

Adaptive Path Planning for Unmanned Aircraft Using In-flight Wind Velocity Estimation

Sebastian Benders*, Andreas Wenz†, Tor Arne Johansen†

*German Aerospace Center (DLR)
Institute of Flight Systems
Braunschweig, Germany

†Centre for Autonomous Marine Operations and Systems
Department of Engineering Cybernetics
Norwegian University of Science and Technology, Trondheim, Norway

Abstract—Small fixed-wing unmanned aerial vehicle’s path following performance is highly dependent to the local prevailing wind conditions because of their limited airspeed and flight envelope. In the proposed approach the path following performance is improved, not by optimized control algorithms, but by using wind adaptive path planning. We use a wind velocity estimation, which is capable of estimating steady and turbulent wind using a basic set of small unmanned aircraft on-board sensors. The path planning algorithm considers the aircraft’s kinematics, flight envelope and wind estimate. Simulation results show an improved path following performance and a better exploitation of the flight performance of an unmanned aircraft by the use of the wind adaptive path planning algorithm.

I. INTRODUCTION

Safe and autonomous operation of unmanned aircraft is an important topic of research. The ability of following a planned path with a bounded track error is one aspect of the problem. To achieve this, one can either use path following algorithms and control systems that are designed to limit the path tracking error [15], [22], or alternatively use path planning approaches, which plan paths, that are feasible to track for the unmanned aerial vehicle (UAV), by taking into account kinematic constraints of the aircraft itself and environmental conditions. In this paper we will focus on kinematically feasible path planning incorporating the local prevailing wind provided by wind velocity estimation.

Wind is one of multiple environmental conditions with an impact on the flight behavior of small unmanned aerial vehicles. The effect of wind on the flight path grows with the wind speed to airspeed ratio. As the wind might be varying in both time and space, the correct estimation of the wind speed, direction and turbulence is usually impossible solely from ground measurements. Knowledge about the actual wind conditions enables path planning, with less conservative bounds in order to exploit more of the unmanned vehicles flight envelope. In this work, we combine a path planning algorithm, which is capable of planning UAV paths considering wind, and a method for wind velocity estimation, which is capable to estimate the local prevailing wind by using only sensors that are part of a standard autopilot’s sensor

suite even low-cost UAVs are equipped with. This work is a demonstration of closing the feedback loop between the wind velocity estimation and the path planning module, in order to increase the performance and safety of UAV operations. Figure 1 depicts the proposed approach in the context of a guidance navigation and control (GNC) architecture of an unmanned aircraft.

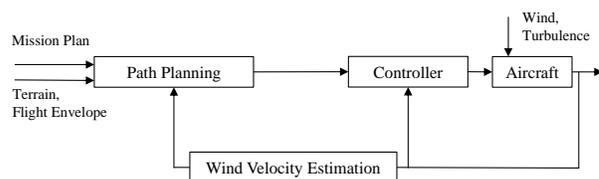


Fig. 1: Closing the feedback-loop between wind velocity estimation and path planning.

The topic of wind adaptive path planning was addressed by several authors, which describe the subsystems for path planning, wind velocity estimation and combinations of these subsystems.

Due to the intention of tracking harmful ocean debris, [18] presents a UAV path planning approach, which considers wind and icing conditions from weather forecasts for planning in oceanic scenarios. A similar approach including a spherical earth model was used for a trans pacific crossing simulation [17]. It was shown, that the fuel-consumption and flight-time could be reduced by considering wind information for the path planning. In [16] waypoint-based paths are planned and a wind dependent look-ahead distance for initiating turning flight in varying wind conditions is used for path following purposes. The simulation shows the better path following by the usage of a wind velocity estimation for the look-ahead distance during turns in wind conditions. The authors of [20] plan paths in a 2D horizontal plane using an Accelerated A* algorithm. A local trochoidal Dubins path planner is used, which considers wind. The path following behavior is flight tested by a blended wing body UAV with

the result of an improved path following behavior if the wind was considered for the path planning.

In order to receive information about the prevailing wind, weather forecast information might be the first choice. But the temporal and spatial resolution and availability may vary depending on the area and provider. Reference [24] describes the change of the wind direction as slowly varying and dependent to large-scale weather systems. The authors of [7] plan UAV paths in time varying wind fields generated by a numeric weather prediction tool. In [11] a path planning approach using forecast wind maps and particle swarm optimization is presented. In this work, we focus on using on-board wind velocity estimation which can provide independent and up-to-date wind information of consistent quality. In addition an on-board wind velocity estimation is able to capture terrain or local weather system induced characteristics of the wind conditions.

Wind velocity estimation for small fixed-wing UAVs has been an active research topic in the past years. A general challenge when estimating wind velocities for this class of aircraft is that they are often not equipped with sensors specifically designed to measure angle of attack and sideslip angle due to restrictions on cost, size and weight. Therefore methods have been developed to estimate these air data parameters and the wind velocities using only sensors which are part of a standard autopilot sensor suite.

One can generally differ between model-based and model-free approaches. In the first category kinetic and kinematic models are used in order to estimate angle of attack, sideslip angle and wind velocities [21], [3] and [6]. A downside of these methods is the need for a large set of aerodynamic parameters to be known, which might be difficult to obtain if no wind tunnel data for the aircraft is available. The second category only uses kinematic relationships, avoiding knowledge about the aerodynamic parameters [12], [14]. However in case of non constant wind fields, frequent maneuvers are needed in order to excite the estimators. To overcome these issues a combination of a model-based and a model-free approach has been studied in [26], [25]. In these references kinematic, aerodynamic and stochastic wind models are combined and the necessary aerodynamic coefficients are estimated online, avoiding the need for prior knowledge about the UAV and limiting the need for maneuvers to excite the estimator to the beginning of the flight.

Contribution of the paper

In this work, we present a method which combines wind velocity based path planning in a 3D environment with inflight wind velocity estimation. The paths are frequently updated with estimates of the local wind velocity. In addition turbulence intensities can be estimated. These estimates are used to plan paths around obstacles with adaptive safety margins to ensure the aircraft's safety, although the aircraft deviates from its planned path.

II. PROBLEM STATEMENT

A. Environment

For the scope of this paper the wind is modeled as constant velocity and direction and constant air density of $\delta = 1.225 \text{ kg/m}^3$. In the scenario with turbulence, turbulent winds are generated using a Dryden wind model [1].

The obstacle environment is assumed to be completely known and represented by a static polygonal model.

B. Vehicle

In this work we use the model of a Skywalker X8 blended wing body unmanned aircraft as depicted in figure 2. This UAV is a popular research platform and has been used for various applications [8].



Fig. 2: Skywalker X8

For the scope of this work, paths are planned by a kinematic model following the works of [5], [19], [23]. The paths are generated from a concatenation of straight and climbing turn flight segments. Equations (1)-(3) describe the trajectories parameterized by time t with ground velocity $\mathbf{v}_g = (u_g, v_g, w_g)$ and speed $V_g = |\mathbf{v}_g|$ in North-East-Down (NED) coordinate system. The motion depends on airspeed $\mathbf{v}_a = (u_a, v_a, w_a)$, $V_a = |\mathbf{v}_a|$ with constant $V_{a,xy} = \sqrt{u_a^2 + v_a^2}$, initial heading ψ_0 , turn rate $\dot{\psi}$, flight path angle γ , wind velocity $\mathbf{v}_w = (u_w, v_w, w_w)$, $V_w = |\mathbf{v}_w|$, $V_{w,xy} = \sqrt{u_w^2 + v_w^2}$, and ψ_w as the azimuth of the wind vector. Transitions between flight path angles are represented as a 4th degree polynomial. As soon as available, the wind \mathbf{v}_w will be provided by the estimation of steady wind velocity \mathbf{v}_s^n .

$$u_g(t) = V_{a,xy} \cos(\dot{\psi}t + \psi_0) + V_{w,xy} \cos(\psi_w) \quad (1)$$

$$v_g(t) = V_{a,xy} \sin(\dot{\psi}t + \psi_0) + V_{w,xy} \sin(\psi_w) \quad (2)$$

$$w_g(t) = \begin{cases} at^4 + bt^3 + ct^2 + dt + e, & (\dot{\gamma} \neq 0) \\ V_g \sin(\gamma), & (\dot{\gamma} = 0) \end{cases} \quad (3)$$

Following the approach described in [5] the path planning further considers a flight envelope which is mainly constrained by the maximum power P_{max} , maximal lift coefficient $C_{L,max}$, load factor n for the planned airspeed V_a and maximal bank angle ϕ . An overview of the model parameters used for the path planning and simulation is given in table I.

TABLE I: Performance properties: Blended wing body UAV Skywalker X8.

wing area	S	0.75 m ²
mass	m	3.36 kg
propeller efficiency	η	0.8
aspect ratio	Λ	5.88
drag coefficient zero	C_{D_0}	0.0102
Oswald efficiency number	e	0.9
max. lift coefficient	$C_{L,max}$	1.0987
max. power	P_{max}	300 W
max. load factor	n_{max}	2
bank angle limitation	ϕ_{max}	45°
airspeed	V_a	15 m/s

C. Adaptive path planning problem

Path planning for UAVs is desired to pre-calculate feasible paths for the UAV. Feasible means, that the planned paths do not violate no-fly zones or other obstacles. In addition, the path planning algorithm has to ensure, that the vehicle is able to track the planned paths. This means meeting the kinematic constraints as a bounded curvature and satisfying the flight envelope, while considering the environmental conditions, especially the prevailing wind velocity. As the wind speed and direction changes with the altitude, a wind measurement at the ground can only serve as an educated guess to be used for path planning for UAVs in higher altitudes. An estimation of wind speed and direction during the flight and a subsequent path planning is intended to generate paths which are feasible to track for fixed-wing UAVs. The estimation of gusts can improve the planning further in order to consider additional safety margins for an expected decreased path following capability.

III. METHOD

A. System Structure

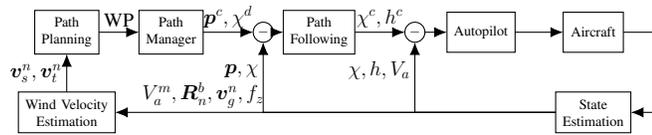


Fig. 3: Simplified diagram of the guidance, navigation and control structure.

Figure 3 shows how the proposed path planning and wind velocity estimation modules fit into the overall GNC system architecture of the UAV. We assume the UAV to be controlled by an autopilot which controls course angle χ , altitude h and airspeed V_a by adjusting the control surfaces and the thrust. The desired values for these three parameters are given by an outer loop where a path following controller minimizes the path error between the actual and controlled position \mathbf{p} , \mathbf{p}^c and the error between actual and desired course angle χ , χ^d . The current waypoint (WP) is picked out of a discretized path, provided by the path planner, using a path manager which switches between waypoints based on a distance criterion. Attitude and heading (represented as the rotation matrix from inertial to body frame \mathbf{R}_n^b) cannot be

directly measured by sensors but have to be estimated from an Inertial Measurement Unit (IMU), a 3-axis magnetometer and Global Navigation Satellite System (GNSS) data by an attitude and heading reference system. The velocity over ground (\mathbf{v}_g) can be measured by the GNSS receiver or estimated using a translational motion observer [3], [13], [9]. These estimates are also inputs to the wind velocity estimator, along with the measurements of the vertical accelerometer f_z and the pitot-static tube V_a^m . The wind velocity estimator provides steady wind velocity \mathbf{v}_s^n and turbulent wind velocity \mathbf{v}_t^n estimates to the path planner.

B. Path planning

The path planner used in this work plans en-route paths for unmanned fixed-wing aircraft, considering their kinematic constraints, flight envelope and static wind conditions. The path planning algorithm is based on the approach of [4]. This approach uses a 3D free-space roadmap (figure 4) to discretize the a priori known and static obstacle environment to allow a run-time efficient multi-query path planning. For the scope of this work, we use a quasi-random sampling and an equidistant grid sampling to discretize the free-space. Subsequently to the sampling, the samples are connected by linear, collision free edges and cylindrical free-space volumes around the nodes are computed.

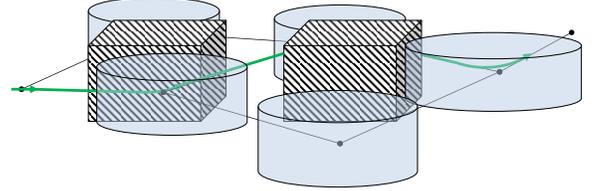


Fig. 4: Schematic illustration of a 3D free-space roadmap with cylindrical free-space volumes.

Once the free-space roadmap is constructed, several planning requests can be processed efficiently on the same free-space roadmap without the need of updating or rebuilding the roadmap, even if the vehicle's flight envelope or wind conditions change. The path planning is performed by the well known A* algorithm. Costs are calculated during the search, based on transitions within the cylindrical free-space volumes. The cost function represents the flight-time for the scope of this work. A planned path consists of a sequence of straight and climbing-turn flight segments. Turn segments are represented by trochoids (see equations (1)-(3)) considering the wind velocity \mathbf{v}_w and a constant airspeed V_a . Every segment is checked on collision by the free-space roadmap and the flight envelope.

The flight paths resulting from the graph search respect the kinematic constraints and is feasible to track for the UAV. However, they may be suboptimal due to the limited sampling density. Thus, we use a short-cutting algorithm to smooth the flight paths. As a last step the resulting path is discretized with a step size less than 5 m and forwarded to the path manager. The described method allows to efficiently re-plan flight paths considering updates of the wind estimation.

C. Wind velocity estimation

Small unmanned aerial vehicles are most often equipped with an autopilot which uses a set of sensors to measure or estimate its position, attitude, velocity over ground and airspeed. These sensors usually include an IMU, a GNSS receiver, a pitot-static tube and sometimes a magnetometer. In most cases, measurements of angle of attack and sideslip angle are not available due to restrictions on size, cost and weight of sensors like multi-hole probes or vanes. In this case wind velocities cannot be directly observed but need to be estimated from the available sensor set using a wind velocity estimator.

In this paper we will use a wind velocity estimator that uses kinematic, aerodynamic and stochastic wind turbulence models in a moving horizon estimator (MHE). This estimator combines the above mentioned input data and outputs estimates of the steady \mathbf{v}_s^n and turbulent wind velocity \mathbf{v}_t^n in inertial frame, as well as estimates of the lift coefficients while filtering measurement noise. These estimates can also be used to calculate the angle of attack and the sideslip angle via the wind triangle [3, ch.2].

In this paper we will only give a short description of the estimator, for a more detailed description and discussion we refer to [26] and [25]. This section distinguishes in the variable superscript between b (body-frame) and n (inertial frame).

Within the estimator a simplified model of the aerodynamic lift force is used:

$$f_z = -KV_a^2(C_{L_0} + C_{L_\alpha}\alpha), \quad (4)$$

where K is a constant factor, α is the angle of attack and C_{L_0} and C_{L_α} are the constant and linear lift force coefficients.

This model is combined with a kinematic model:

$$\mathbf{u}_g^b = \mathbf{d}_1 \mathbf{R}_n^b (\mathbf{v}_s^n + \mathbf{v}_t^n) + V_a \cos(\alpha) \quad (5)$$

$$\mathbf{w}_g^b = \mathbf{d}_3 \mathbf{R}_n^b (\mathbf{v}_s^n + \mathbf{v}_t^n) + V_a \sin(\alpha) \quad (6)$$

with:

$$\mathbf{d}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{d}_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

and

$$\alpha = \arctan\left(\frac{w_a}{u_a}\right) \quad (7)$$

$$\mathbf{v}_a^b = \mathbf{v}_g^b - \mathbf{R}_n^b (\mathbf{v}_s^n + \mathbf{v}_t^n) = [u_a \quad v_a \quad w_a]^T, \quad (8)$$

where \mathbf{v}_a^b is the relative air velocity decomposed in body frame. The wind velocity is modeled as a combination of a steady wind velocity \mathbf{v}_s^n and a turbulent wind velocity \mathbf{v}_t^n , both decomposed in the inertial frame. To model the turbulent wind velocity we use the discrete Dryden wind model [1]:

$$\mathbf{v}_{t,k+1}^n = \mathbf{v}_{t,k}^n - T_s V_{a,k} \begin{pmatrix} \frac{u_t^n}{L_u} \\ \frac{v_t^n}{L_v} \\ \frac{w_t^n}{L_w} \end{pmatrix} \Big|_k + \begin{pmatrix} \sigma_u \sqrt{2T_s \frac{V_a}{L_u}} \eta_{u_t} \\ \sigma_v \sqrt{2T_s \frac{V_a}{L_v}} \eta_{v_t} \\ \sigma_w \sqrt{2T_s \frac{V_a}{L_w}} \eta_{w_t} \end{pmatrix} \Big|_k, \quad (9)$$

where k is the current time step, T_s is the sampling period, $[L_u \ L_v \ L_w]$ are spatial wave lengths, $[\sigma_u \ \sigma_v \ \sigma_w]$ are the gust amplitudes and $[\eta_{u_t} \ \eta_{v_t} \ \eta_{w_t}]$ are noise variables. The spatial wave lengths and gust amplitudes are dependant on the wind velocity 6 m above ground and the altitude of the aircraft [1].

These three models are combined in a moving horizon estimator (MHE) to estimate the steady and turbulent wind velocities and the lift coefficients. Additionally a pitot tube scaling factor can be estimated. The required inputs are the rotation matrix between inertial and body frame \mathbf{R}_n^b , given by an attitude and heading reference system and the airspeed V_a^m , measured by a pitot-static tube. The model outputs are compared to the specific force in z-direction f_z , measured by an accelerometer and the aircraft's velocity over ground decomposed in body frame \mathbf{v}_g^b , measured by a GNSS receiver.

In order to excite the estimator, attitude changes are necessary after take off. Once the estimator has converged, natural excitation created by turbulence or maneuvering as part of the mission plan are usually sufficient to avoid drift of the estimates.

IV. CONTROL STRUCTURE

In the following sections we will discuss the path manager, the path following controller and the autopilot.

1) *Path Manager*: The task of the path manager is to extract the right waypoints out of the list of waypoints provided by the path planner and to pass them on to the path following controller. The waypoints of interest are the previous waypoint \mathbf{p}_A and the next waypoint \mathbf{p}_B which are needed to calculate the path error \mathbf{e} as well as a predicted waypoint along the path \mathbf{p}_{pred} . This predicted waypoint is always at a look-ahead time of ΔT and determines the desired course angle χ^d . The effect of this is a smoothing of the bank angle steps assumed in the path planning which are infeasible to follow exactly due to delays in the system. The value of ΔT can be regarded as a tuning factor, where a small value leads to a more aggressive maneuver and a large value leads to a more smooth turning. On the other hand, a too small value leads to overshooting and a too large value leads to cutting of curves. The look-ahead time was manually tuned in the simulation. For the considered vehicle, a value of $\Delta T = 0.5$ s resulted in a well balanced turning behavior.

The algorithm of the path manager is described in [3, ch.10]. The path manager switches to the next waypoint if the UAV crosses the halfplane separating the waypoint straight-line segments AB and BC , where \mathbf{p}_C is the next but one waypoint:

$$0 \geq (\mathbf{p} - \mathbf{p}_B)^T \mathbf{n} \quad (10)$$

with:

$$\mathbf{n} = \frac{\mathbf{q}_{AB} - \mathbf{q}_{BC}}{\|\mathbf{q}_{AB} - \mathbf{q}_{BC}\|} \quad (11)$$

where $\mathbf{q}_{AB} = [q_{AB,n}, q_{AB,e}, q_{AB,d}]$ is the NED path vector, originating at \mathbf{p}_A and ending at \mathbf{p}_B and \mathbf{q}_{BC} is the next path vector originating at \mathbf{p}_B and ending at \mathbf{p}_C .

2) *Path following*: In order to follow the waypoints a path following controller is used. For the path following we assume the horizontal and vertical motion of the UAV to be decoupled.

For the horizontal path following guidance controller we use a line of sight (LOS) controller as described in [3, ch.9]

The LOS course controller is given by the following control law:

$$\chi^c = \chi^d + \tan^{-1} \left(-\frac{1}{\Delta L} e_{py} \right) \quad (12)$$

Where χ^c is the commanded course angle, which is passed on to the course controller, χ^d is the desired path angle at the predicted waypoint \mathbf{p}_{pred} and ΔL is the look-ahead distance, set to $\Delta L = 12$ m. The cross track error e_{py} is defined as the projection of the path error $\mathbf{e} = \mathbf{p}_A - \mathbf{p}$ onto the horizontal plane:

$$e_{py} = [-\sin \chi^p \quad \cos \chi^p \quad 0]^T \mathbf{e} \quad (13)$$

with $\chi^p = \tan^{-1} \left(\frac{q_{AB,e}}{q_{AB,n}} \right)$.

For the vertical path following we use the control law established in [3, ch. 10]:

$$h^c = -p_{A,d} + \sqrt{s_n^2 + s_e^2} \left(\frac{q_{AB,d}}{\sqrt{q_{AB,n}^2 + q_{AB,e}^2}} \right). \quad (14)$$

Where $\mathbf{p}_{A,d}$ is the desired position in down direction on the path. Vector $\mathbf{s} = [s_n, s_e, s_d]$ is the projection of the path error \mathbf{e} onto the vertical plane containing the path vector \mathbf{q}_{AB} (for an illustration see [3, p.176]).

3) *Autopilot*: The autopilot design follows [3, ch. 6] closely, where successive loop closure with proportion-integral-derivative (PID) controllers are used to control lateral and longitudinal movement of the UAV. In order to handle the different stages of the flight a state machine is used. For an in depth discussion on the autopilot design and tuning we refer to [3, ch. 6] and will only discuss changes we made in order to increase the autopilot's performance when wind is present.

To follow the path by controlling the course angle, a course controller is required to minimize the error between commanded course χ^c and actual course χ . This is implemented by commanding a course angle change with the following proportional-integral (PI) controller:

$$\dot{\chi}^c = k_{p\chi}(\chi_c - \chi) + \int k_{i\chi}(\chi_c - \chi) \quad (15)$$

Here $k_{p\chi}$ and $k_{i\chi}$ are the controller gains which can be selected as described in [3, ch. 6.3.2]. Assuming a coordinated turn, we can convert the course rate into a roll angle using [3, p. 166]:

$$\phi^c = \tan^{-1} \left(\frac{V_g \dot{\chi}^c}{g \cos \zeta} \right) \quad (16)$$

Where V_g is the speed relative to the inertial frame, g is the gravitational acceleration and $\zeta = \chi - \psi$ is the crab angle. Calculating the crab angle requires knowledge of the heading

angle which can be estimated using e.g. a magnetometer. If the crab angle is small we can assume $\cos \zeta \approx 1$ and get:

$$\phi^c = \tan^{-1} \left(\frac{V_g \dot{\chi}^c}{g} \right) \quad (17)$$

This approximation avoids the need for a heading reference. In flights with low wind condition one can regard the approximation error as a disturbance which can be handled by the PI controller. However in situations with high winds the approximation error can grow significantly making the use of the crab angle compensation desirable. Note that when flying in windy conditions, equation (16) becomes time varying since V_g and ζ will vary significantly.

V. SIMULATION SETUP

The simulation of the Skywalker X8 is implemented in a Matlab / Simulink environment. The kinematic and aerodynamic models used in the simulator are described in [3, ch.2-4]. Coefficients for the Skywalker X8 have been retrieved from measurements on the UAV and computational fluid dynamics (CFD) analysis [10].

In this paper we will not consider the problem of attitude and velocity estimation. Therefore we assume the availability of an attitude \mathbf{R}_n^b and velocity over ground \mathbf{v}_g estimate, which is bias free. Furthermore, we assume that measurement noise is handled in the attitude estimator and that the attitude estimate has negligible noise levels. For the velocity over ground estimate we assume it to be affected by Gaussian white measurement noise with a variance of 10^{-4} m/s. The simulated airspeed and accelerometer measurements are also affected by Gaussian white noise with variances of 10^{-3} m/s and 10^{-3} m/s² respectively.

VI. RESULTS

In this section three scenarios are presented to show effects of the proposed combination of path planning and wind velocity estimation in a simulation environment with the Skywalker X8 UAV. Ahead of the simulated scenarios, the effect of the small angle approximation of the crab angle ζ in wind conditions is outlined. The simulation scenarios are a level flight scenario, climbing flight scenario and obstacle scenario with turbulent wind conditions. The evaluation closes with a comparative section of the presented scenarios.

A. Effect of crab angle compensation

In order to explore the performance difference between the two course controllers described by equations (16) and (17), we compare the resulting cross track errors for both approaches and for scenarios shown in figure 5 and 6. Figure 5 and 6 show the cross-track errors for the scenarios in steady wind and turbulent wind conditions respectively.

The controller with crab angle compensation shows less overshoot in turns than the one without, especially in turbulent conditions. This is expected since this controller takes the crab angle explicitly into account. The root mean square (RMS) errors are 3.8 m for the controller with crab angle compensation and 4.25 m for the one without for the flight shown in figure 5, and 5.3 m and 10.8 m respectively for

the flight shown in figure 6. Note that in real flight, errors in the heading reference could reduce these benefits. In the following, the controller with crab angle compensation will be used.

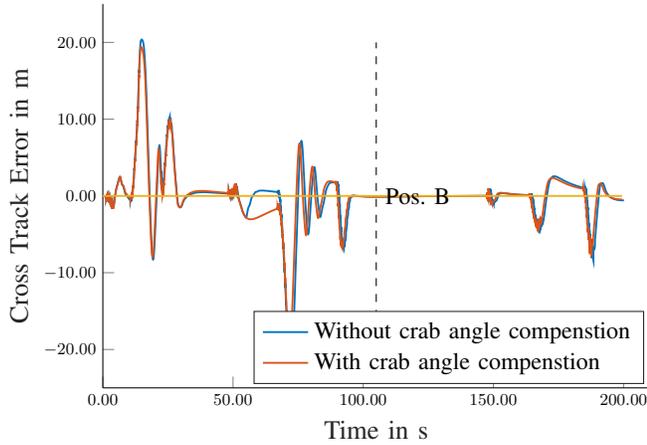


Fig. 5: Effect of crab angle compensation on cross track error (flight path shown in figure 7).

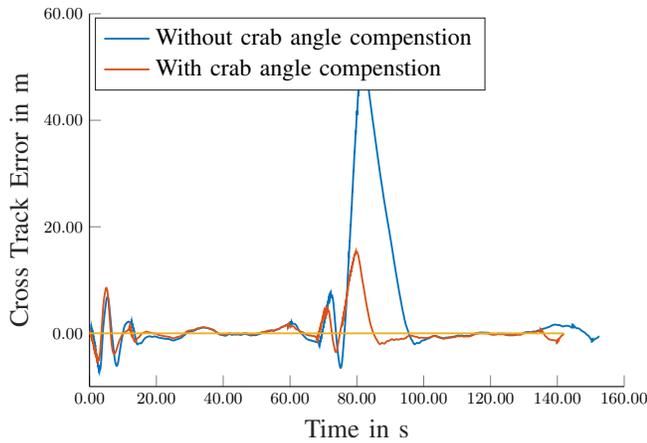


Fig. 6: Effect of crab angle compensation on cross track error in turbulent conditions (flight path shown in figure 16).

B. Level flight in wind

In figure 7 a top-view on the simulated trajectory for a level flight scenario is presented. The scenario is simulated with airspeed $V_a = 15$ m/s, a prevailing wind of $V_w = 5$ m/s and $\psi_w = 180^\circ$ (see simulated wind in figure 8). The initial path (part A) is planned without knowledge about the prevailing wind conditions. The path planning assumes windless conditions ($V_w = 0$ m/s) and does not consider conservative $\dot{\psi}$, respectively turn radii. Starting at position A, the aircraft climbs to its flight level and follows the initially planned triangular path. Especially in tailwind conditions the UAV deviates from the planned path, due to the bank angle limitation of $\phi_{max} = 45^\circ$. Figure 8 shows the estimated and simulated wind velocities. The first flight segment provides the wind velocity estimator with sufficient excitation, so

that the the wind velocity estimation converges to an wind velocity estimate of $V_w^m = 5$ m/s from north direction with very little error. Convergence is achieved at around $t = 100$ s. When passing position B for the second time at flight time $t_f = 105$ s the second triangular path segment (part B) is re-planned considering the estimated wind. The planning time of the re-planning was about $t_p = 1.1$ s. Figure 5 shows the cross track error during the flight. The dashed line indicates the point of re-planning. After the point of re-planning the cross track error is clearly smaller, for both controllers. In detail, the RMS cross track error decreases from 5 m to 1.4 m for the controller with crab angle compensation. This means, that the wind adaptive re-planning results in an improved path following performance. This can also be seen in table IV.

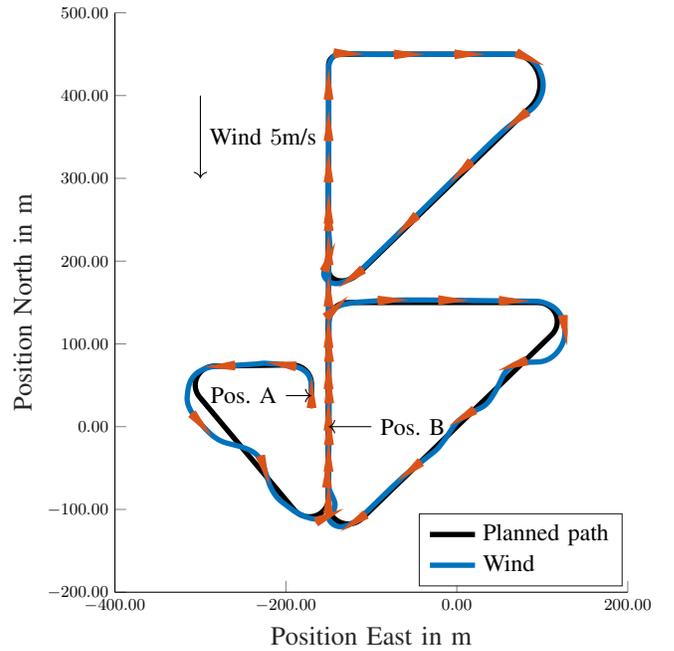


Fig. 7: Level flight starting at Position A and with wind adaptive in-flight re-planning at Position B. The lower triangles are planned and flown without, and the upper with knowledge about the prevailing wind velocity.

C. Climbing flight in wind

Beside the effects of the wind on the level flight path, the vertical flight performance also depends on the prevailing wind conditions. In figures 9 and 10 a path segment of climbing flight which is planned in headwind ($V_w = 6$ m/s, $\psi_w = 180^\circ$) conditions is simulated in varying wind conditions. The planning and simulation airspeed is $V_a = 15$ m/s. The aircraft is able track the planned path if the conditions in the planning and simulation match (headwind case in the plot). If the assumed wind in the simulation is set to $V_w = 0$ m/s or the wind-direction to tailwind ($V_w = 6$ m/s, $\psi_w = 0^\circ$), the aircraft is not capable to achieve the planned flight path angle.

The scenario shown in figures 9 and 10 demonstrates how an adaptive path planning considering the estimated

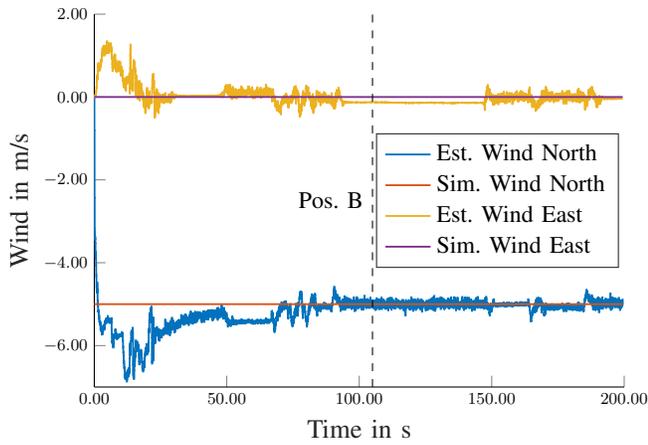


Fig. 8: Estimated and simulated wind velocities.

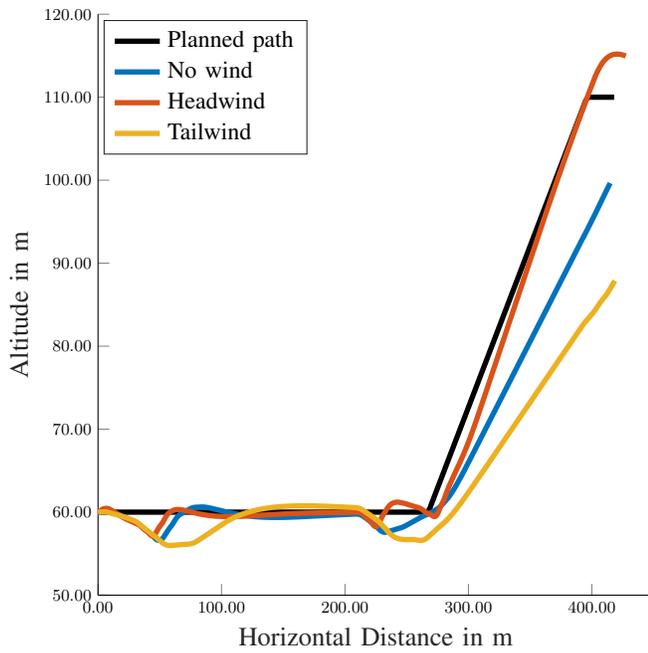


Fig. 9: Climbing flight path segment in varying wind conditions. The planned path assumes headwind conditions.

local prevailing wind is able to exploit the flight envelope capabilities of the aircraft in terms of maximal flight path angle, using information about the actual wind conditions.

In figure 9 the maximum flight path angle of the aircraft in tailwind conditions is not sufficient to reach the new altitude within the given flight distance. If the wind velocity estimation measures tailwind conditions, a re-planning generates feasible paths for this scenario. In figure 12 and 11 the same path segment was planned for tailwind conditions ($V_w = 6 \text{ m/s}, \psi_w = 0^\circ$). The planning algorithm inserts a full turn to let the UAV reach the target altitude with a flight path angle, which is feasible to track. Figure 11 shows the altitude versus the position along the path for this scenario. Note how the UAV is able to track the vertical profile in headwind and tailwind conditions. However in north-wind

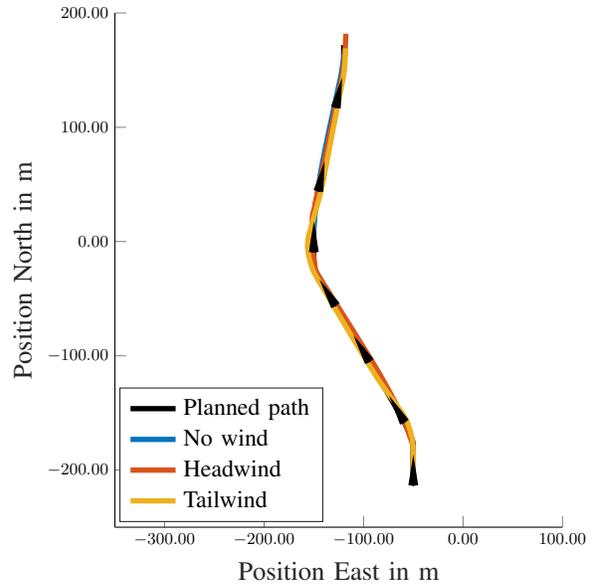


Fig. 10: Horizontal path of climbing flight scenario in figure 9.

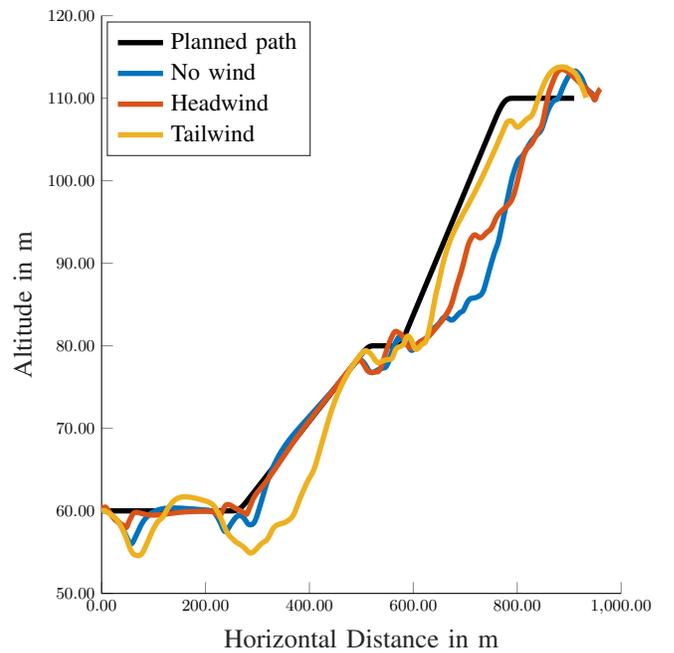


Fig. 11: Vertical path in varying wind conditions. The planned path assumes tailwind conditions.

(depicted as headwind-case, $V_w = 6 \text{ m/s}, \psi_w = 180^\circ$) and windless ($V_w = 0 \text{ m/s}$) conditions a larger cross track error occurs (see table IV), because the UAV flies temporarily in tailwind conditions during the full turn.

D. Turbulence

The scenario presented in figures 13 and 16 demonstrate the effect of turbulence on the path following behavior and path planning. Turbulent winds are generated using the Dryden wind model with a wind velocity at ground of 8 m/s .

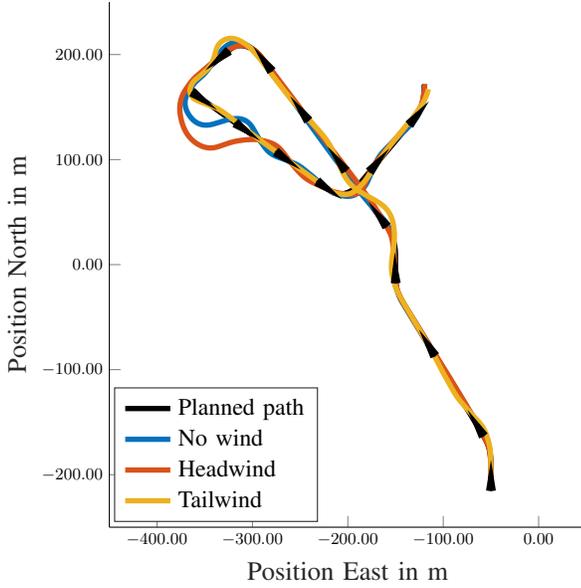


Fig. 12: Horizontal path of climbing flight scenario in figure 11.

In case of low to medium turbulent wind conditions the controller limits the path deviations to small values in the simulation. Therefore we focus on highly turbulent wind gusts, where the wind velocity temporarily reaches or exceeds the aircraft's airspeed ($V_a = 15$ m/s).

If the wind velocity, which is modeled as a combination of steady and turbulent wind velocity, exceeds the aircraft's airspeed for a period of time, the aircraft loses the ability to control its lateral position which results in local high deviations from the path. The time it takes for the path following controller to steer the aircraft back to the path also depends on the difference between airspeed and wind velocity. Reference [2] shows that straight line path following is only possible if for the lateral wind velocity w_y holds:

$$|w_y| < |V_a \cos(\gamma_{max})| \quad (18)$$

(here: $w_y < 14$ m/s), while during a turn the following condition has to hold:

$$|w_y| < |V_a \cos(\gamma_{max}) \cos(\psi^c - \chi^p)|. \quad (19)$$

We can reformulate equation (19) using the lateral path following control law from equation (12).

$$|w_y| < \left| \frac{V_a \cos(\gamma_{max})}{\sqrt{\Delta L^2 + e_{py}^2}} (\cos \zeta \Delta L - \sin \zeta e_{py}) \right| \quad (20)$$

Note, that it is assumed here that the commanded course is followed perfectly by the course controller.

Figure 14 shows the maximal tolerable lateral wind velocity and the actual wind velocity which exceeds the first one in several time spans, which can be interpreted as a loss of controllability. In these areas we also see the largest cross track errors in figure 15.

The flight path in figure 13 is not safe, as the aircraft leaves its safe flight corridor defined by the safety margin of 10 m. Because turbulent wind is not represented in the path planning model (equations (1)-(3)), we use additional safety margins S depending on the maximal cross track error to ensure safe flight in case of turbulent wind:

$$S \geq \max(e_{py,i}) = 20 \text{ m}. \quad (21)$$

The effect of adapted safety margins results in a detour for this scenario as shown in figure 16, but the aircraft's trajectory does not violate the safety margins anymore.

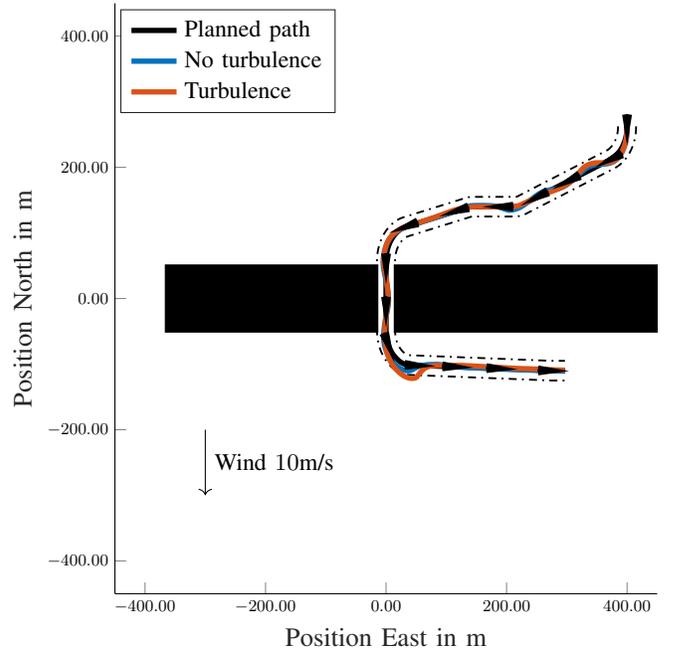


Fig. 13: Obstacle scenario, planned with a safety margin of $S = 10$ m.

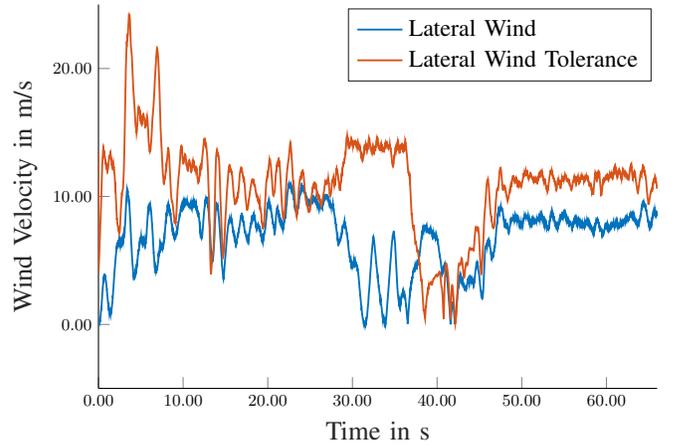


Fig. 14: Maximal tolerable lateral wind velocity and actual wind velocity for obstacle scenario in figure 13.

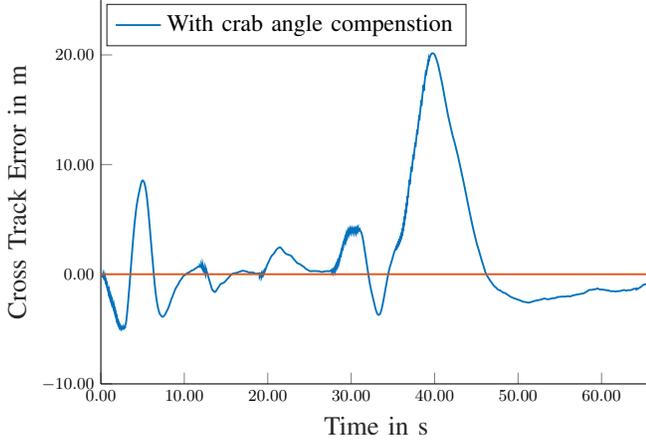


Fig. 15: Cross track error for obstacle scenario in figure 13.

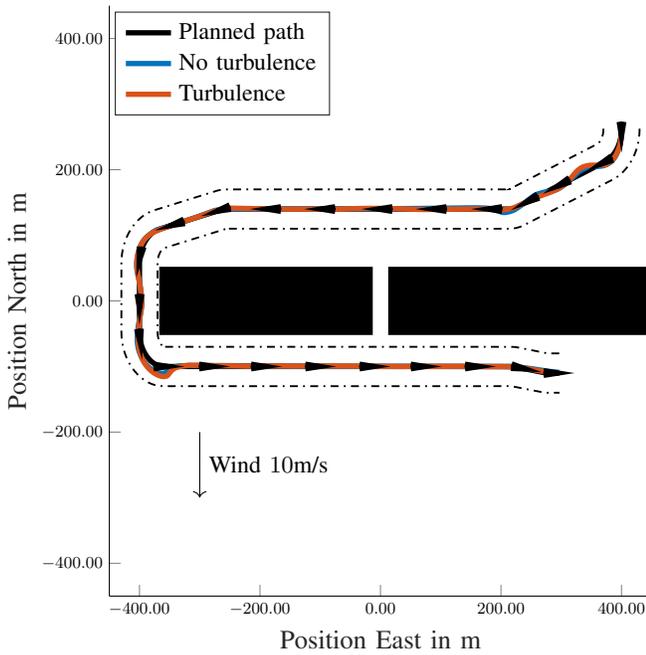


Fig. 16: Obstacle scenario, planned with a safety margin of $S = 30$ m.

E. Comparison of the scenarios

Table II lists the planning run-times t_p on a ground control station¹ and shows, that the planning run-times compared to the planned path cost (estimated flight time t_{est}) are

¹WIN64, i7-6700 3,4 GHz, 16 GB RAM

TABLE II: Path planning run-times.

Scenario	t_p	$\frac{t_p}{t_{est}}$
level flight part A (figure 7)	0.6 s	$5.9 \cdot 10^{-3}$
level flight part B (figure 7)	1.1 s	$11.7 \cdot 10^{-3}$
climbing flight (figure 9)	1.5 s	$33.3 \cdot 10^{-3}$
climbing flight (figure 11)	1.6 s	$40.0 \cdot 10^{-3}$
obstacle scenario (figure 13)	0.3 s	$4.8 \cdot 10^{-3}$
obstacle scenario (figure 16)	2.6 s	$19.5 \cdot 10^{-3}$

sufficiently low for an online path planning. The memory consumption of the 3D free-space roadmap used for path planning are in a range of 3-10 MB for the presented scenarios. These results imply, that the approach is feasible for on-board re-planning applications.

Table III compares estimated t_{est} and simulated flight times t_{sim} . The estimated flight times are provided by the path planning algorithm. In the level flight scenario, the match of estimated to simulated flight time is improved in part B where the planning is adapted to the estimated wind conditions.

In the climbing flight scenario, the estimated and simulated flight times match better if the correction of wind conditions is used for the planning. This also applies for the level flight scenario.

The obstacle scenario compares turbulent and non turbulent wind conditions. As turbulent wind is not incorporated into the kinematic model used for path planning, the estimated and simulated flight times deviate in case of turbulence.

TABLE III: Comparison of estimated and simulated flight times.

Scenario	t_{est}	t_{sim}	diff.
level flight part A (figure 7)	94 s	105 s	+11%
level flight part B (figure 7) *	93 s	96 s	+3%
climbing flight no wind (figure 9)	45 s	28 s	-37%
climbing flight headwind (figure 9) *	45 s	46 s	+2%
climbing flight tailwind (figure 9)	45 s	21 s	-53%
climbing flight no wind (figure 11)	62 s	64 s	+2%
climbing flight headwind (figure 11)	62 s	85 s	+37%
climbing flight tailwind (figure 11) *	62 s	59 s	-4%
obstacle scenario no turbulence (figure 13) *	66 s	64 s	-2%
obstacle scenario turbulence (figure 13)	66 s	70 s	+6%
obstacle scenario no turbulence (figure 16) *	146 s	163 s	+5%
obstacle scenario turbulence (figure 16)	146 s	163 s	+10%

* wind in path planning and simulation match

As can be seen in figures 5 and 6 the cross track error varies depending on the wind conditions taken into account in the path planning and simulation. Table IV shows the cross track error e_{py} of the corresponding scenarios. It can be seen, that in scenarios where the wind considered in the path planning matches the wind in the simulation, result in lowest cross track errors.

TABLE IV: Cross track error e_{py} .

Scenario	RMS e_{py}	Max. e_{py}
level flight part A (figure 7)	5 m	19.6 m
level flight part B (figure 7) *	1.43 m	7.9 m
climbing flight no wind (figure 9)	1.2 m	2 m
climbing flight headwind (figure 9) *	1.3 m	2.5 m
climbing flight tailwind (figure 9)	2.2 m	6.3 m
climbing flight no wind (figure 11)	4.7 m	21.8 m
climbing flight headwind (figure 11)	8.3 m	38.5 m
climbing flight tailwind (figure 11) *	3.2 m	11.5 m
obstacle scenario no turbulence (figure 13) *	3 m	9.3 m
obstacle scenario turbulence (figure 13)	5.38 m	20 m
obstacle scenario no turbulence (figure 16) *	2.4 m	11 m
obstacle scenario turbulence (figure 16)	2.8 m	15.3 m

* wind in path planning and simulation match

VII. CONCLUSION AND FUTURE WORK

This paper presents an approach for adaptive path planning for fixed-wing UAVs using in-flight wind velocity estimation. The path planning algorithm uses terrain information and vehicle specific kinematic constraints together with wind velocity and direction information to plan feasible paths. A wind velocity estimator provides the path planner with the necessary estimates which are estimated from a standard autopilot's sensor suite without any prior knowledge about the UAV's aerodynamic characteristics.

The results of three simulation scenarios show that the wind adaptive path planning approach provides shorter flight times, better horizontal and vertical path following performance, as well as the possibility to take turbulent wind into account by adapting safety margins around obstacles. In further research, a metric could be used to define sufficiently large safety distances as a function of estimated turbulence intensity, in order to prevent path tracking deviations greater than the defined safety margins.

The method can further be extended by considering time varying wind and weather forecasts which could be locally updated by the in-flight wind velocity estimation. Flight tests, and especially beyond visual line of sight (BVLOS) flight tests are a consistent next step to show the benefits of wind adaptive path planning.

VIII. ACKNOWLEDGMENTS

This project has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 642153. The research was also funded by the Research Council of Norway through the Centres of Excellence funding scheme, grant number 223254 NTNU AMOS. We would like to thank Kristoffer Gryte for the development of the UAV simulator.

REFERENCES

- [1] MIL-STD-1797A: Flying Qualities of Piloted Aircraft. Technical report, 2004.
- [2] Randal W. Beard, Jeff Ferrin, and Jeffrey Humpherys. Fixed wing UAV path following in wind with input constraints. *IEEE Transactions on Control Systems Technology*, 22(6):2103–2117, 2014.
- [3] Randal W. Beard and Timothy W. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012.
- [4] Sebastian Benders. Reconfigurable path planning for fixed-wing unmanned aircraft using free-space roadmaps. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, Dallas, 2018. IEEE.
- [5] Sebastian Benders and Simon Schopferer. A line-graph path planner for performance constrained fixed-wing UAVs in wind fields. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 79–86, Miami, 2017. IEEE.
- [6] Kasper T. Borup, Thor I. Fossen, and Tor A. Johansen. A Nonlinear Model-Based Wind Velocity Observer for Unmanned Aerial Vehicles. In *10th IFAC Symposium on Nonlinear Control Systems*, volume 49, pages 276–283, Monterrey, CA, USA, 2016.
- [7] Anjan Chakrabarty and Jack Langelaan. UAV flight path planning in time varying complex wind-fields. In *American Control Conference (ACC)*, 2013, pages 2568–2574. IEEE, 2013.
- [8] Joao Borges de Sousa, Joao Pereira, Jose Pinto, Paulo Claro Lourenfo, Joao Madaleno Galocha, Jorge Fontes, Monica Silva, Kanna Rajan, Tor Arne Johansen, Joao Alves, Andrea Munafò, Konstantinos Pelekanakis, Roberto Petroccia, Marina Carreiro Silva, and Michael Ince. Rapid Environmental Picture Atlantic exercise 2015: A field report. In *OCEANS 2016*, pages 1–6, Monterey, 9 2016. IEEE.
- [9] Håvard Fjær Grip, Thor I. Fossen, Tor A. Johansen, and Ali Saberi. Nonlinear observer for GNSS-aided inertial navigation with quaternion-based attitude estimation. In *American Control Conference*, pages 272–279, 6 2013.
- [10] Kristofer Gryte, Richard Hann, Mushfiqul Alam, and Jan Roháč. Aerodynamic modeling of the Skywalker X8 Fixed-Wing Unmanned Aerial Vehicle. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, Dallas, 2018.
- [11] Anthony Reinier Hovenburg, Fabio Augusto, De Alcantara Andrade, Christopher Dahlin Rodin, Tor Arne Johansen, and Rune Storvold. Inclusion of Horizontal Windmaps in Path Planning Optimization of small UAS. In *International Conference on Unmanned Aircraft Systems*, Dallas, 2018.
- [12] Tor A Johansen, Andrea Cristofaro, Kim Sørensen, Jakob M Hansen, and Thor I Fossen. On estimation of wind velocity, angle-of-attack and sideslip angle of small UAVs using standard sensors. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, Denver, 2015.
- [13] Derek B Kingston and Randal W. Beard. Real-time Attitude and Position Estimation for Small UAVs using Low-cost Sensors. *Proc. of the AIAA Unmanned Unlimited Technical Conference, Workshop and Exhibit*, pages 2004–6488, 2004.
- [14] Jack W Langelaan, Nicholas Alley, and James Neidhoefer. Wind Field Estimation for Small Unmanned Aerial Vehicles. *Journal of Guidance Control and Dynamics*, 34:1016–1030, 2011.
- [15] Derek R Nelson, D Blake Barber, Timothy W McLain, and Randal W Beard. Vector field path following for miniature air vehicles. *IEEE Transactions on Robotics*, 23(3):519–529, 2007.
- [16] John Osborne and Rolf Rysdyk. Waypoint guidance for small UAVs in wind. In *Infotech@ Aerospace*, page 6951. 2005.
- [17] Juan Carlos Rubio and Sean Kragelund. The trans-pacific crossing: long range adaptive path planning for UAVs through variable wind fields. In *Digital Avionics Systems Conference, 2003. DASC'03. The 22nd*, volume 2, pages 8–B. IEEE, 2003.
- [18] Juan Carlos Rubio, Juris Vagners, and Rolf Rysdyk. Adaptive path planning for autonomous UAV oceanic search missions. In *AIAA 1st Intelligent Systems Technical Conference*, pages 20–22, 2004.
- [19] Simon Schopferer and Thorsten Pfeifer. Performance-aware flight path planning for unmanned aircraft in uniform wind fields. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1138–1147. IEEE, 2015.
- [20] Martin Selecký, Petr Váňa, Milan Rollo, and Tomáš Meiser. Wind corrections in flight path planning. *International Journal of Advanced Robotic Systems*, 10(5):248, 2013.
- [21] Mohammad Shaqura and Christian Claudel. A hybrid system approach to airspeed, angle of attack and sideslip estimation in Unmanned Aerial Vehicles. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 723–732, 2015.
- [22] PB Sujit, Srikanth Saripalli, and Joao Borges Sousa. Unmanned aerial vehicle path following: A survey and analysis of algorithms for fixed-wing unmanned aerial vehicles. *IEEE Control Systems*, 34(1):42–59, 2014.
- [23] Laszlo Techy and Craig A Woolsey. Minimum-time path planning for unmanned aerial vehicles in steady uniform winds. *Journal of Guidance, Control, and Dynamics*, 32(6):1736–1746, 2009.
- [24] Eric van Doorn, Brindesh Dhruva, Katepalli R Sreenivasan, and Victor Cassella. Statistics of wind direction and its increments. *Physics of Fluids*, 12(6):1529–1534, 2000.
- [25] Andreas Wenz and Tor Arne Johansen. Estimation of Wind Velocities and Aerodynamic Coefficients for UAVs using standard Autopilot Sensors and a Moving Horizon Estimator. *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1267–1276, 2017.
- [26] Andreas Wenz, Tor Arne Johansen, and Andrea Cristofaro. Combining model-free and model-based angle of attack estimation for small fixed-wing UAVs using a standard sensor suite. In *International Conference on Unmanned Aircraft Systems (ICUAS 2016)*, pages 624–632, Arlington, VA, 2016.