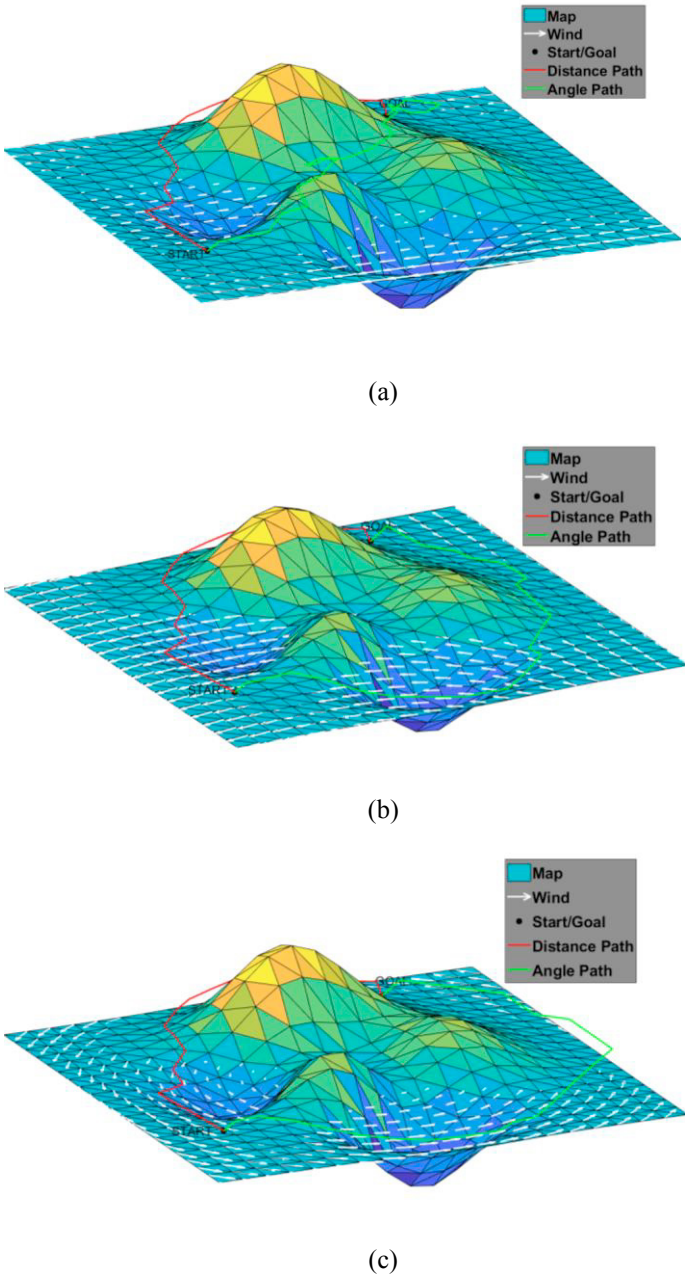

Fig. 6 Simulation of path planning with distance and minimum angle criteria in: (a) Case 1. (b) Case 3. (c) Case 3.

The durations of distance and angle paths, considering the average wind speed are shown in Table 1. Using time equation, it is shown in table 1 that in the first and third case the minimum angle flight is executed in shorter time than this of minimum distance, unless the wind is completely reversed to drone's flight path.

Table 1. Simulation Results

	Minimum Distance	Minimum angle Wind
Case 1	3.11s	2.30s
Case 2	3.06s	3.63s
Case 3	4.16s	2.51s

In Table 2, the flight time with the hybrid angle cost function is evaluated with under various w_1 and w_2 values. Experiments reveal that the increase of w_2 leads to considerably shorter flight times for 1st and the 2nd case. If $w_2 = 1$ the minimum angle wind function can be trapped in local optimum since it does not take into account the distance to the goal position, as happened in case 2, while the hybrid angle function can find the best path, combining both distance and wind field. Fig. 7 shows a top view of map with high and low w_1 and w_2 respectively and vice versa. Also, the execution times of table 2 for 1st and 3rd case are examined with respect to optimal weights of 2nd case and show that the hybrid angle A* performs always better than the minimum distance A*. Also, hybrid angle A* presents better results compared to the minimum angle A*, which is unable to converge to a good position with opposed wind fields, while in the case of goal targeted wind the minimum angle A* performs slightly better than the hybrid one, as shown in table 2.

Table 2. Hybrid angle A* weights

Case	Distance weight	Angle Weight	Exec. Time
2	0.9	0.1	3.06s
2	0.6	0.4	3.02s
2	0.4	0.6	2.95 s
2	0.1	0.9	2.95s
1	0.1	0.9	2.15s
3	0.1	0.9	2.56

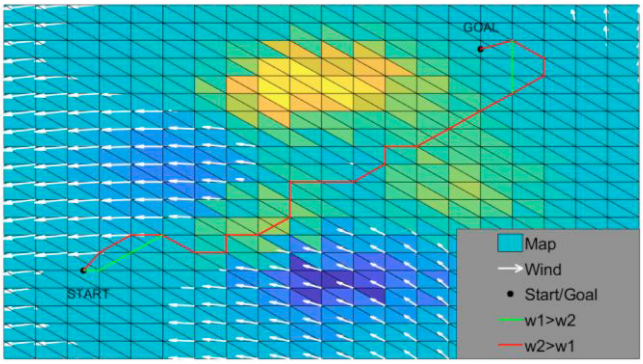

Fig. 7 Simulation of path planning hybrid angle with weights: $w_1 > w_2$ and $w_2 > w_1$ respectively.

Finally, comparative results of the processing time of path generation, using uniform graph as well as PRM graph for the

case of minimum distance and angle criteria, are shown in Tables 3 and 4 respectively. The maximum size of PRM graph random nodes was set to three quarters of the total amount of uniform graph nodes, due to the fact that path formation remains quite similar. The above processing times have been measured for our MATLAB implementation running in single core Intel Core i7 workstation, operating at 3.5GHz with 16 Gb of RAM.

Table 3. Processing Time with minimum distance criteria

	Uniform Graph	PRM Graph
All cases	0.51s	0.15s

Table 4. Processing Time with minimum angle criteria

	Uniform Graph	PRM Graph
All cases	1.07s	0.3s

Tables 3,4 show that in terms of processing time, PRM is much more efficient than this of Uniform graph. This speed up of PRM can offset the processing bottleneck between minimum angle and minimum distance cost function calculation in uniform graph, as processing time of the minimum angle cost function is slightly longer than this of minimum distance. Hybrid angle A* has the same processing cost compared to the minimum distance A*.

5. CONCLUSIONS

This paper presents a smart creation of graph, introducing a veil pattern of states (nodes) above the simulated terrain environment. The proposed variants of A* cost functions, based on either the use of wind orientation only or the use of distance and wind orientation, compared to a minimum distance A*, outperform in terms of path's execution time. However, it was shown that minimum angle A* traps into local minimum with completely opposed wind fields, in contrast to the hybrid angle A*, which ensures a path in shorter time than the distance-based A* under various wind criteria and obstacle presence, using the proposed collision avoidance technique.

The use of PRM offers a speedup in the path planning process, highlighting the potentials for UAV onboard processing. Finally, the hybrid angle A* sets the foundations for a wind optimal global planner, which can be extended with a local planner, which will be able to handle local wind variances in orientation and speed during navigation, which are common in real-time waypoint missions. Finally, the wind aware local planner, running onboard, can be utilized in applications with limited communication resources of a quadcopter with ground station.

ACKNOWLEDGEMENT

This research is a part of work of the project funded by the Interreg Balkan-Mediterranean program: "SFEDA-Forest Monitoring System for Early Fire Detection and Assessment in the Balkan-Med Area (MIS: 5013503)".

REFERENCES

Al-Sabban, W. H., Gonzalez, L. F., Smith, R. N., & Wyeth, G. F. (2012, February). Wind-energy based path planning for electric unmanned aerial vehicles using markov decision processes. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE.

Kavraki, L., Svestka, P., & Overmars, M. H. (1994). *Probabilistic roadmaps for path planning in high-dimensional configuration spaces* (Vol. 1994). Unknown Publisher.

Langelaan, J. (2008). Biologically inspired flight techniques for small and micro unmanned aerial vehicles. In *AIAA guidance, navigation and control conference and exhibit* (p. 6511).

Lawrance, N. R. J., Acevedo, J. J., Chung, J. J., Nguyen, J. L., Wilson, D., & Sukkarieh, S. (2014). Long endurance autonomous flight for unmanned aerial vehicles. *AerospaceLab*, (8), p-1.

Liu, L., & Sukhatme, G. S. (2018). A Solution to Time-Varying Markov Decision Processes. *IEEE Robotics and Automation Letters*, *3*(3), 1631-1638.

MahmoudZadeh, S., Powers, D. M. W., Sammut, K., & Yazdani, A. (2016). Differential evolution for efficient AUV path planning in time variant uncertain underwater environment. *Robotics (cs. RO)*.

Mohammed, F., Idries, A., Mohamed, N., Al-Jaroodi, J., & Jawhar, I. (2014, May). UAVs for smart cities: Opportunities and challenges. In *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on* (pp. 267-273). IEEE.

Moulianitis, V. C., Thanellas, G., Xanthopoulos, N., & Aspragathos, N. A. (2018, June). Evaluation of UAV Based Schemes for Forest Fire Monitoring. In International Conference on Robotics in Alpe-Adria Danube Region (pp. 143-150). Springer, Cham.

Techy, L., & Woolsey, C. A. (2009). Minimum-time path planning for unmanned aerial vehicles in steady uniform winds. *Journal of guidance, control, and dynamics*, *32*(6), 1736-1746.

Windfinder.com (2018), Wind, wave, & weather reports, forecasts and statistics worldwide, Last accessed: 25 Dec 2018.