Moving Horizon Estimation of Air Data Parameters for UAVs

Andreas Wenz and Tor Arne Johansen

***Abstract*—Local wind velocities, angle of attack and lift coefficients of a fixed-wing unmanned aerial vehicle (UAV) are estimated by fusing kinematic, aerodynamic and stochastic wind models with data from an inertial measurement unit, a global navigation satellite system receiver and a pitot-static tube in a Moving Horizon Estimator. Experimental validation with two different UAVs and two sensor sets of different quality, show promising results for both wind velocity and angle of attack estimation.**

***Index Terms*—state estimation, unmanned aircraft, moving horizon estimation, avionics, sensor fusion**

I. Introduction

The air data parameters Angle of attack (α), side slip angle (β) and airspeed (V_a) are some of the most useful variables to assess flight performance and safety of fixed-wing aircraft. They are especially important when encountering abnormal conditions, such as stall and strong and turbulent winds. In manned aviation and on larger high-cost unmanned aircraft, dedicated sensors are available to measuring these variables. Devices for acquiring this data include vanes or pressure sensors like multi-hole pitot-static tubes. Additionally, wind tunnel and CFD data are often available for large aircraft allowing identification of a full parametric dynamical model of the aircraft. This model can then be used together with inertial measurements to estimate the air data parameters and providing model based redundancy. However, for small fixed-wing UAVs the mentioned sensor systems are often difficult to use since strict limitations on size, weight and cost prove to be prohibiting. Also, wind tunnel data is often unavailable or too expensive to obtain for specific air frames, prohibiting the use of extensive parametric aerodynamic models for air data estimation. Some of these aerodynamic parameters might even change with different payload configurations (e.g. externally mounted antennas or cameras). What makes the estimation problem more challenging is that unlike in manned aviation airspeeds are generally in the same order of magnitude as the local wind velocities. In this case wind velocities have to be accounted for in the modeling process. Since UAVs often operate at low altitudes close to ground, the wind gust intensity changes significantly depending on altitude and terrain, creating a need for wind gust models that capture these effects.

Due to these circumstances it is desirable to develop estimation concepts that do not rely on the availability of a full aerodynamic model and only use sensors that are part of a standard sensor suite consisting of a GNSS (Global Navigation Satellite System) receiver, an IMU (Inertial Measurement Unit) and a standard single-directional pitot-static tube. This sensor data can then be fused, using kinematic, aerodynamic and wind models within an estimator. In order to simultaneously estimate the aerodynamic coefficients and the wind velocities impacting the aircraft, which allows the computation of the air data parameters.

The availability of wind velocity and aerodynamic coefficient estimates during a flight is desirable as this provides a basis for a variety of different use cases. One would be the possibility to monitor the aerodynamic coefficients for changes as they can appear in icing scenarios [17] allowing the precise use of de-icing methods [38]. Another area of application could be path planning were the availability of wind velocity estimates allow on the one side more efficient flight trajectories and on the other side less conservative flight trajectories when flying close to mountains or buildings.

An angle of attack estimate makes it possible to limit the angle of attack during maneuvers in order to prevent stall, improving flight safety. Furthermore, the knowledge of the air data parameters allows the exploitation of wind features resulting in a more energy efficient flight and increased range of the UAV [22] [36].

Recently several methods have been proposed to estimate the air data parameters and aerodynamic coefficients. One popular methods is the Extended Kalman Filter (EKF), which has been used in [11] , [21], [24], and the Unscented Kalman Filter (UKF) which has been applied to the problem in [33], [12]. Tian et.al [40] compare the use of an EKF, Output Error Minimization and a Complimentary Filter to improve measurements of the air data parameters obtained from a multi-hole probe. [8] uses a detailed aircraft model and a nonlinear observer for wind estimation. In [23] kinematic vehicle models are used together with IMU, GNSS and multi-directional airspeed sensors to estimate the wind field. [35] propose a method using optical flow measurements. A hybrid system approach using Bayesian estimation is presented in [37] achieving promising results. An air data estimation method using simplified aerodynamic and kinematic models have been presented in [37], [5] and [8]. However in the described method a detailed model of the aerodynamics is used and the parameters for this might not always be available.

This paper builds up on several previously published papers. In [18] wind velocities were estimated using only kinematic relationships, resulting in a linear time varying systems in which states can be estimates using a Kalman Filter approach. The underlying assumption was that the wind velocity is only slowly time varying limiting the performance of the estimator

The authors are with the Center for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics at the Norwegian University of Science and Technology, Trondheim, Norway.

in turbulent wind conditions, as they are often experienced close to ground due to boundary layer effects.

In [44] the above mentioned sensor suite was used together with aerodynamic and kinematic relationships and a stochastic wind model to estimate time varying wind speeds. The necessary coefficients of the aerodynamic model are estimated together with steady and turbulent wind velocities. An EKF was used to fuse the sensor information and models. A general problem regarding the system properties is the lack of observability in periods where no attitude changes occur. In these cases the parameter estimates of the EKF tend to drift over time and the estimated error covariances are inconsistent, causing the need for compensation by periodic attitude changes. Additionally the EKF estimates were sensitive to input noise, due to the nonlinearity of the used estimation models.

To overcome these issues, we proposed to use a Moving Horizon Estimator (MHE) in [43]. A MHE uses the data gathered form the sensor measurements in a time window of length L until the current time together with a system model to estimate the state trajectory in the time window. This is done by minimizing the error between the system output and the measurement vector forming a Nonlinear Program (NLP). MHE has been an active research topic in the last years [34]. [32] compares different estimation methods including the MHE, the extended and unscented Kalman filter as well as the particle filter. It is shown that the MHE can capture nonlinearities better than the EKF resulting in lower estimation error and better consistency of the filter. In [43] we have shown results from simualted flight data which showed promising performance and significant improvements over the EKF approach.

The goal of the paper is to extend and validate the MHE approach shown in [43] using experimental flight data from two different airframes. The estimator presented in [43] has been extended by a novel measurement model, which provides a highly accurate airspeed and angle of attack estimate, from the pitot-static tube and accelerometer measurements. In order to be usable with real sensor data, outlier removal and scaling errors of the pitot-static tube have been taken into account.

In comparison to [18] the estimator proposed in this work is better capable of estimating time varying wind velocities which is key to improve the performance of the angle of attack estimation.

The MHE estimates the wind velocities, and the coefficients of an aerodynamic model. These estimates can be used to calculate the airspeed, the angle of attack and the angle of sideslip. In order to avoid additional sensors this is done only with a standard autopilot sensor suite consisting of a global navigation satellite system (GNSS) receiver, an inertial measurement unit (IMU), a three-axis magnetometer and a pitot-static tube. It is assumed that an attitude and heading reference system (AHRS) and a translational motion observer (TMO) are already existing and are therefore only briefly discussed in this paper. Estimation performance will be shown for two different airframes each having two different sensor configurations of different quality. This allows an assessment on how the sensor quality affects estimation performance.

A disadvantage of the MHE is the increasing computational cost with increasing window length. We therefore have used a direct collocation method for NLP construction and an unscented Kalman filter for arrival cost approximation. This allows a short window length and creates a sparse NLP which is efficient to solve [20], [7], [13], [30].

II. Methods

A. Modeling

As a first step a model of the UAV is established. The modeling process can be separated into four main sub-models: A kinematic model describing the relationship between relative velocity, wind velocity and velocity over ground, an airspeed measurement model, an aerodynamic model describing the UAV and a stochastic wind model used for turbulence prediction. These models will be described in detail in the following subsections.

1) Kinematic Model: In the following we indicate variables decomposed in body coordinate frame with ^{b} and variables decomposed in inertial coordinate frame with ^{n}. The wind triangle [5] is given by

$$\boldsymbol{v}_r^b = \boldsymbol{v^b} - \boldsymbol{R}_n^b \boldsymbol{v}_w^n \tag{1}$$

It describes the relationship between the velocity of the aircraft relative to the surrounding airmass $\boldsymbol{v}_r^b = \begin{bmatrix} u_r & v_r & w_r \end{bmatrix}^T$, its ground velocity $\boldsymbol{v}^b = \begin{bmatrix} u & v & w \end{bmatrix}^T$, both decomposed in body frame and the local wind velocity $\boldsymbol{v}_w^n$, decomposed in inertial frame. $\boldsymbol{R}_n^b$ is the rotation matrix from inertial to body frame. Knowledge of the relative airspeed allows for the calculation of the air data parameters:

- Airspeed
$$V_a = \|\boldsymbol{v}_r^b\| \tag{2}$$
- Angle of Attack
$$\alpha = \tan^{-1}\left(\frac{w_r}{u_r}\right) \tag{3}$$
- Sideslip Angle
$$\beta = \sin^{-1}\left(\frac{v_r}{V_a}\right) \tag{4}$$

The following measurements are available:

- An estimate of the ground velocity $\tilde{\boldsymbol{v}}^{\boldsymbol{n}}$ in the inertial frame given by the TMO driven by GNSS velocity measurements.
- An estimate of the rotation matrix $\tilde{\boldsymbol{R}}_{\boldsymbol{n}}^{\boldsymbol{b}}$ given by the AHRS.
- A measurement of the airspeed $\tilde{V}_a^m$ given by the pitot-static tube measurement.

2) Airspeed Measurement Model: Research conducted in [14] has shown that for small angle of attack, sideslip-angle and misalignment angle the pitot-static tube still measures the airspeed (i.e. the length of the relative air velocity vector). The maximum angle to the air stream for which this assumption is true depends on the diameter, the geometry of the probe and the airspeed. This angle varies typically between $\pm 10^\circ$ and $\pm 60^\circ$ (see [14, pp. 41-44]). In the following we will assume this assumption to be fulfilled, due to the regarded flight scenario (airspeed far below the speed of sound, no

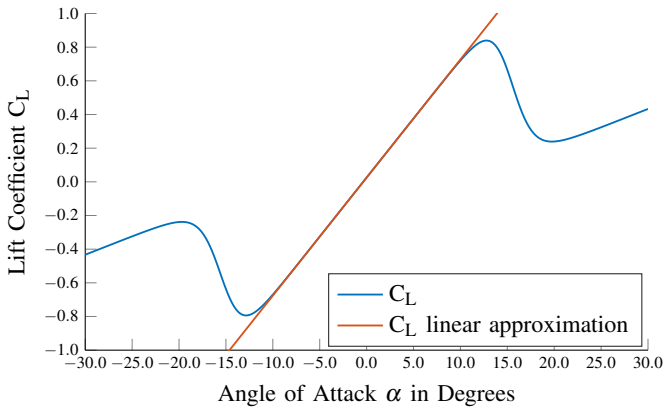

Fig. 1: Linear approximation of the aerodynamic model

stall). A pitot-static probe measures the difference between the dynamic pressure at the tip of the probe and the static pressure at the side of the probe. The indicated airspeed, which we will denote as $\tilde{V}_a^m$, is a function of the differential pressure ΔP and air density ρ.

$$\tilde{V}_a^m = \sqrt{\frac{2\Delta P}{\rho}} \tag{5}$$

Air density is dependent on altitude, temperature and atmospheric pressure and therefore can vary throughout a flight, creating errors in the airspeed measurement. The airspeed measurement is obtained from the autopilot where a calibration can be made pre-flight, yielding an estimate of the air density $\hat{\rho}$. However variations of air density during the flight are not always compensated for. Additionally, the airspeed measurement is effected by the positioning of the probe. These are caused by errors in the measurement of the static pressure, which ideally would measure the pressure of the undisturbed airflow. However due to restrictions on mounting and length of the used probe the static pressure measurement can be disturbed by the fuselage and wings of the aircraft which might also be dependent on the sideslip angle and the angle of attack.

We therefore propose the following measurement model:

$$V_a = \frac{\tilde{V}_a^m - \eta_{V_a}}{\gamma} = \frac{V_a^m}{\gamma} \tag{6}$$

with $\gamma = \sqrt{\frac{\hat{\rho}}{\rho}}$.

The airspeed measurement $\tilde{V}_a^m$ is assumed to be effected by Gaussian white measurement noise (η_{V_a}), caused by sensor noise and position errors as well as calibration errors due to air density changes of the probe and other scale factor errors caused by positioning and sensor errors, which are compensated by a calibration factor γ, which will be estimated. In the following V_a^m will be used for the uncalibrated airspeed.

3) Aerodynamic Model: By analyzing (1) it is immanent that the wind velocity vector is only fully observable, when rotations of the aircraft are performed. For the case of time varying wind velocity this would mean that sufficiently frequent rotations would have to be performed to achieve persistence of excitation. This issue has been discussed in more detail in [18].

To overcome this issue we proposed in [44] to use a lift force model in order to ensure observability of the angle of attack, resulting in observability of the relative airspeed in vertical direction in body coordinate frame. To do this we use a simplified model of the aerodynamic force in vertical direction:

$$\frac{F_z}{m} = f_z = -\frac{1}{2} V_a^2 \rho \frac{S}{m} \left(C_{L,0} + \alpha C_{L,\alpha}\right) + \nu_a \tag{7}$$

where m is the aircraft's mass, f_z is the specific force in vertical direction measured by an accelerometer, ρ is air density, S is the surface area of the airfoil, $C_{L,0}$ is the constant lift coefficient, $C_{L,\alpha}$ is the linear lift coefficient and ν_a is the modeling error. This model is a linearization with respect to α of the full nonlinear model of the aerodynamic lift found for example in [5, chap.4]. It neglects influences of the pitch rate and the elevator deflection on the lift force which are normally one order of magnitude smaller than the lift force created by the airfoil. The modeling error is small for normal flying conditions below stall angle (cf. Fig 1). In the estimator the model (7) is reformulated to:

$$f_z = -V_a^2 \frac{\rho S}{2m} \left(C_{L,0} + \alpha C_{L,\alpha}\right) \tag{8}$$

As discussed in Section II-A2 the air density might vary during flight. Since the uncalibrated airspeed will be effected by the inverse of this error, we can compensate for that by reformulating (8) to:

$$f_z = -\left(V_a^m\right)^2 K \left(C_{L,0} + \alpha C_{L,\alpha}\right) \tag{9}$$

with $K = \frac{\hat{\rho} S}{2m}$.

4) Dryden Wind Model: The wind velocity is modeled assuming frequency separation. The wind velocity $\boldsymbol{v}_w^n$ is split in a steady, slowly time varying part $\boldsymbol{v}_s^n = \begin{bmatrix} u_s^n & v_s^n & w_s^n \end{bmatrix}^T$ and a turbulent part $\boldsymbol{v}_t^n = \begin{bmatrix} u_t^n & v_t^n & w_t^n \end{bmatrix}^T$.

The steady wind velocity component is assumed to be slowly time varying,

$$\dot{\boldsymbol{v}}_{s,k}^{\boldsymbol{n}} \approx 0 \tag{10}$$

while the turbulent wind velocity component is described by the discrete time Dryden wind model specified in [28],

$$\boldsymbol{v}_{t,k+1}^n = \boldsymbol{v}_{t,k}^n - \Delta T V_{a,k} \begin{pmatrix} \frac{u_t^n}{L_u} \\ \frac{v_t^n}{L_v} \\ \frac{w_t^n}{L_w} \end{pmatrix} \Bigg|_k + \begin{pmatrix} \sigma_u \sqrt{2\Delta T \frac{V_a}{L_u}}\, \nu_{u_t} \\ \sigma_v \sqrt{2\Delta T \frac{V_a}{L_v}}\, \nu_{v_t} \\ \sigma_w \sqrt{2\Delta T \frac{V_a}{L_w}}\, \nu_{w_t} \end{pmatrix} \Bigg|_k \tag{11}$$

where ΔT is the sampling interval, k is the current time step, $\boldsymbol{\nu}_{\boldsymbol{x},k} = \begin{bmatrix} \nu_{u_t} & \nu_{v_t} & \nu_{w_t} \end{bmatrix}^T$ is Gaussian white noise, and the spatial wavelengths are given by:

$$L_{u,k} = L_{v,k} = \frac{h_k}{\left(0.177 + 0.0027 \cdot h_k\right)^{1.2}} \tag{12}$$

$$L_{w,k} = h_k \tag{13}$$

where h_k is the altitude above ground. The noise amplitudes are given by:

$$\sigma_{u,k} = \sigma_{v,k} = V_{w,G} \frac{1}{\left(0.177 + 0.0027 \cdot h_k\right)^{0.4}} \tag{14}$$

$$\sigma_w = 0.1 \cdot V_{w,G} \tag{15}$$

$V_{w,G}$ denotes the wind velocity measured 6 meters above ground.

B. State Space Model

The estimation problem can be divided in a state and a parameter estimation part. The states to be estimated are the turbulent wind velocities in inertial frame, the parameters to be estimated are the steady wind velocities in inertial frame, the two lift coefficients, and a scaling factor γ governing pitot-static tube calibration:

$$\boldsymbol{x} = \begin{bmatrix} u_t^n & v_t^n & w_t^n \end{bmatrix}^T \tag{16}$$

$$\boldsymbol{p} = \begin{bmatrix} u_s^n & v_s^n & w_s^n & KC_{L_0} & KC_{L_\alpha} & \gamma \end{bmatrix}^T \tag{17}$$

Inputs to the observer are the estimates of the velocity over ground, altitude and attitude given by a quaternion $\boldsymbol{q}_n^b$ describing the rotation from inertial frame to body frame [10]. $\boldsymbol{R}_n^b$ denotes the corresponding rotation matrix. These inputs are estimated by a translational motion observer (TMO) and an attitude and heading reference system (AHRS), based on IMU, GNSS and barometric pressure measurements.

$$\tilde{\boldsymbol{u}} = \begin{bmatrix} \boldsymbol{v}^b & \boldsymbol{q}_n^b & h \end{bmatrix}^T \tag{18}$$

Popular approaches for attitude and heading estimation are the EKF and nonlinear observers with a global or semi-global region of attraction [5], [19], [16]. These approaches provide locally stable (in case of the EKF) or (semi-) globally stable estimates of the attitude (in case of the nonlinear observer) while filtering sensor noise. For tuning of the wind velocity and parameter estimator we neglect noise on the attitude estimation. The altitude is affected by noise on the RTK-GNSS measurements, however this noise has a minor influence since the altitude is only used to calculate the spatial wavelength of the wind gusts in (12)-(13) which is insensitive to small variations. To reduce noise levels on both the velocity over ground and the altitude inputs a translational motion observer (TMO) is applied (i.e. [5], [16]). The system setup with AHRS and TMO is shown in Figure 2. In the figure $\breve{\boldsymbol{f}}$ is the specific force and $\breve{\boldsymbol{\omega}}$ is the angular rate both measured by the IMU, $\breve{\boldsymbol{\psi}}$ is a heading reference vector and $\breve{\boldsymbol{p}}^{\boldsymbol{n}}$ is the position and $\breve{\boldsymbol{v}}^{\boldsymbol{n}}$ the velocity over ground in inertial frame measured by the GNSS receiver.

In the following we will summarize the input noise in the vector $\eta_{\boldsymbol{u}}$ and define $\boldsymbol{u}_k$ as the system input at time step k.

$$\tilde{\boldsymbol{u}}_k = \begin{bmatrix} \tilde{\boldsymbol{v}}^b \\ \boldsymbol{q}_n^b \\ h \end{bmatrix}_k = \begin{bmatrix} \boldsymbol{v}^b + \boldsymbol{\eta}_{v^b} \\ \boldsymbol{q}_n^b \\ h \end{bmatrix}_k = \boldsymbol{u}_k + \eta_u \tag{19}$$

The state transition function is given by:

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) = -\| \boldsymbol{v}^b - \boldsymbol{R}_n^b (\boldsymbol{v}_s^n + \boldsymbol{v}_t^n) \| \begin{pmatrix} \frac{u_t}{L_u} \\ \frac{v_t}{L_v} \\ \frac{w_t}{L_w} \end{pmatrix} \tag{20}$$

The process noise input function is given by:

$$\boldsymbol{w}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\eta}_{\boldsymbol{x},k}, \boldsymbol{p}) = \begin{pmatrix} \sigma_u \sqrt{2\Delta T \frac{V_a}{L_u}} \nu_{u_t} \\ \sigma_v \sqrt{2\Delta T \frac{V_a}{L_v}} \nu_{v_t} \\ \sigma_w \sqrt{2\Delta T \frac{V_a}{L_w}} \nu_{w_t} \end{pmatrix} \Bigg|_{\substack{\boldsymbol{u}_k \\ \boldsymbol{x}_k \\ \boldsymbol{p}_k}} \tag{21}$$

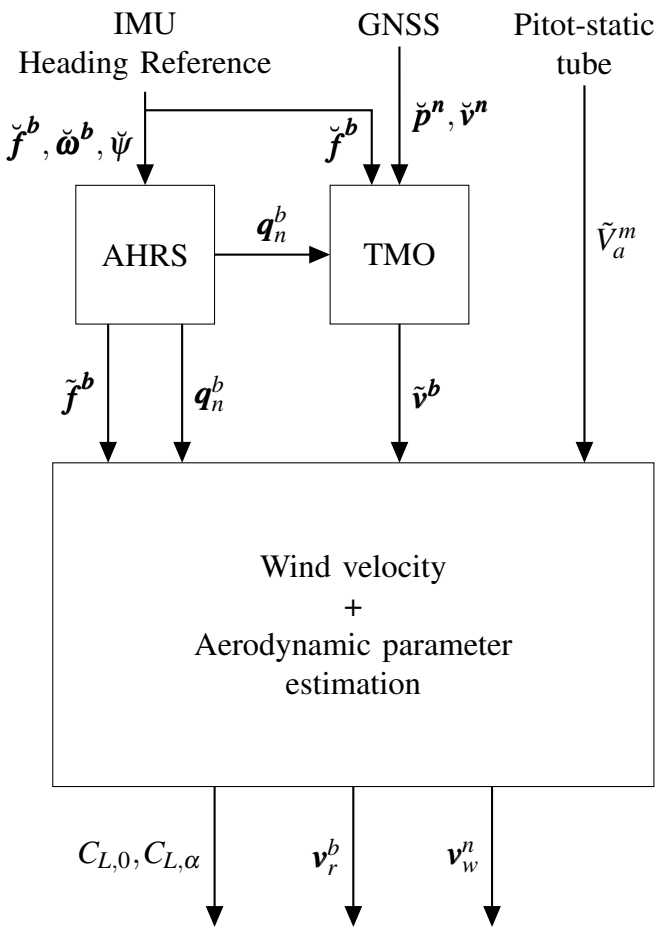

Fig. 2: Estimator structure

with $\boldsymbol{\nu}_{\boldsymbol{x},k} = \begin{bmatrix} \nu_{u_t} & \nu_{v_t} & \nu_{w_t} \end{bmatrix}^T$ and V_a defined as in (2). The predicted state is then given by:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \Delta T \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{p}_k) + \boldsymbol{w}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\nu}_{\boldsymbol{x},k}, \boldsymbol{p}_k) \tag{22}$$

For the parameters we assume slowly time varying behavior:

$$\boldsymbol{p}_{k+1} = \boldsymbol{p}_k + \boldsymbol{\nu}_{\boldsymbol{p},k} \tag{23}$$

where $\boldsymbol{\nu}_{\boldsymbol{p}}$ is a Gaussian distributed white process noise variable. The covariance matrix of the combined process noise $\boldsymbol{\nu}_{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{\nu}_{\boldsymbol{x}} & \boldsymbol{\nu}_{\boldsymbol{p}} \end{bmatrix}$ is given by $\boldsymbol{Q}$.

We can use the definitions given in equations (2)-(4) to represent the relative airspeed as:

$$\boldsymbol{v}_r^b = V_a \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix} \tag{24}$$

Here α can be estimated using the model of the vertical aerodynamics described by (9). Estimation of β through an aerodynamic model is more challenging since in the lateral direction forces originate from a number of different sources greatly increasing the number of coefficients to be estimated (see [5]) making it hard to achieve persistence of excitation. Under normal flying conditions, β can be assumed to be small so that we can approximate $\cos\beta \approx 1$ and $\sin\beta \approx 0$. The

approximation error is accounted for in the noise variable $\boldsymbol{\nu}_{ki}$. Hence, we get together with (1)

$$\begin{bmatrix} u^b \\ w^b \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \boldsymbol{R}_n^b \boldsymbol{v}_w^n + V_a \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} + \boldsymbol{\nu}_{ki} \tag{25}$$

Measurements used for correcting the predicted state vector are the specific force in vertical direction in body frame $\tilde{f}_z$ measured by an accelerometer, the airspeed measured by the pitot-static tube and the longitudinal and vertical velocity over ground in body frame $\tilde{u}^b$ and $\tilde{w}^b$.

The outputs of the system, velocity over ground, specific force and airspeed measurements are assumed to be affected by additive white Gaussian measurement noise.

$$\tilde{\boldsymbol{z}}_k = \begin{bmatrix} \tilde{f}_z \\ \tilde{V}_a^m \\ \tilde{u}^b \\ \tilde{w}^b \end{bmatrix} = \begin{bmatrix} f_z + \eta_{f_z} \\ V_a^m + \eta_{V_a} \\ u^b + \eta_{u^b} \\ w^b + \eta_{w^b} \end{bmatrix} = \boldsymbol{z}_k + \boldsymbol{\eta}_z \tag{26}$$

Since the velocity over ground is both an input and an output to the system, input and output noise variables are combined in a measurement noise vector $\boldsymbol{\eta}_m = \begin{bmatrix} \eta_{v^b} & \eta_{V_a} & \eta_{f_z} \end{bmatrix}^T$ with a covariance matrix $\boldsymbol{W}_m$.

The time varying measurement function is given by:

$$\boldsymbol{h}(\boldsymbol{x_k}, \boldsymbol{u_k}, \boldsymbol{p}) = \begin{bmatrix} -K\left(V_a^m\right)^2 \left(C_{L_0} + C_{L_\alpha}\alpha\right) \\ V_a \gamma \\ \boldsymbol{d}_1 \boldsymbol{R}_n^b (\boldsymbol{v}_s^n + \boldsymbol{v}_t^n) + V_a \cos(\alpha) \\ \boldsymbol{d}_3 \boldsymbol{R}_n^b (\boldsymbol{v}_s^n + \boldsymbol{v}_t^n) + V_a \sin(\alpha) \end{bmatrix} + \boldsymbol{\nu}_y \tag{27}$$

with:

$$\boldsymbol{d}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$
$$\boldsymbol{d}_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

The first measurement equation in (27) uses the z-accelerometer together with the aerodynamic model (9) to estimate the aerodynamic coefficients and the wind velocities via V_a and α. The second measurement equation $V_a^m = \gamma V_a$ uses the definition of the airspeed (2) and the calibration factor γ to relate the measured airspeed to the wind velocities. The third and fourth measurement equations utilizes the wind triangle (1) to relate the measurements of the velocity over ground to the relative longitudinal velocity u_r^m measured by the pitot-static tube. This is similar to the method described in [18].

The approximation errors within the models used in the measurement function are summarized in the noise variable $\boldsymbol{\nu_y} = \begin{bmatrix} \nu_a & \nu_v & \boldsymbol{\nu}_{ki} \end{bmatrix}$ which is assumed to be white Gaussian noise with a covariance matrix $\boldsymbol{R}$.

We have chosen to separate noise variables for the measurement noise of input and output variables ($\boldsymbol{\eta}_m$) and the process noise resulting from model mismatches in the state transition ($\boldsymbol{\nu}_q$) and output functions ($\boldsymbol{\nu}_y$). The main motivation for this noise modeling is ease of tuning of the estimator since measurement noise covariances can be chosen according to sensor specifications, whereas the covariance matrices of the process noise can be chosen according to the magnitude of the errors in the aerodynamic (9) and kinematic (1) model.

As discussed in [44] and [18] the system is not uniformly observable. Therefore, attitude changes are necessary in order to ensure persistence of excitation.

C. Moving Horizon Estimation

The Moving Horizon Estimator (MHE) [20], [31] uses the sensor measurements gathered in a data window containing L samples to estimate the states $\boldsymbol{x}$ and parameters $\boldsymbol{p}$ of a system. These measurements can be interpreted as a series of inputs $\boldsymbol{u}_j$ and outputs $\boldsymbol{z}_j$ of a state space model of the system. This system is affected by the process noise $\boldsymbol{\nu_j} = \begin{bmatrix} \boldsymbol{\nu}_{q,j} & \boldsymbol{\nu}_{y,j} \end{bmatrix}$ as well as input measurement noise $\boldsymbol{\eta}_{\boldsymbol{u},j}$ and output measurement noise $\boldsymbol{\eta}_{\boldsymbol{z},j}$, summarized in a measurement noise vector $\boldsymbol{\eta}_{m,j}$. The goal of the MHE is it to minimize the error between the model output and the measurements while fulfilling continuity conditions given by the state transition function. To estimate the system states, system parameters, and noise parameters, the following nonlinear program is solved in each time step:

$$\min_{\substack{\boldsymbol{x}_{k-L},\ldots,\boldsymbol{x}_k \\ \boldsymbol{\eta}_{k-L},\ldots,\boldsymbol{\eta}_k \\ \boldsymbol{\nu}_{k-L},\ldots,\boldsymbol{\nu}_k \\ \boldsymbol{\theta}_{k-L,0},\ldots,\boldsymbol{\theta}_{k,d} \\ \boldsymbol{p}_{k-L},\ldots,\boldsymbol{p}_k}} \left(\left\| \begin{matrix} \boldsymbol{x}_{k-L} - \hat{\boldsymbol{x}}_{k-L} \\ \boldsymbol{p}_{k-L} - \hat{\boldsymbol{p}}_{k-L} \end{matrix} \right\|^2_{\hat{\boldsymbol{P}}_{k-L}^{-1}} + \cdots \right.$$
$$\left. \sum_{j=k-L}^{k} \left\| \tilde{\boldsymbol{z}}_{\boldsymbol{j}} - \boldsymbol{\eta}_{\boldsymbol{z},j} - \boldsymbol{h}(\boldsymbol{x}_j, \tilde{\boldsymbol{u}}_{\boldsymbol{j}} - \boldsymbol{\eta}_{\boldsymbol{u},j}, \boldsymbol{p}_j) \right\|^2_{\boldsymbol{V}} + \sum_{j=k-L}^{k-1} \left\| \boldsymbol{\eta}_j \right\|^2_{\boldsymbol{W}} \right) \tag{28}$$

where $\boldsymbol{\eta}_j = \begin{bmatrix} \boldsymbol{\nu}_{\boldsymbol{x},j} & \boldsymbol{\nu}_{\boldsymbol{p}_j} & \boldsymbol{\eta}_{\boldsymbol{m},j} \end{bmatrix}$. This NLP is subject to equality and inequality constraints. The equality constraints are continuity constraints necessary to capture the system dynamics. This is done using a direct collocation method which uses parameter vectors $\boldsymbol{\theta}_j$ to subsample the states at collocation points between the sampling instants (see Section II-C1).

Note that all norms are 2-norms that are weighted by a matrix indicated in the subscript of the norm i.e. $\|\boldsymbol{x}\|^2_{\boldsymbol{A}} = \boldsymbol{x}^T \boldsymbol{A} \boldsymbol{x}$. The first term in the penalty function is the so called arrival cost which summarizes the information before the current estimation window in a state estimate $\hat{\boldsymbol{x}}_{k-L}$, a parameter estimate $\hat{\boldsymbol{p}}_{k-L}$ and a error covariance matrix $\hat{\boldsymbol{P}}_{k-L}$. The arrival cost is key to the stability of the MHE [20], [31] and will therefore be discussed in more detail in section II-C2. The second term penalizes deviations of the output of the state space model from the measurement $\boldsymbol{z}_j$ with a weighting matrix $\boldsymbol{V}$. The last term penalizes the noise vector in the state transition function with a weight matrix $\boldsymbol{V}$. Following Robertson et.al [34], we use the inverse of the measurement and process noise covariance matrices as weights and adjust for the sampling frequency.

$$\boldsymbol{V} = \boldsymbol{R}^{-1} \tag{29}$$

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{Q}^{-1} & \boldsymbol{0}_{[9x5]} \\ \boldsymbol{0}_{[5x9]} & \boldsymbol{W}_m^{-1} \end{bmatrix} \Delta T^{-1} \tag{30}$$

The rationale behind this choice, follows from the assumption of Gaussian white measurement noise. By choosing the inverses of the noise covariance matrices as weights, we ensure that the values of $\boldsymbol{x}_j$ and $\boldsymbol{p}_j$ which minimize (28) are *maximum-aperiori* estimates of the states and parameters at each time step [34].

1) Direct Collocation: In order to capture the system dynamics, continuity constraints have to be added to the nonlinear program given in equation (28). In this paper we chose a direct collocation formulation for the states in the

nonlinear program, proposed for model predictive control in [6], [41]. This formulation adds additional degrees of freedom to the states, representing the high frequent turbulent wind velocities, by using d additional collocation points τ_i between each time step j, given by the roots of a Legendre polynomial of order d. These roots are denoted as $\mathscr{L}_1 \dots \mathscr{L}_d$. The trajectory of the state transition and the noise transfer functions are then approximated on the interval $[t_j, t_{j+1}]$ by Lagrange polynomials $\Gamma_i(t)$ with weight vector $\boldsymbol{\theta}_{j,i}$. On each collocation point the gradient condition (31) and at the end of an collocation interval the continuity condition (32), is added to the equality constraints of the NLP. Additionally equality constraints (33) - (34) ensure that the weight vector matches the state vector at the beginning and end of the collocation interval. The equality constraints for the parameters are only enforced at the sampling instants, since the parameters are assumed to be only slowly time varying.

This results in the following equality constraints:

$$\sum_{r=0}^{d} \dot{\boldsymbol{\Gamma}}_r(\tau_i)\boldsymbol{\theta}_{j,r} - \boldsymbol{f}(\boldsymbol{\theta}_{j,i}, \boldsymbol{u}_j, \boldsymbol{p}_j) + \tfrac{1}{\Delta T}\boldsymbol{w}(\boldsymbol{\theta}_{j,i},, \boldsymbol{u}_j, \boldsymbol{\nu}_{\boldsymbol{x}_j}, \boldsymbol{p}_j) = \boldsymbol{0} \tag{31}$$

$$\sum_{i=0}^{d} \boldsymbol{\Gamma}_i(1)\boldsymbol{\theta}_{j,i} - \boldsymbol{\theta}_{j+1,0} = \boldsymbol{0} \tag{32}$$

$$\boldsymbol{x}_j - \boldsymbol{\theta}_{j,0} = \boldsymbol{0} \tag{33}$$

$$\boldsymbol{x}_{j+1} - \boldsymbol{\theta}_{j,d} = \boldsymbol{0} \tag{34}$$

$$\boldsymbol{p}_{j+1} - \boldsymbol{p}_j - \boldsymbol{\nu}_{\boldsymbol{p}_j} = \boldsymbol{0} \tag{35}$$

$$\text{for } i = 0, \dots, d \text{ and } j = k-L, \dots, k-1$$

where:

$$\boldsymbol{\tau} = \begin{bmatrix} 0 & \mathscr{L}_1 & \dots & \mathscr{L}_d \end{bmatrix} \tag{36}$$

$$\boldsymbol{\Gamma}_i(t) = \prod_{r=0, r\neq i}^{d} \frac{t - \tau_r}{\tau_i - \tau_r} \tag{37}$$

The result is a very large but also very sparse NLP, which can be solved efficiently by dedicated NLP solvers. We chose to use IPOPT [42] as the solver since it was readily available in Casadi which we use for algorithmic differentiation.

One of the benefits of the MHE is the possibility to impose constrains on the estimated states, noise variables and parameters. We will only impose constraints on the estimated lift coefficients and the pitot-static tube calibration factor since bounds for these parameters can be roughly estimated, whereas wind velocity bounds are unknown. Additionally the angle of attack is constrained in order to force a unique solution of the trigonometric functions used in the model, resulting in the following inequality constraints:

$$\begin{aligned} \boldsymbol{p}_{min} \leq \boldsymbol{p} \leq \boldsymbol{p}_{max} \\ -45^\circ \leq \alpha \leq 45^\circ \end{aligned} \tag{38}$$

2) Arrival Cost Approximation: As mentioned in section II-C, the arrival cost term is important for the stability of the estimator [20], [31]. Ideally one would like to use the exact arrival cost, i.e. (28) is writen as:

$$\min \sum_{j=0}^{k} \|\boldsymbol{y}_j - \boldsymbol{h}(\boldsymbol{x}_j, \boldsymbol{p}, \boldsymbol{u}_j)\|^2_{\boldsymbol{V}} + \sum_{j=0}^{k} \|\boldsymbol{\eta}_j\|^2_{\boldsymbol{W}} \tag{39}$$

However with increasing k, the amount of data to store, and the computation time needed to solve this problem, rises drastically and can therefore not be implemented. To solve this problem several approaches have been proposed. Lopez-Negrete et.al. [25] considers different Kalman filter type of approaches including the EKF and the Unscented Kalman Filter (UKF), as well as the Particle Filter (PF). It was demonstrated that filters that avoid linearizations, like the UKF and the PF, have significant performance advantages compared to the EKF when approximating the arrival cost, allowing the use of shorter window lengths and thus compensating for the higher computational cost of these filters.

In this paper the UKF approach to arrival cost approximation was used, as described in [30]. The arrival cost is updated in the following way: We start with an augmented state vector $\boldsymbol{x}^a_{v-1|v-1}$ which includes the optimal state estimates at the start of the moving horizon ($\boldsymbol{x}^*_{v-1|v-1}$) and the optimal parameter $\boldsymbol{p}^*_{v-1}$ estimates of the previous solution of the NLP (28), with $v = k-L$. The augmented state vector includes also the first moments of the process, measurement and input noise distributions, which are, following the previous assumptions, equal to zero.

$$\boldsymbol{x}^a_{v-1|v-1} = \begin{bmatrix} \boldsymbol{x}^*_{v-1|v-1} & \boldsymbol{p}^*_{k-1} & \boldsymbol{0}_{q_x} & \boldsymbol{0}_{q_p} & \boldsymbol{0}_r & \boldsymbol{0}_u \end{bmatrix}^T \tag{40}$$

The corresponding covariance matrix of the estimation error is:

$$\boldsymbol{P}^a_{v-1|v-1} = \begin{bmatrix} \hat{\boldsymbol{P}}_{v-1|v-1} & \boldsymbol{0}_{[9x9]} & \boldsymbol{0}_{[9x2]} & \boldsymbol{0}_{[9x5]} \\ \boldsymbol{0}_{[9x9]} & \boldsymbol{Q}\Delta T & \boldsymbol{0}_{[9x2]} & \boldsymbol{0}_{[9x5]} \\ \boldsymbol{0}_{[2x9]} & \boldsymbol{0}_{[2x9]} & \boldsymbol{R} & \boldsymbol{0}_{[2x5]} \\ \boldsymbol{0}_{[5x9]} & \boldsymbol{0}_{[5x9]} & \boldsymbol{0}_{[5x2]} & \boldsymbol{W}_m\Delta T \end{bmatrix} \tag{41}$$

We then generate a set of sigma points:

$$\begin{aligned} &\boldsymbol{\chi}^a_{v-1} = \boldsymbol{X}^a_{v-1|v-1} + \cdots \\ &\left(\boldsymbol{0}_{[24x1]}, \quad \sqrt{(n^a + \kappa)\boldsymbol{P}^a_{v-1|v-1}}, \quad -\sqrt{(n^a + \kappa)\boldsymbol{P}^a_{v-1|v-1}}\right) \end{aligned} \tag{42}$$

and corresponding weights:

$$\Omega_i = \begin{cases} \frac{\kappa}{n_a + \kappa} & \text{if } i = 1 \\ \frac{1}{2(n_a + \kappa)} & \text{otherwise} \end{cases} \tag{43}$$

where n^a is the size of $\boldsymbol{x}^a$ which is in this case 24 and κ is a tuning parameter set to -21 in accordance with [30]. $\boldsymbol{X}^a_{v-1|v-1}$ is the expanded $n^a \times (2n^a + 1)$ matrix with $\boldsymbol{x}^a_{v-1|v-1}$ as each column. We then use the state space model for prediction:

$$\begin{aligned} \boldsymbol{\chi}^x_v = \quad & \boldsymbol{\chi}^x_v + \Delta T \boldsymbol{f}(\boldsymbol{\chi}^x_{v-1}, \tilde{\boldsymbol{u}}_{v-1} - \boldsymbol{\chi}^u_{v-1}, \boldsymbol{\chi}^p_{v-1}) + \cdots \\ & \boldsymbol{w}(\boldsymbol{\chi}^x_{v-1}, \tilde{\boldsymbol{u}}_{v-1} - \boldsymbol{\chi}^u_{v-1}, \boldsymbol{\chi}^{q_x}_{v-1}, \boldsymbol{\chi}^p_{v-1}) \end{aligned} \tag{44}$$

$$\boldsymbol{\chi}^p_v = \quad \boldsymbol{\chi}^p_{v-1} + \boldsymbol{\chi}^{q_p}_{v-1} \tag{45}$$

$$\boldsymbol{\gamma}_v = \quad \boldsymbol{h}(\boldsymbol{\chi}^x_v, \tilde{\boldsymbol{u}}_v - \boldsymbol{\chi}^u_{v-1}, \boldsymbol{\chi}^p_k) + \chi^r_{v-1} \tag{46}$$

With:

$$\boldsymbol{\chi}^a_{v-1} = \quad \begin{bmatrix} \boldsymbol{\chi}^x_{v-1} & \boldsymbol{\chi}^p_{v-1} & \boldsymbol{\chi}^{q_x}_{v-1} & \boldsymbol{\chi}^{q_p}_{v-1} & \boldsymbol{\chi}^r_{v-1} & \boldsymbol{\chi}^u_{v-1} \end{bmatrix}^T \tag{47}$$

Afterwards we can calculate the predicted state and parameter vectors as well as the predicted covariance matrix of the estimation error:

$$\boldsymbol{x}_{v|v-1} = \sum_{i=1}^{2n^a+1} \Omega_i \boldsymbol{\chi}^x_{i,v} = \hat{\boldsymbol{x}}_{k-L} \tag{48}$$

$$\boldsymbol{p}_{v|v-1} = \sum_{i=1}^{2n^a+1} \Omega_i \boldsymbol{\chi}^p_{i,v} = \hat{\boldsymbol{p}}_{k-L} \tag{49}$$

$$\hat{\boldsymbol{z}}_{\boldsymbol{v}} = \sum_{i=1}^{2n^a+1} \Omega_i \gamma_{i,v} \tag{50}$$

$$\boldsymbol{P}^{\boldsymbol{x}}_{v|v-1} = \sum_{i=1}^{2n^a+1} \Omega_i \left[\boldsymbol{\chi}^x_{i,v} - \boldsymbol{x}_{v|v-1}\right]\left[\boldsymbol{\chi}^x_{i,v} - \boldsymbol{x}_{v|v-1}\right]^T \tag{51}$$

$$\boldsymbol{P}_{y,v} = \sum_{i=1}^{2n^a+1} \Omega_i \left[\boldsymbol{\gamma}_{i,v} - \hat{\boldsymbol{y}}_v\right]\left[\boldsymbol{\gamma}_{i,v} - \hat{y}_v\right]^T \tag{52}$$

$$\boldsymbol{P}_{yz,v} = \sum_{i=1}^{2n^a+1} \Omega_i \left[\boldsymbol{\chi}^x_{i,v} - \boldsymbol{x}_{v|v-1}\right]\left[\boldsymbol{\gamma}_{i,v} - \hat{y}_v\right]^T \tag{53}$$

As a final step the covariance matrix of the estimation error is updated with the measurement uncertainty and is then used as weight for the arrival cost.

$$\boldsymbol{K}_v = \boldsymbol{P}_{xy,v}\boldsymbol{P}^{-1}_{y,v} \tag{54}$$

$$\boldsymbol{P}^{\boldsymbol{x}}_{v|v} = \boldsymbol{P}_{v|v-1} - \boldsymbol{K}_v\boldsymbol{P}_{y,v}\boldsymbol{K}^T_v = \hat{\boldsymbol{P}}_{k-L} \tag{55}$$

3) Outlier Detection: The estimation results can be disturbed by outliers, i.e. from GNSS measurements, resulting in wrong parameter estimates and inconsistent error covariance estimates. To detect these outliers we evaluate the estimation result of the MHE by comparing it with the predicted state of the arrival cost approximation. For this only the estimates of the aerodynamic coefficients and the pitot-tube scaling factor are used since these can be assumed to be constant, where as the steady wind velocities are slowly time varying. If the deviation of any of these elements of the optimal estimated parameter vector $\boldsymbol{p}^*$ from the predicted coefficient vector $\hat{p}_{k-L}$ is more than three times the estimated error standard deviation we consider $\boldsymbol{p}^*$ an outlier. Since the noise on the parameter estimates is a assumed to be Gaussian distribution, this test corresponds to a false detection probability of 0.1%.

$$\|\hat{p}_{j,k-L} - p^*_{j,k}\| > 3\sqrt{\hat{P}_{k-L,jj}} \quad \text{for } j = 4\ldots6 \tag{56}$$

In this case we will keep the estimate of the previous estimation step.

III. Experimental Setup

Flights with two different aircraft were performed, the Skywalker X8, and the Cruiser-Mini. Both were equipped with the same sensor payload and an autopilot. Aircraft parameters can be found in table I

A. Experimental Sensor Payload

In order to record input data for the estimator as well as a reference to compare the estimation results to, an experimental sensor payload was designed. This payload was used on both airframes and consists of the following sensor set:

- **GNSS receivers**: 3 ublox NEO M8T GNSS receivers were used to acquire postion data of the aircraft. Two of these receivers were connected to antennas on each wing $1.5m$ apart from each other. The third receiver was placed on the fuselage. Additionally a basestation with the same receiver was used to allow the use of real-time kinematic GPS (RTK-GPS) resulting in centimeter level position accuracy. RTKLib [39] was used as an implementation of RTK-GPS. The sampling frequency is $5Hz$. This setup allows the use of the two position measurements as a heading reference. In addition Doppler based velocity measurements are available.
- **IMU**: A Sensonor STIM 300 MEMS IMU was used, providing 3 axis acceleration and angular velocity measurements. The sampling frequency is $500Hz$.
- **Multi directional pitot-static tube**: The Aeroprobe micro Air data system was used. This pitot-static system calculates angle of attack, sideslip angle and airspeed from 5 dynamic pressure and one static pressure measurement. This data was used as a reference for the estimator. The Aeroprobe provides data at a frequency of $100Hz$, accuracy of the flow angle measurements is 1° and $1m/s$

TABLE I: Aircraft Parameters

	Skywalker X8	Magline Cruiser-Mini
Type of Aircraft	Tailless Flying-wing aircraft	T-tail aircraft
Weight	$4kg$	$8kg$
Endurance	45min	1.5h
Cruise Speed	$15m/s$	$17m/s$
Wing Span	$2.1m$	$2.6m$

Fig. 3: Cruiser-Mini

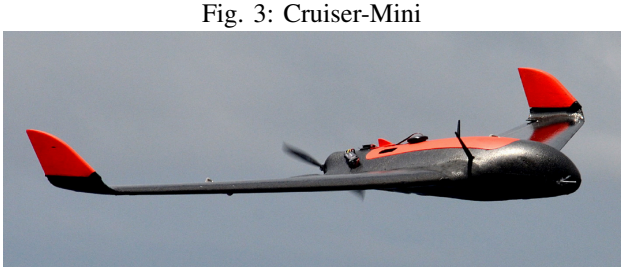

Fig. 4: Skywalker X8

for the airspeed measurements.

These sensors where synchronized and timestamped on a hardware level with high resolution. For more detail on the hardware and software realization of the sensor timing board we refer to [3].

B. Inertial Navigation

To estimate the attitude and translation motion of the aircraft from the sensor measurements, we use the nonlinear and linear observers described in detail in [9]. The Attitude and Heading Reference System (AHRS) uses the measurements obtained by the STIM IMU to estimate the rotation matrix between inertial and body fixed frame using a nonlinear observer (NLO) based on complimentary filtering on SO(3) [26]. The used NLO has first been presented in [15], [16]. [10] shows implementation details of the observer. The observer estimates attitude and heading by comparing specific force vectors in inertial and body frame for low-frequency accuracy while integrating the angular velocities for high frequency accuracy. Simultaneously gyro biases are estimated to compensate for sensor measuring errors. A heading reference is obtained from the RTK position solution of the two different GNSS receivers on the wings of the aircraft using the following equation:

$$\breve{\psi} = \text{atan2}\left(\breve{p}_{l,N} - \breve{p}_{r,N}, \breve{p}_{l,E} - \breve{p}_{r,E}\right) \tag{57}$$

where $\breve{\psi}$ is the heading reference and $\breve{p}_{l,N}$ is the north component of the position measurement of the left wing.

The AHRS has three tuning factors: k_1 and k_2 which are cut-off frequencies for low-frequency information from the accelerometers and the heading reference vector. For frequencies above these cut off frequencies the gyroscope information is used. The third tuning factor k_I governs gyro bias sensitivity.

As a Translational Motion Observer (TMO) we use the loosely coupled implementation described in [10] which results in gains given by a Riccati equation similar to the Kalman Filter. The loosely coupled TMO uses the tangent frame representation of the strapdown equation in order to estimate position and velocity from position, velocity and accelerometer measurements given an attitude estimate, using the following model:

$$\dot{\boldsymbol{p}}^n = \boldsymbol{v}^n \tag{58}$$

$$\dot{\boldsymbol{v}}^n = \boldsymbol{R}_b^n \boldsymbol{f}^b + \boldsymbol{g}^n \tag{59}$$

Position and velocity measurements are here obtained from the third GNSS receiver with the antenna placed on the fuselage.

Advantages of this modular observer design are low computational cost and uniform semi-global exponential stability.

C. Autopilot

Additionally data from the autopilot on both aircrafts was available. The autopilot provides another set of GNSS and IMU data as well as measurements from a single directional pitot-static tube and a 3 axis magnetometer. These sensors provide data with somewhat higher noise level, less accuracy and lower sampling frequencies compared to the experimental sensor payload.

The Skywalker X8 was equipped with a Pixhawk autopilot which provides the following sensor set:

- **GNSS receiver**: uBlox LEA-6H, sampled at $5Hz$.
- **IMU**: ST Micro L3GD20H 16 bit gyroscopes and ST Micro LSM303D 14 bit accelerometers, sampled at $25Hz$.
- **Magnetometer**: HMC5983
- **Pitot-static tube**: Measurement Specialties 4525DO sensor, sampled at $10Hz$.

The Cruiser-Mini was equipped with the newer Pixhawk 2 autopilot. In comparison to the earlier version the IMUs are temperature controlled and isolated from vibrations. The Pixhawk 2 provides data from the following sensor package:

- **GNSS receiver**: uBlox NEO-M8T sampled at $5Hz$.
- **IMU**: ST Micro L3GD20H 16 bit gyroscopes and ST Micro LSM303D 14 bit accelerometers, sampled at $25Hz$.
- **Magnetometer**: HMC5983
- **Pitot-static tube**: Measurement Specialties 4525DO sensor, sampled at $10Hz$.

For navigation an Extended Kalman Filter (EKF) is used to estimate position, velocity and attitude.

D. MHE Implementation and Tuning

The data collected from the test flights was analyzed offline in a Matlab environment. After passing through the AHRS and TMO, wind velocities and lift coefficients were estimated using the methods described in section II-C. Within Matlab, Casadi [4] was used for symbolic calculations and algorithmic differentiation. In order to speed up computation, C++ code and precompiled .mex functions were generated from the Matlab / Casadi code. The number of collocation points d was set to 5 and the window length L was set to 6 samples. Note that there is a trade-off between computation time and the window length L. A longer window length is expected to improve the estimation result, which might be desirable when using low cost sensors, however at the cost of a higher computation time. Limits for the coefficient estimates were set to:

$$\begin{bmatrix} -0.2 \\ 0 \\ -0.5 \end{bmatrix} \leq \begin{bmatrix} KC_{L,0} \\ KC_{L,\alpha} \\ \gamma \end{bmatrix} \leq \begin{bmatrix} 0.2 \\ 2 \\ 0.5 \end{bmatrix} \tag{60}$$

Since the pressure sensors of the pitot-static tube are only sampled at $10Hz$ according to the Shannon-Nyquist theorem, only frequency components up to $5Hz$ can be recorded. Spectral analysis of the Aeroprobe data showed that all major frequency components are below $5Hz$. Therefore, the sampling rate of the MHE was set to $5Hz$ as well. In order to further speed up computation we use the result of the previous time-step as a starting point for the current time-step by shifting the result vector by one time step. This is referred to as "warm starting" and is explained in more detail in [20].

During initialization the window length is gradually scaled up until L samples have been received, using the initial values for states, parameters and covariances as the arrival cost. Afterwards the arrival cost is approximated as described in Section II-C2.

Since motor vibrations induce significant oscillations on the accelerometers, their signals were passed through a FIR lowpass filter with a cut-off frequency of $20Hz$ and $60dB$ dampening at $50Hz$.

When tuning the estimator for a specific sensor set and airframe different aspects have to be considered:

- Initial Conditions: For the steady and turbulent wind velocities these can be set to zero. The corresponding initial covariances should be set low for the turbulent wind velocities and high for the steady wind velocities to achieve fast convergence. The initial value for the linear lift coefficient should be set to a non-zero value. The choice depends on the airframe used. If the wing area, the air density and the mass of the aircraft are known, an initial guess can be obtained using $K\hat{C}_{L,\alpha} = \frac{S\hat{\rho}}{m}\pi$ (cf. thin-airfoil theory [1, Chap. 4]). The inital value of the pitot-static tube calibration factor should be set to 1 and the corresponding covariance can be set low if the autopilot already provides a pitot-static tube calibration routine and higher if this is not he case.
- Input Noise Covariance Matrix: This matrix can be tuned according to sensor datasheet values. Note that in the here presented setup, the velocity over ground input is not the the GNSS measurement directly but the velocity estimate provided by the TMO. Since the TMO reduces noise levels significantly, we assume that the cross-correlation of the input noise is negligible and use covariance values which are one order of magnitude lower than when using the GNSS measurements directly. If it is desired to use the GNSS measurements directly one would need to consider the cross-correlated noise as well.
- Process Noise and Measurement Noise Matrix: These matrices have to be tuned with regard to the used airframe. In a less rigid airframe the covariance should be set higher, in a stiff airframe the values of the noise matrices can be set lower. These values are independent of the prevailing wind conditions.

The used values for these parameters can be found in tables II and III.

TABLE II: MHE Parameters: Skywalker X8

Parameter	Sensor Payload
$\boldsymbol{x}_0$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$
$\boldsymbol{p}_0$	$\begin{bmatrix}0 & 0 & 0 & 0 & 0.3 & 1\end{bmatrix}^T$
$\boldsymbol{P}_0$	$\operatorname{diag}\left(\begin{bmatrix}10^{-6} & 10^{-6} & 10^{-6} & 10^{-2} & \cdots \\ 10^{-2} & 10^{-6} & 10^{-5} & 10^{-5} & 10^{-5}\end{bmatrix}\right)$
$\boldsymbol{Q}$	$\operatorname{diag}\left(\begin{bmatrix}10^{-1} & 10^{-1} & 1 & 10^{-4} & \cdots \\ 10^{-4} & 10^{-6} & 10^{-15} & 10^{-30} & 10^{-15}\end{bmatrix}\right)$
$\boldsymbol{R}$	$\operatorname{diag}\left(\begin{bmatrix}1 & 10^{-4} & 10^{-2} & 10^{-2}\end{bmatrix}\right)$
$\boldsymbol{W}_m$	$\operatorname{diag}\left(\begin{bmatrix}10^{-6} & 10^{-6} & 10^{-6} & 1 & 10^{-10}\end{bmatrix}\right)$
Parameter	**Autopilot**
$\boldsymbol{x}_0$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$
$\boldsymbol{p}_0$	$\begin{bmatrix}0 & 0 & 0 & 0 & 0.3 & 1\end{bmatrix}^T$
$\boldsymbol{P}_0$	$\operatorname{diag}\left(\begin{bmatrix}10^{-6} & 10^{-6} & 10^{-6} & 10^{-2} & \cdots \\ 10^{-2} & 10^{-6} & 10^{-5} & 10^{-5} & 10^{-5}\end{bmatrix}\right)$
$\boldsymbol{Q}$	$\operatorname{diag}\left(\begin{bmatrix}10^{-1} & 10^{-1} & 1 & 10^{-4} & \cdots \\ 10^{-4} & 10^{-6} & 10^{-15} & 10^{-30} & 10^{-15}\end{bmatrix}\right)$
$\boldsymbol{R}$	$\operatorname{diag}\left(\begin{bmatrix}1.4 & 10^{-4} & 10^{-2} & 10^{-2}\end{bmatrix}\right)$
$\boldsymbol{W}_m$	$\operatorname{diag}\left(\begin{bmatrix}10^{-4} & 10^{-4} & 10^{-4} & 1 & 10^{-6}\end{bmatrix}\right)$

TABLE III: MHE Parameters: Cruiser-Mini

Parameter	Sensor Payload
$\boldsymbol{x}_0$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$
$\boldsymbol{p}_0$	$\begin{bmatrix}0 & 0 & 0 & 0 & 0.2 & 1\end{bmatrix}^T$
$\boldsymbol{P}_0$	$\operatorname{diag}\left(\begin{bmatrix}10^{-6} & 10^{-6} & 10^{-6} & 10^{-2} & \cdots \\ 10^{-2} & 10^{-6} & 10^{-5} & 10^{-5} & 10^{-5}\end{bmatrix}\right)$
$\boldsymbol{Q}$	$\operatorname{diag}\left(\begin{bmatrix}10^{-1} & 10^{-1} & 1 & 10^{-4} & \cdots \\ 10^{-4} & 10^{-6} & 10^{-15} & 10^{-30} & 10^{-15}\end{bmatrix}\right)$
$\boldsymbol{R}$	$\operatorname{diag}\left(\begin{bmatrix}1 & 10^{-4} & 10^{-2} & 10^{-2}\end{bmatrix}\right)$
$\boldsymbol{W}_m$	$\operatorname{diag}\left(\begin{bmatrix}10^{-6} & 10^{-6} & 10^{-6} & 1 & 10^{-10}\end{bmatrix}\right)$
Parameter	**Autopilot**
$\boldsymbol{x}_0$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$
$\boldsymbol{p}_0$	$\begin{bmatrix}0 & 0 & 0 & 0 & 0.2 & 1\end{bmatrix}^T$
$\boldsymbol{P}_0$	$\operatorname{diag}\left(\begin{bmatrix}10^{-6} & 10^{-6} & 10^{-6} & 10^{-2} & \cdots \\ 10^{-2} & 10^{-6} & 10^{-5} & 10^{-5} & 10^{-5}\end{bmatrix}\right)$
$\boldsymbol{Q}$	$\operatorname{diag}\left(\begin{bmatrix}10^{-1} & 10^{-1} & 1 & 10^{-4} & \cdots \\ 10^{-4} & 10^{-6} & 10^{-15} & 10^{-30} & 10^{-15}\end{bmatrix}\right)$
$\boldsymbol{R}$	$\operatorname{diag}\left(\begin{bmatrix}1.4 & 10^{-4} & 10^{-2} & 10^{-2}\end{bmatrix}\right)$
$\boldsymbol{W}_m$	$\operatorname{diag}\left(\begin{bmatrix}10^{-4} & 10^{-4} & 10^{-4} & 1 & 10^{-6}\end{bmatrix}\right)$

IV. Results

The data from these flights was analyzed off-line on a Laptop with a Core I7 CPU.

A. X8 Flights

1) Flight Path: We performed two different flights with the Skywalker X8. Flight 1 consists of two parts. In the first part we follow a race track pattern and in the second part a figure eight pattern. These maneuvers include narrow and wide turns with different bank angles. In the last part of the flight the aircraft is flown manually in varying altitudes. The goal of this flight plan is to excite the estimator and assess its performance in favorable circumstances. The aircraft's path and altitude are shown in Figures 5 and 6.

In the second flight a race track pattern is flown at a higher altitude of $90m$ above ground. During this flight the airspeed was varied between $14m/s$ and $26m/s$ to vary the angle of

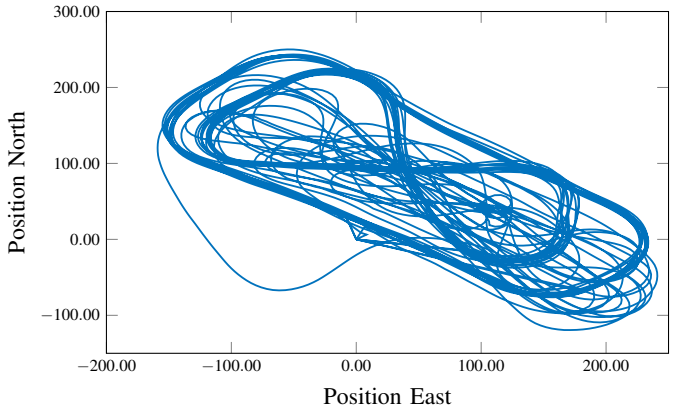

Fig. 5: Flight path X8 Flight 1

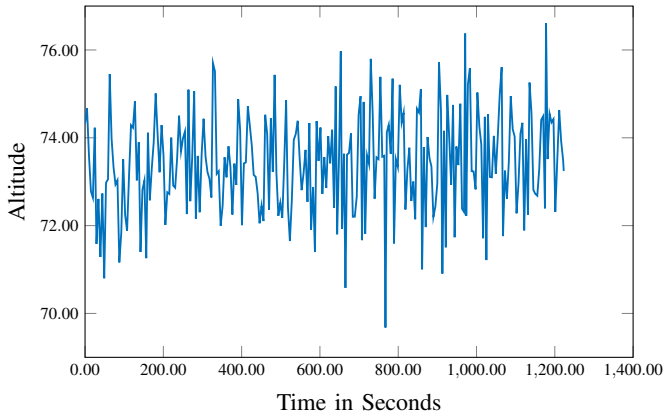

Fig. 6: Altitude X8 Flight 1

attack. The aircraft's path and altitude are shown in Figures 7 and 8.

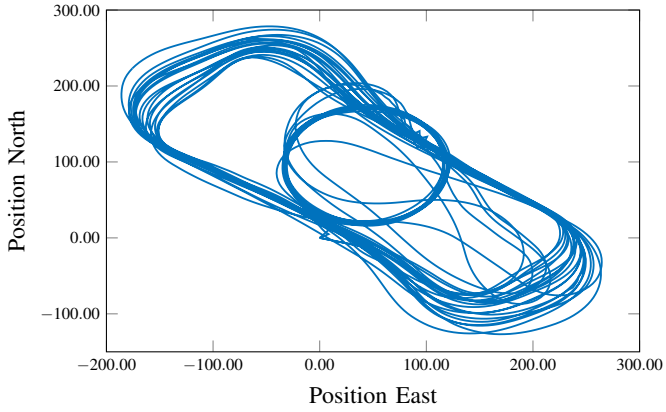

Fig. 7: Flight path X8 Flight 2

The wind velocity during both flights was $3m/s$ from a north-western direction, highest gust velocity was $4m/s$ indicating little turbulence. The flights were performed at an airfield outside Trondheim, Norway on an altitude close to sea level.

2) Wind Velocity Estimates: In this section resulting wind estimates in the frame are shown. The reference for these estimates is calculated from the Aeroprobe's measurements

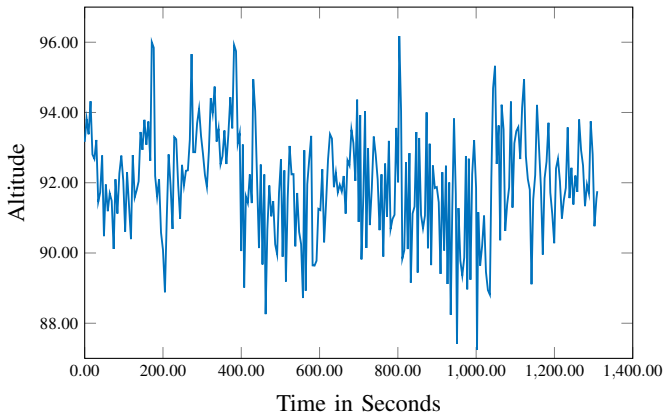

Fig. 8: Altitude X8 Flight 2

by first calculating the relative airspeed vector using (24) and afterwards the wind velocity vector with the wind triangle (1). As ground velocity measurements in inertial frame the RTK-GPS solutions were used and rotated into body frame using the solution of the observer described in section III-B. It should be noted that the quality of the wind estimates greatly depends on the accuracy of the velocity over ground and attitude estimates.

Figures 9 - 11 show the resulting wind velocity estimation errors for the first flight, and Figures 12 - 14 for the second. Estimates in longitudinal direction have a slight constant bias of about $1m/s$. This a result of the bias in the airspeed estimate shown in Section IV-A4, which due to a low angle of attack manifests itself in the longitudinal wind velocity estimate (cf. (25)). While the wind velocity estimate in vertical direction shows low estimation errors and fast convergence, the errors are higher in lateral direction. Since no model for the lateral direction is used, less information is available and the wind velocity can only be observed by changing the attitude of the aircraft. The estimation performance then mainly depends on how turbulent the prevailing wind conditions are and how often the roll or yaw angle changes occur. Root mean square errors can be found in Table IV and Table V.

Between the two flights the estimation results are quite similar and close to the accuracy limits of the reference ($1m/s$). Comparing the two sensor sets the performance of the autopilot data set is somewhat worse than the payload dataset. This is mainly due to the fact that in the payload dataset RTK-GNSS is used, while in the autopilot only standard GNSS is used resulting in a lower accuracy of the speed over ground estimates.

One could improve the performance in lateral direction by including a model of the lateral kinetics. However, doing this proves to be challenging since the lateral aerodynamics depend not only on the sideslip angle but also on ailerons and rudder movements as well as roll and yaw rates (see [5] equation (4.18)). These dependencies greatly increase the number of unknown coefficients and persistence of excitation conditions are generally hard to fulfill.

3) Coefficient Estimates: Figures 15 and 16 show the linear lift coefficient estimates for both flights. The convergence for these coefficients is quite slow. It is noticeable for the

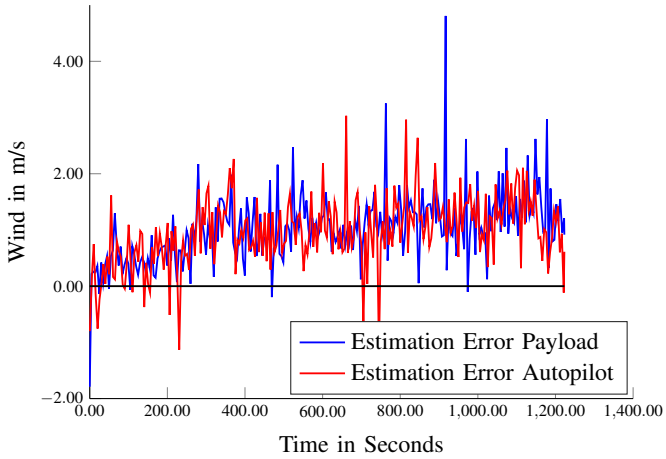

Fig. 9: Wind velocity estimation error in longitudinal direction, X8 Flight 1

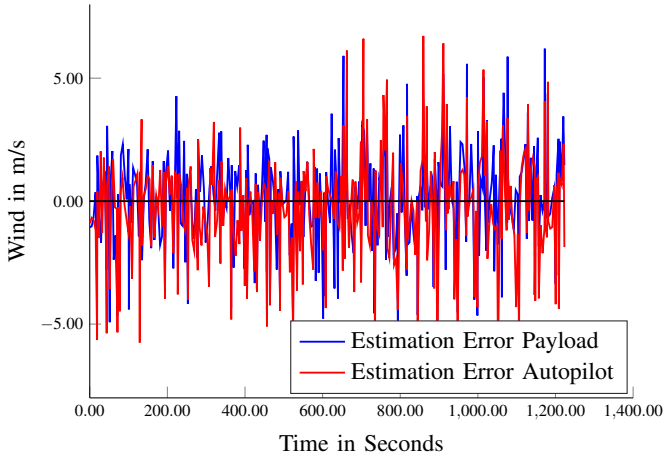

Fig. 10: Wind velocity estimation error in lateral direction, X8 Flight 1

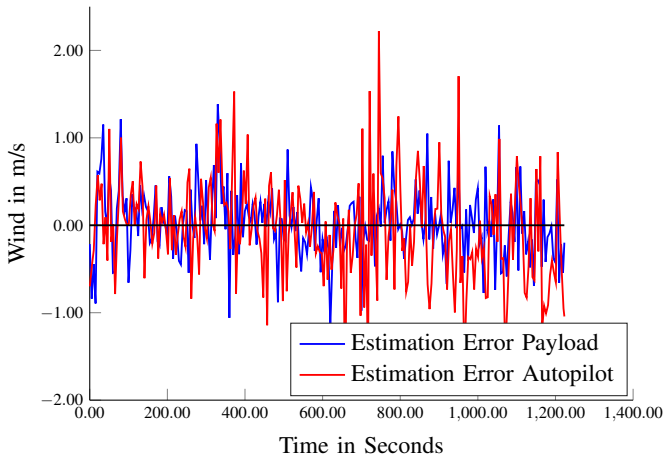

Fig. 11: Wind velocity estimation error in vertical direction, X8 Flight 1

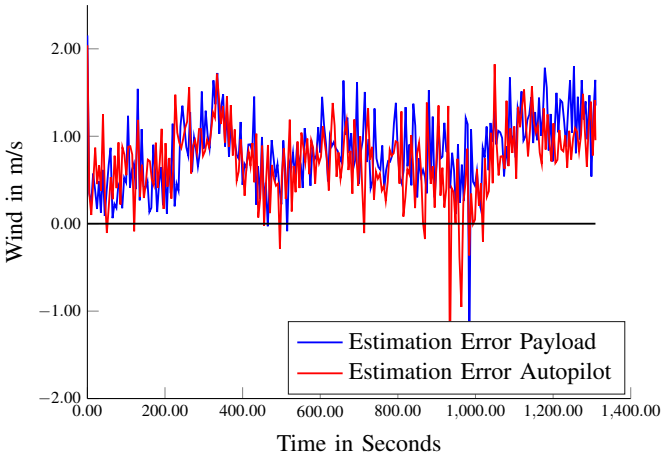

Fig. 12: Wind velocity estimation error in longitudinal direction, X8 Flight 2

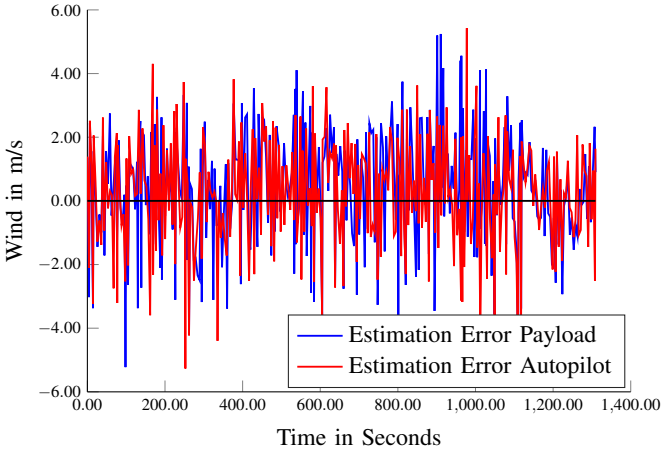

Fig. 13: Wind velocity estimation error in lateral direction, X8 Flight 2

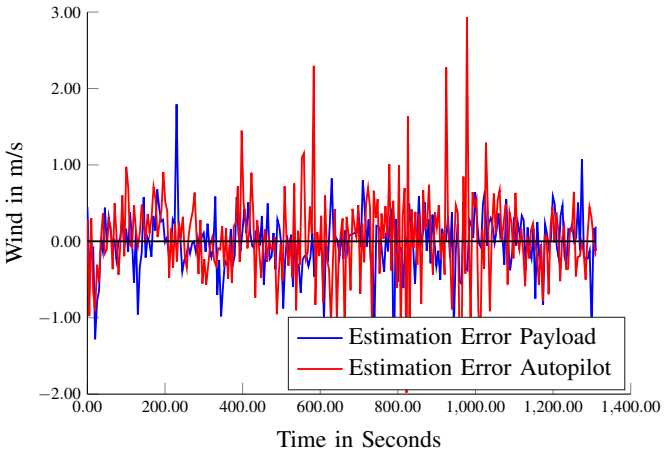

Fig. 14: Wind velocity estimation error in vertical direction, X8 Flight 2

second flight at 400s and 900s the coefficient converges faster. This is when the angle of attack is changed during the flight which results in stepwise convergence. In the first flight, where the airspeed is kept constant, convergence of the linear lift coefficient is not fully achieved by the end of the flight. This shows that airspeed changes which result in variations of the

angle of attack, enhance convergence of the estimator.

The lift coefficients converge to similar values in both flights but show differences in the convergence rate between the two different sensors sets. This might be due to different IMU alignment inside the aircraft as well as different noise characteristics. The slight changes over time can be explained by nonlinearities in the lift curves which are not captured by the model. These result from non rigid airfoils as well as air density changes in different altitudes and temperatures.

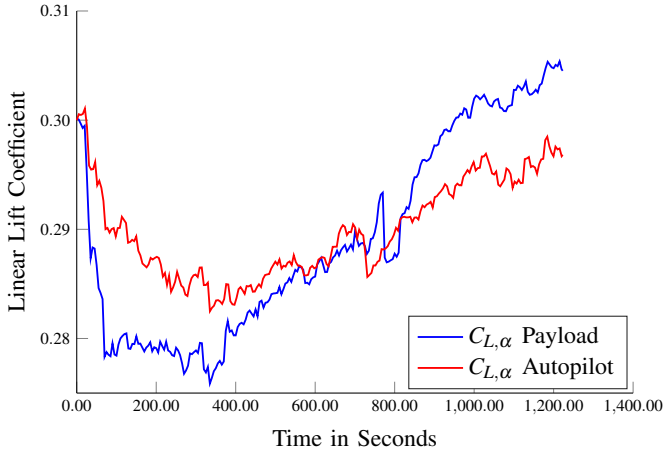

Fig. 15: Linear lift coefficient estimates X8, Flight 1

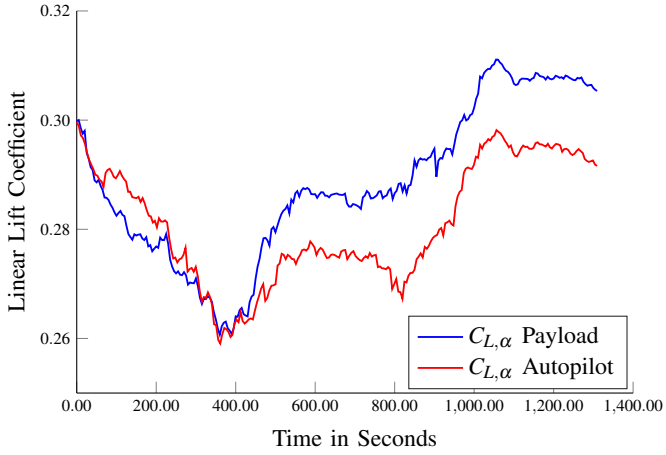

Fig. 16: Linear lift coefficient estimates X8, Flight 2

Estimates of the constant lift coefficients are shown in Figures 17 and 18. These estimates converge quickly to a steady value and only show slight variations over time which are mainly due to estimation errors in the linear lift coefficient estimates. Comparing the two flights the coefficient, estimates are similar. The slight differences could be explained as a result of the different maneuvers in the flights.

Figures 19 and 20 show estimates of the pitot-tube scaling factor. These estimates converge quickly but show some small fluctuations. These variations have different causes: The pitot-static system of the Pixhawk, used in both datasets, is of low quality and the used pitot-static tubes are short causing fluctuations in the flow from the wings and fuselage of the UAV. This causes noise and biases on the measurements. Additionally the measurements are not compensated for air

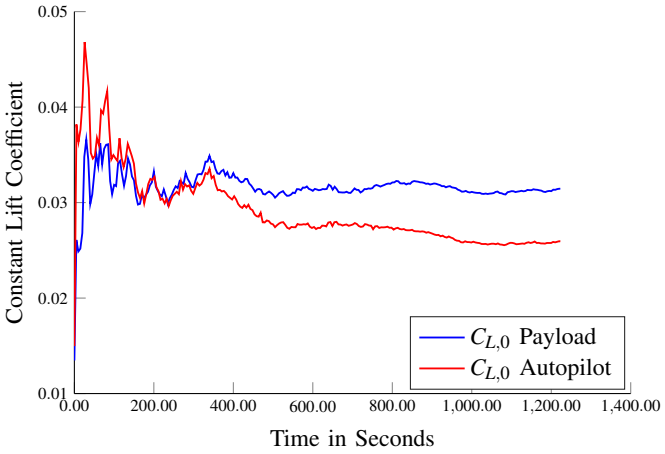

Fig. 17: Constant lift coefficient estimates X8, Flight 1

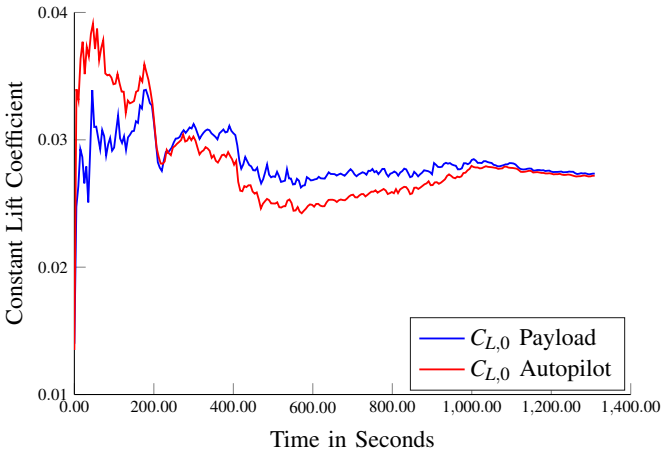

Fig. 18: Constant lift coefficient estimates X8, Flight 2

density changes, which are a result of temperature and altitude changes during the flight, and positioning errors of the probe. The air density varies by approximately 3% during the flight.

4) Angle of Attack and Airspeed Estimates: The angle of attack estimation errors are shown in Figures 21 and 22. Reference values for the angle of attack and the airspeed are provided by the measurements from the multi hole probe.

The estimation errors quickly converge for both sensor sets and flights. In general the estimation errors are larger for the autopilot dataset than for the sensor payload dataset. This is due to the higher quality of the sensors in the payload. RMSE values can be found in tables IV and V, showing generally good performance. The largest errors occur mainly during turns when the sideslip angle is nonzero. Additionally, errors are introduced due to non-rigid body of the Skywalker X8 which is mainly made out of Styrofoam.

The airspeed estimation errors, shown in Figures 23 and 24, have low noise characteristics and are generally close to the range of the accuracy of the reference measurement ($1m/s$), but show a negative bias. This behavior could be explained by the maneuvers flown in the test flights. In both flights the majority of the path consists of turns. Since the X8 flying wing is not equipped with a rudder it tends to sideslip in turns, violating the assumption made in Equation (25). Due to the

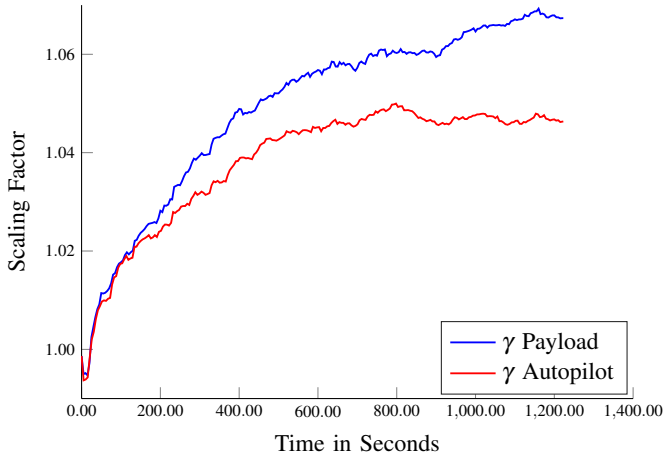

Fig. 19: Calibration factor estimates X8, Flight 1

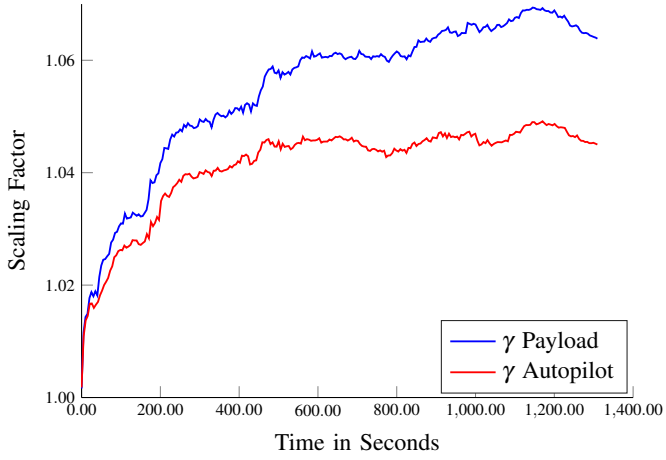

Fig. 20: Calibration factor estimates X8, Flight 2

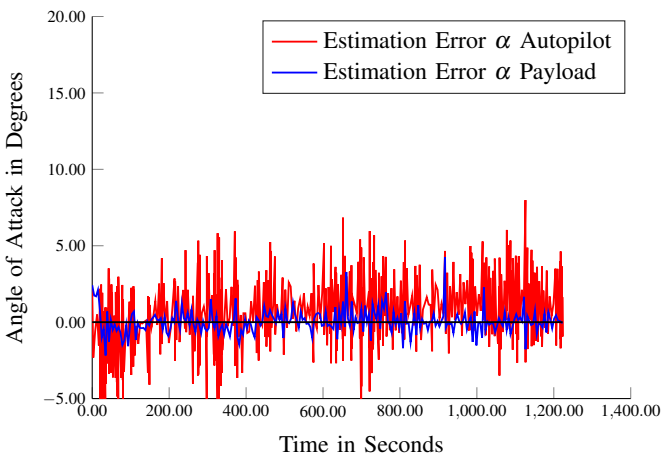

Fig. 21: Angle of Attack estimation error X8, Flight 1

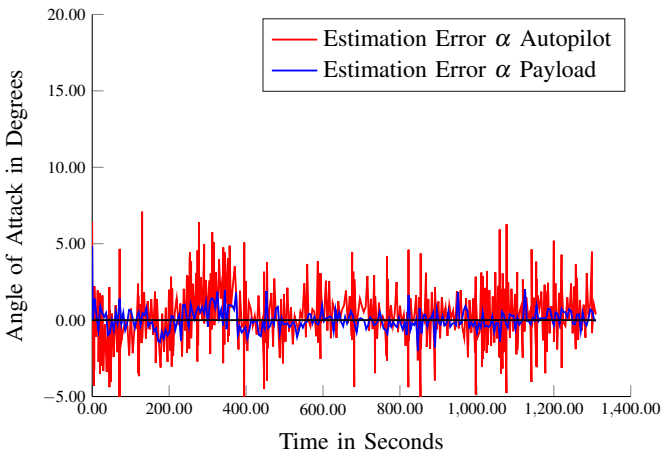

Fig. 22: Angle of Attack estimation error X8, Flight 2

TABLE IV: Results Skywalker X8 Flight 1

Variable	Payload Sensor	Autopilot Sensors	Ardupilot Estimate
RMSE $v_{w,x}$	$1.15m/s$	$1.13m/s$	$0.86m/s$
RMSE $v_{w,y}$	$1.72m/s$	$1.78m/s$	$1.98m/s$
RMSE $v_{w,z}$	$0.38m/s$	$0.56m/s$	$0.94m/s$
RMSE V_a	$1.15m/s$	$0.95m/s$	$0.69m/s$
RMSE α	$0.58°$	$1.62°$	$3.69°$
RMSE β	$6.48°$	$6.35°$	$9.4°$
$V_{w,G}$	$3.5m/s$	$3.5m/s$	$3.5m/s$
Runtime / Iteration	$0.05s$	$0.07s$	–

TABLE V: Results Skywalker X8 Flight 2

Variable	Payload Sensor	Autopilot Sensors	Ardupilot Estimate
RMSE $v_{w,x}$	$0.88m/s$	$0.80m/s$	$0.75m/s$
RMSE $v_{w,y}$	$1.56m/s$	$1.48m/s$	$1.85m/s$
RMSE $v_{w,z}$	$0.39m/s$	$0.47m/s$	$0.76m/s$
RMSE V_a	$0.86m/s$	$0.67m/s$	$0.67m/s$
RMSE α	$0.57°$	$1.36°$	$2.84°$
RMSE β	$6.13°$	$5.77°$	$8.07°$
$V_{w,G}$	$3m/s$	$3m/s$	$3m/s$
Runtime / Iteration	$0.05s$	$0.07s$	–

constant turning the approximation error $\boldsymbol{\nu}_{ki}$ has a non-zero mean, resulting in a biased airspeed estimate.

The autopilot dataset in the first test flight show some outliers which are caused by GNSS measurement outliers. These do not occur in the payload dataset showing the benefits of RTK-GNSS. Some of the remaining errors in maneuvers with fast changes of attitude, are caused by a discrete low pass filter within the autopilot given by:

$$y_k = 0.7y_{k-1} + 0.3u_k \tag{61}$$

where u is the input and y is the output of the filter.

B. Cruiser-Mini Flights

1) Flight Path: Two flights were undertaken with the Cruiser-Mini. One visual line of sight (VLOS) flight including race track and loitering patterns (Fig. 25) and one longer range beyond visual line of sight (BVLOS) flight (Fig. 27). The main difference between the two flights is that the second one includes long cruise phases ($8km$) where there are few attitude changes making it possible to assess the performance of the estimator in periods of low excitation. During the second flight the wind speed on ground level was $6m/s$ and gusts were frequent. Therefore the autopilot did not keep the commanded

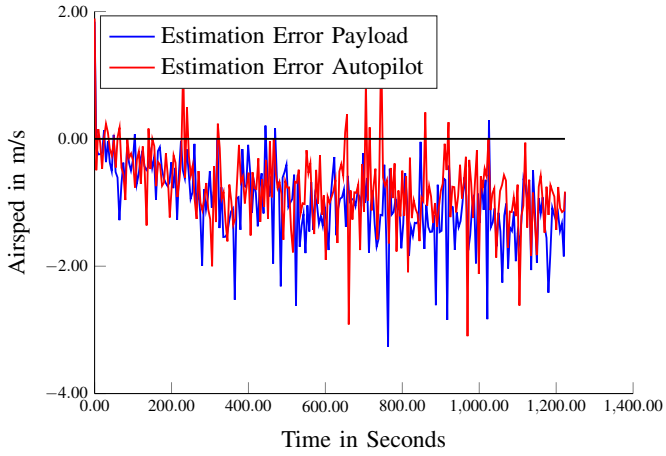

Fig. 23: Airspeed estimation error X8, Flight 1

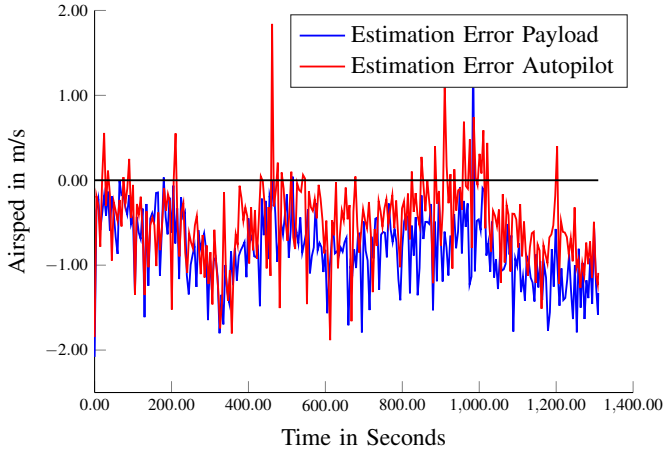

Fig. 24: Airspeed estimation error X8, Flight 2

altitude exactly but showed more up and down movements (see Fig. 28) compared to the first flight (Fig. 26). The wind velocity during the first flight was 3.5m/s from a north-western direction with little turbulence. The first flight was done at the same airfield as the X8 flights. The second flight was performed at a different location than the other flights on a take-off altitude of 700m above sea level, close to Røros, Norway.

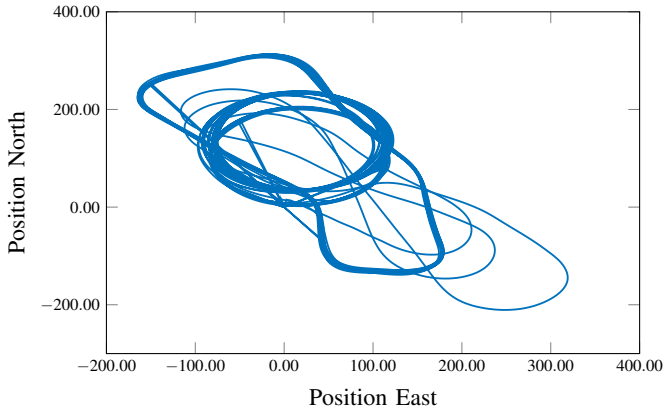

Fig. 25: Flight path Cruiser Mini Flight 1

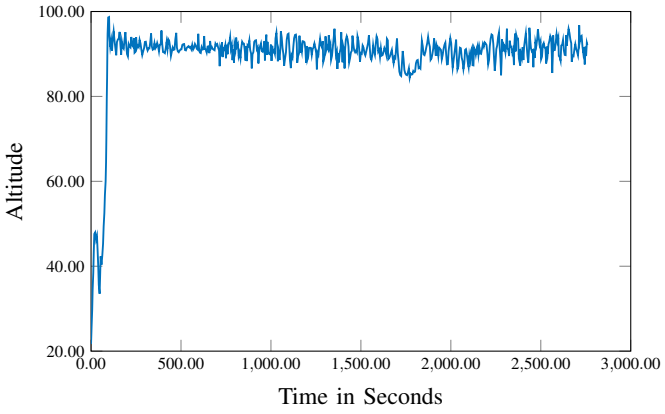

Fig. 26: Altitude Cruiser Mini Flight 1

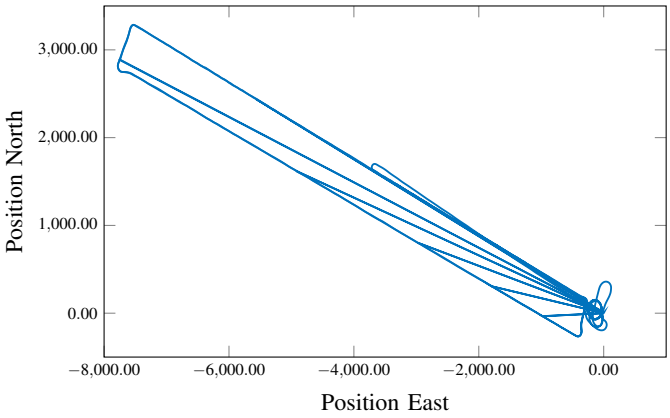

Fig. 27: Flight path Cruiser Mini Flight 2

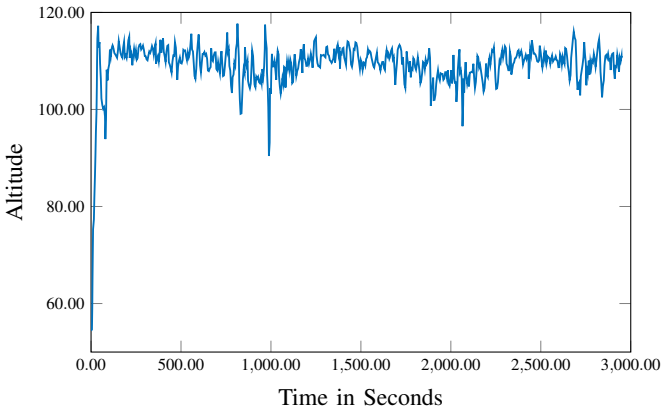

Fig. 28: Altitude Cruiser Mini Flight 2

2) Wind Velocity Estimates: Wind velocity estimates for the two flights are shown in Figures 29 - 31 and Figures 32 - 34. The estimation errors converge quickly after the start of the estimation. As for the X8 flights the estimation errors remain larger for the lateral direction than for the longitudinal and vertical directions. The wind estimates in longitudinal direction in the first flight are slightly biased but are still within the accuracy range of the reference measurement and lower than for the X8 flights. Errors are slightly higher for

the autopilot sensor set than for the payload sensor set which can be explained by the different sensor quality. RMSE values can be found in Table VI and Table VII. In general estimation errors are lower than for the flights with X8 which might be due to the more stiff airframe used.

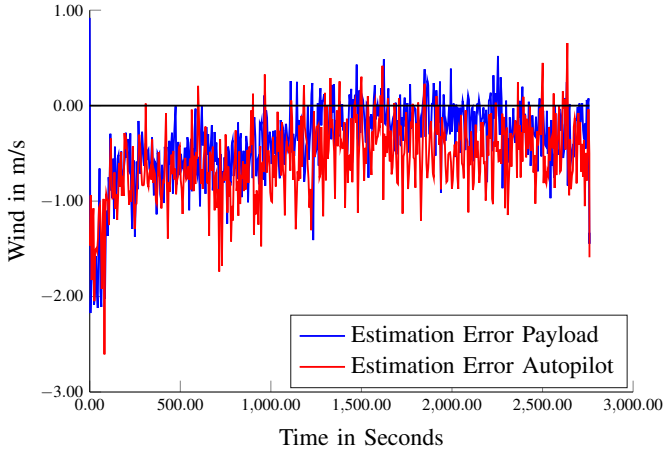

Fig. 29: Wind velocity estimation error in longitudinal direction, Cruiser-Mini Flight 1

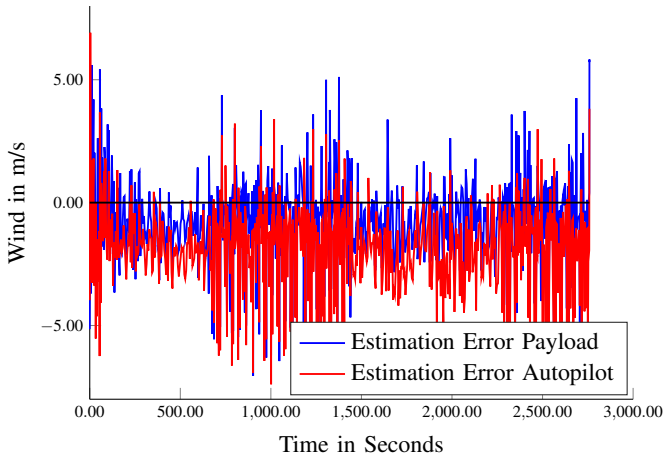

Fig. 30: Wind velocity estimation error in lateral direction, Cruiser-Mini Flight 1

3) Coefficient Estimates: The linear lift coefficient estimates for the Cruiser-Mini flights are shown in Figures 35 and 36. In the first flight convergence of the coefficient estimates is slow. As for the second X8 flight we see stepwise convergence during changes of the angle of attack e.g. at $t = 2000s$ in Flight 1. In the second flight, the angle of attack changes more frequently and the linear lift coefficients converges more gradually during the flight due to more gusts and turbulent winds. The estimated coefficient values are similar for both data sets but vary slightly between the two flights which might be due to different values of the estimated air density $\hat{\rho}$ within the autopilot. The results are indicating that the estimation result is independent of the flown maneuver as long as sufficient excitation is achieved.

Estimates of the constant lift coefficients are shown in Figures 37 and 38. In both flights the estimate converges

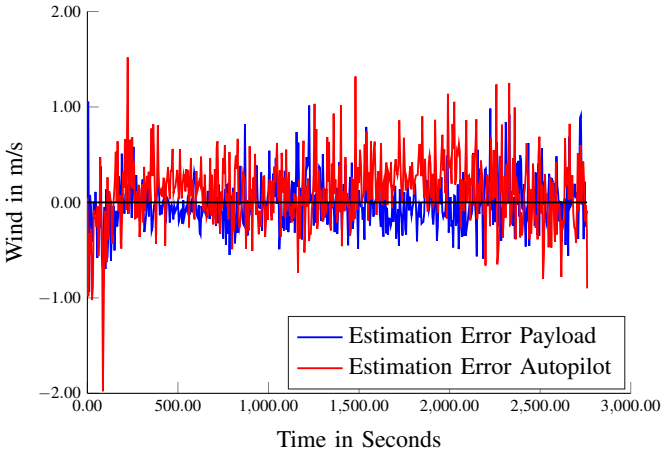

Fig. 31: Wind velocity estimation error in vertical direction, Cruiser-Mini Flight 1

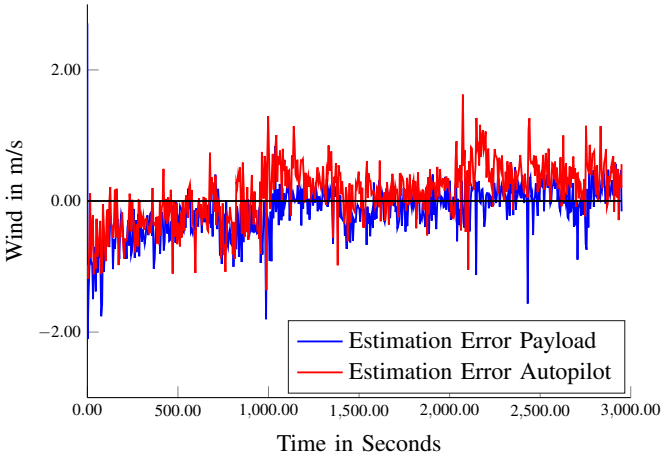

Fig. 32: Wind velocity estimation error in longitudinal direction, Cruiser-Mini Flight 2

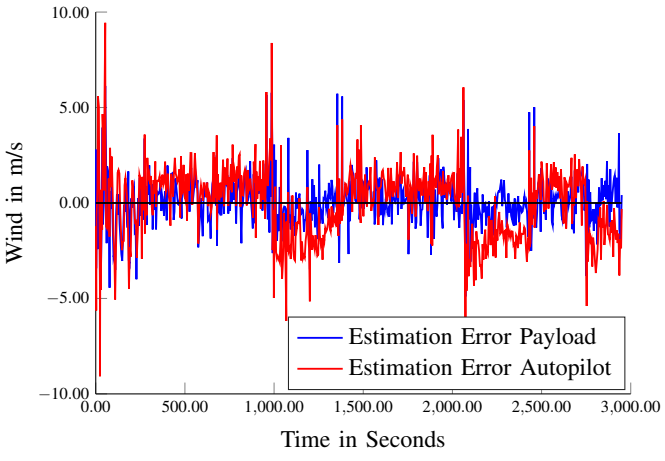

Fig. 33: Wind velocity estimation error in lateral direction, Cruiser-Mini Flight 2

quicker than the linear lift coefficient and only varies slightly over time. The estimates from both sensor sets converge to similar values.

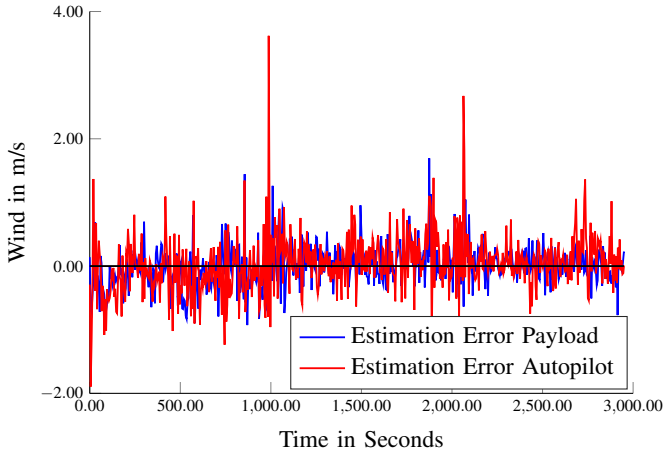

Fig. 34: Wind velocity estimation error in vertical direction, Cruiser-Mini Flight 2

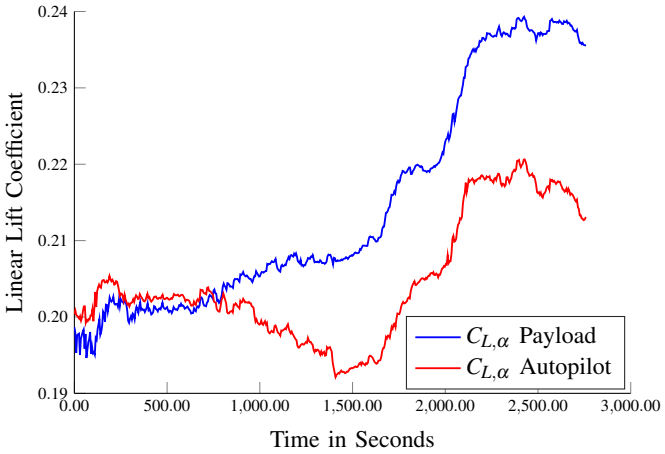

Fig. 35: Linear lift coefficient estimates Cruiser-Mini, Flight 1

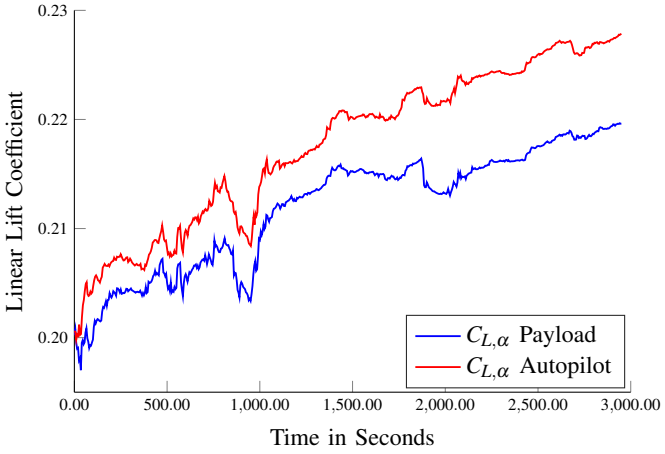

Fig. 36: Linear lift coefficient estimates Cruiser-Mini, Flight 2

The pitot tube calibration factor estimates are shown in Figures 39 and 40. For the first flight the scaling factor estimate converges quickly in the beginning of the flight during the loitering maneuver and afterwards more gradually. Small variations can be noted which might either be due to air density variations or uncertainties in the estimation. In the second flight the calibration factor converges quickly during the initial loitering. Afterwards due to the few heading changes, the turbulent wind and the frequent altitude changes there is more uncertainty in the calibration factor estimate, which leads to slightly varying estimates.

4) Angle of Attack and Airspeed Estimates: The angle of attack estimation errors of both flights (Fig. 41 and 42) converge quickly and remain low during both flights. The initial biases are due to the not converged coefficient estimates. It is noticeable in the second flight that the initial errors are reduced after excitations around $t = 1000s$, afterwards the angle of attack estimation error remains low also during regions of low excitation. As for the Skywalker X8 flights the estimation error is lower for the payload data set than for the autopilot dataset, due to the higher sensor quality. RMSE values can be found in tables VI and VII. They show that the accuracy of the angle of attack estimation is within the accuracy of the reference used (1°).

The airspeed estimation errors for the cruiser-mini flights are shown in Figure 43 and 44. In the first flight, where a circular pattern is flown, there is an initial bias on the airspeed estimate which is slowly reduced over time. The bias is somewhat larger than in the second test flight, but smaller than for the X8 flights. This is due to the fact that the Cruiser Mini is equipped with a rudder, which reduces sideslip in turns, resulting in a lower bias on the airspeed estimate. In flights with long straight line segments, like the second test flight, the bias vanishes. For the Payload dataset the estimation error remains below the stated accuracy of the reference measurement ($1m/s$), but for the autopilot dataset is higher in some instances. This might be due to misalignment of either the reference multi-hole probe or the pitot-static tube or this could be induced by errors in the IMU and GNSS velocity measurements of the autopilot.

Additionally, sideslip angle estimates are available, however as discussed before there accuracy is limited due to the lack of a lateral kinetic model. For completeness RMSE values are included in Tables VI and VII.

TABLE VI: Results Cruiser-Mini Flight 1

Variable	**Payload Sensor**	**Autopilot Sensors**	**Ardupilot Estimate**
RMSE $v_{w,x}$	$0.54m/s$	$0.71m/s$	$0.9m/s$
RMSE $v_{w,y}$	$1.57m/s$	$2.27m/s$	$2.02m/s$
RMSE $v_{w,z}$	$0.26m/s$	$0.36m/s$	$0.57m/s$
RMSE V_a	$0.56m/s$	$0.85m/s$	$1.51m/s$
RMSE α	0.49°	0.92°	2.06°
RMSE β	5.90°	5.14°	5.14°
$V_{w,G}$	$3.5m/s$	$3.5m/s$	$3.5m/s$
Runtime / Iteration	$0.05s$	$0.06s$	–

C. Observability

Observability of the kinematic subsystem using only the second measurements equation in (27) has been studied in

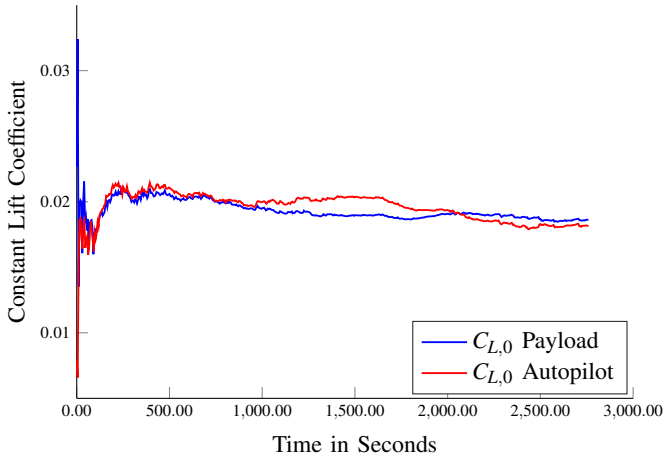

Fig. 37: Constant lift coefficient estimates Cruiser-Mini, Flight 1

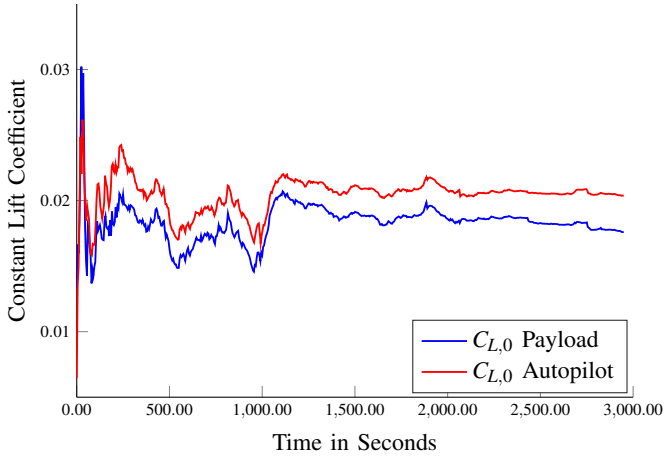

Fig. 38: Constant lift coefficient estimates Cruiser-Mini, Flight 2

TABLE VII: Results Cruiser-Mini Flight 2

Variable	Payload Sensor	Autopilot Sensors	Ardupilot Estimate
RMSE $v_{w,x}$	$0.39m/s$	$0.45m/s$	$1.59m/s$
RMSE $v_{w,y}$	$1.25m/s$	$1.82m/s$	$2.46m/s$
RMSE $v_{w,z}$	$0.33m/s$	$0.43m/s$	$0.84m/s$
RMSE V_a	$0.40m/s$	$0.44m/s$	$0.94m/s$
RMSE α	$0.69°$	$1.06°$	$2.59°$
RMSE β	$4.42°$	$4.37°$	$6.97°$
$V_{w,G}$	$6m/s$	$6m/s$	$6m/s$
Runtime / Iteration	$0.04s$	$0.05s$	–

[18]. Since in this case the system is linear time varying the observability grammian can be calculated for a time window and analytical observability conditions can be obtained. The findings in [18] were that the linear time varying system is observable if sufficient excitation in the rotation matrix $\boldsymbol{R}_n^b$ occurs.

In this paper we have extended this model using the aerodynamic model (7) and the model for the turbulent wind velocitiy (20). The resulting system is nonlinear and non-

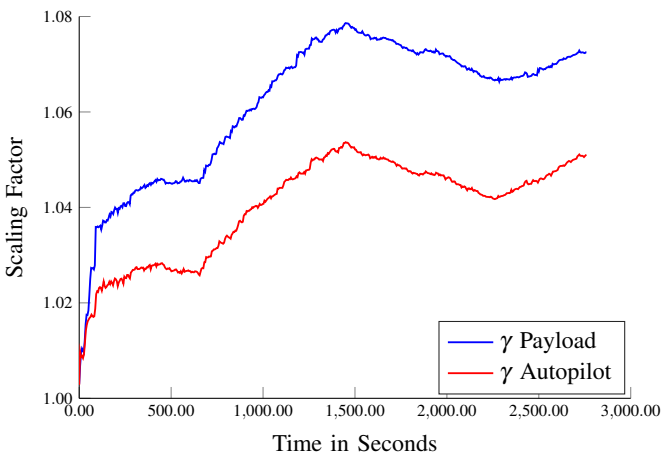

Fig. 39: Calibration factor estimates Cruiser-Mini, Flight 1

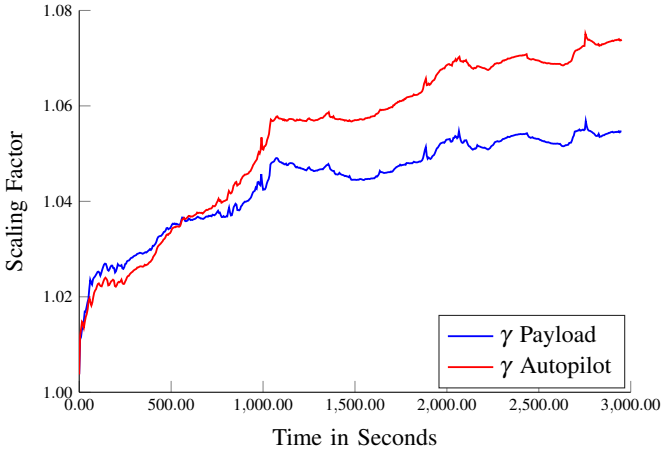

Fig. 40: Calibration factor estimates Cruiser-Mini, Flight 2

affine. Observability conditions for discretized nonlinear systems have been studied i.e. in [27] and [2]. However since in this case the observability depends on the input trajectories it is challenging to achieve analytical observability conditions which are of practical use. In [44] and [43] we have shown through simulations that the system is observable if attitude changes occur in a regarded time frame.

To asses observability we use the largest eigenvalue of the estimated error covariance matrix $\hat{\boldsymbol{P}}_{k-L}$ normalized by the initial error covariance matrix $\hat{\boldsymbol{P}}_0$. High values indicate that one of the states or one of the parameters is not observable with the given input trajectories. An example for the results of this observability measure can be seen in Figure 45. The observability measure quickly converges to a region around 10^5. The maximum eigenvalue does remains in an interval around 10^5 indicating that the eigenvalues of the covariance matrix are bounded. This shows that the error covariance matrix is not singular and that the system is locally observable during both flights.

D. *Influence of window length*

To analyze the effect of the window length L on the estimation performance, we vary L between 1 and 10 and use the RMSE of the longitudinal and vertical wind velocity

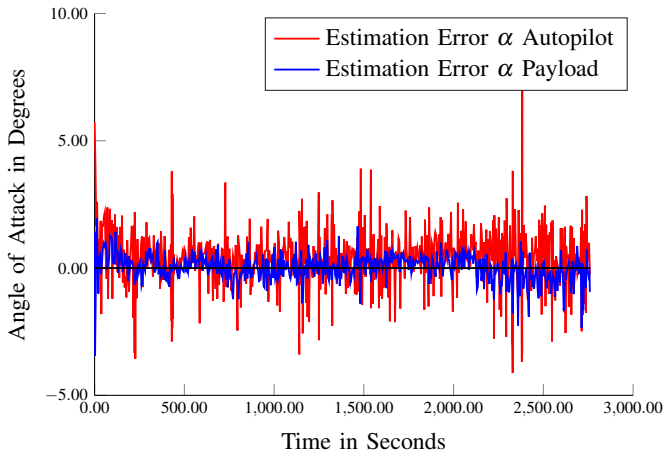

Fig. 41: AoA estimation error Cruiser-Mini, Flight 1

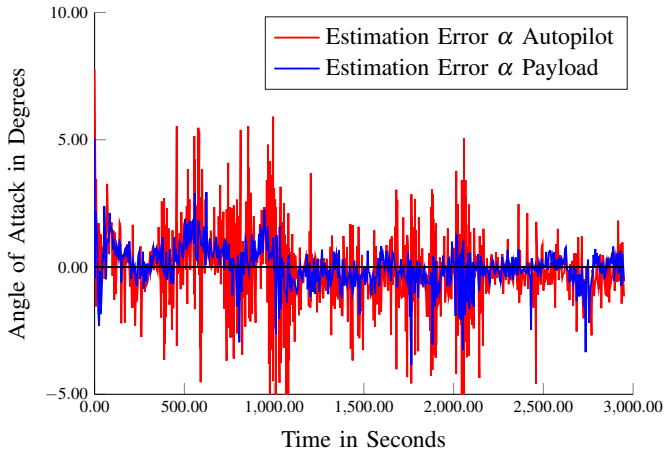

Fig. 42: AoA estimation error Cruiser-Mini, Flight 2

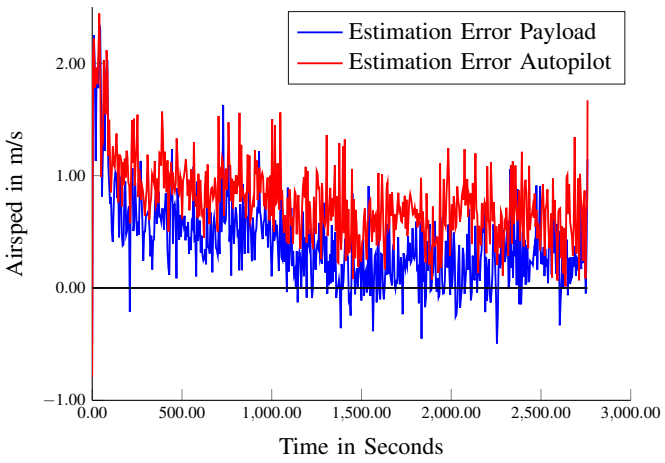

Fig. 43: Airspeed estimation error Cruiser-Mini, Flight 1

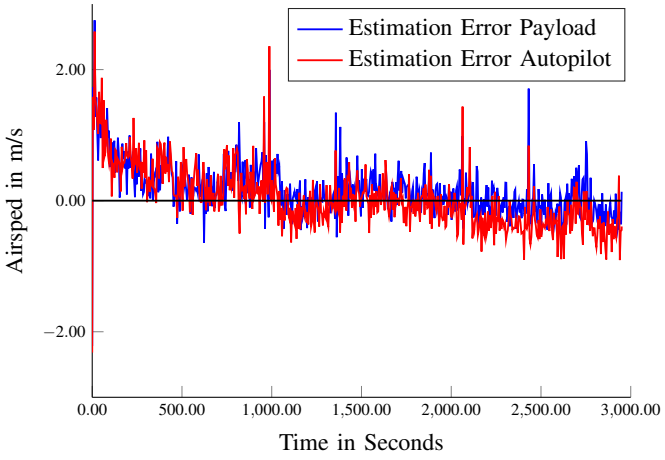

Fig. 44: Airspeed estimation error Cruiser-Mini, Flight 2

estimates as an performance indicator. Note that in the case of $L = 1$ the MHE is equivalent to an Unscented Kalman Filter as long as none of the inequality constraints defined in (60) are active.

Figure 46 shows the results of this analysis for the flight data obtained from *Cruiser-Mini, Flight 1* using the parameters given in Table III. The estimation error decreases with increasing window length for both the payload and the autopilot dataset. For the payload dataset there are no substantial improvements for window lengths larger than $L = 4$, for the autopilot dataset the estimation error continues to decrease until $L = 10$. It should be noted that the estimations are the largest for the case $L = 1$, clearly outlining the improvement in estimation performance of the MHE in comparison to an UKF.

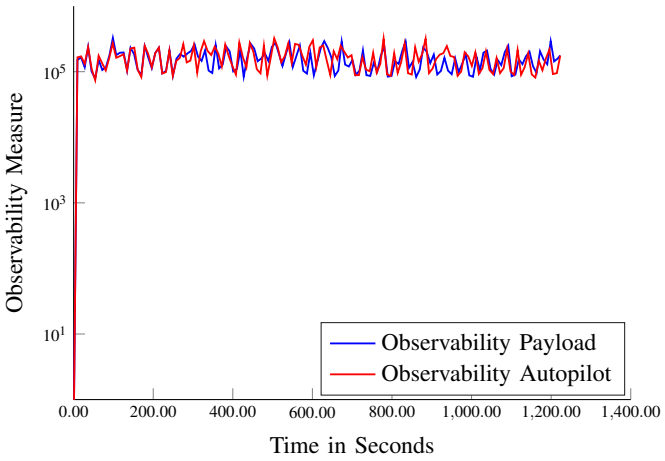

Fig. 45: Observability during X8 Fight 1

E. Runtime

Although real time implementation was not a focus of this paper and the code was not optimized for efficiency, the runtime is analyzed in order to assess the feasibility of a future on-line implementation of the method on a low power on-board UAV computer

Tables IV - VII show that the average runtime of the algorithm is in the range of $0.05s - 0.07s$, which is significantly below the sampling interval of the system of $0.2s$ for all test flights. This shows that the estimator is already able to run in real time on the used hardware (Intel Core i7). It is therefore expected, that a modified and optimized version of the code is able to run on-line on-board an UAV.s

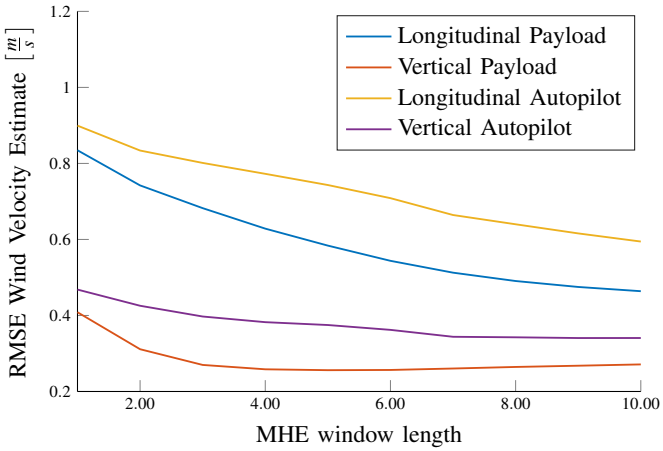

Fig. 46: Wind velocity estimation performance for varying window length

F. Comparison to the Ardupilot's estimator

The software on board that is used on board the Pixhawk autopilot, Ardupilot, provides estimates of the wind velocities in north and east direction as well as estimates of the angle of attack, sideslip angle and airspeed. The estimator is described in [29]. It relies on a kinematic wind model assuming slowly changing wind velocities. In the following we will compare the performance achieved by this commonly used estimator to the MHE based approach presented in this work. Both estimators use the same sensor set, provided by the Pixhawk autopilot.

Tables IV - VII show the RMSE values for the different estimated variables. The wind velocities estimates provided by Ardupilot are reasonably accurate in longitudinal direction if the pitot-static tube is well calibrated, which was not the case in Cruiser-Mini flight 2. The MHE is able to compensate for that, since it estimates the pitot-tube scaling factor. The wind velocity estimation errors in lateral and vertical direction are less accurate than the estimates of the MHE. Note that the performance of the wind velocities decrease with higher wind velocity on the ground due to the higher gust amplitudes, violating the assumption of slowly varying wind velocities. This illustrates the benefits of the wind turbulence model used in this work.

The angle of attack RMSE values exceed 2° in all test cases and are significantly higher than for the MHE. The main reason for this is the use of the aerodynamic model which utilizes the information from the z-accelerometer. The angle of sideslip RMSE values are also increased in comparison to the MHE results.

A comparison of runtimes was not conducted, since the two methods run on different hardware platforms.

V. Conclusions

This paper presents a method to estimate wind velocities, air data parameters and lift coefficients using sensor data from an IMU, a GNSS system and pitot-static tube together with kinematic, aerodynamic and wind models. These sensors are typically part of a standard autopilot sensor suite, avoiding the need for additional sensors solely designed to measure the desired variables. The moving horizon estimator was applied using direct collocation for the construction of the nonlinear program and an unscented Kalman filter for arrival cost approximation. The method was extensively tested using experimental flight data from two different platforms and two different sensor sets. Data from four different test flights was presented covering a variety of different maneuvers including long cruiser sections with low excitation.

The estimator performed very well at estimating angle of attack, wind velocities and aerodynamic coefficients, outperforming the estimator used by the autopilot's software. The wind velocity estimation errors were within the margin of error of the reference used. An estimate of the sideslip angle is also available, however due to the lack of a lateral wind speed measurement the performance is limited.

VI. Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-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 our test pilots Lars Semb and Pål Kvaløy for conducting the flight tests. Also we would like to thank Torleiv Bryne for helping with the implementation of the attitude and translational motion observers.

References

[1] Ira H. Abbott and Albert E. Von Doenhoff. *Theory of wing sections*, volume 249. Dover Publications, 1949.

[2] Francesca Albertini and Domenico D'Alessandro. Observability and forward-backward observability of discrete-time nonlinear systems. *Math. Control. Signals, Syst.*, 15(4):275–290, 2002.

[3] Sigurd M Albrektsen and Tor A Johansen. SyncBoard - A high accuracy sensor timing board for UAV payloads. In *Int. Conf. Unmanned Aircr. Syst.*, pages 1706–1715, Miami, FL, USA, 2017.

[4] Joel Andersson. *A General Purpose Software Framework for Dynamic Optimization*. Phd thesis, KU Leuven, 2013.

[5] Randal W. Beard and Timothy W. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012.

[6] Lorenz T. Biegler. Solution of Dynamic Optimization Problems and Orthogonal Collocation. Technical Report 3, Carnegie-Mellon University The Robotics Institute, 1984.

[7] Lorenz T. Biegler, Arturo M. Cervantes, and Andreas Wächter. Advances in simultaneous strategies for dynamic process optimization. *Chem. Eng. Sci.*, 57(4):575–593, 2002.

[8] Kasper T. Borup, Thor I. Fossen, and Tor A. Johansen. A Nonlinear Model-Based Wind Velocity Observer for Unmanned Aerial Vehicles. In *IFAC-PapersOnLine*, volume 49, pages 276–283, Monterrey, CA, USA, 2016.

[9] Torleiv Håland Bryne. *Nonlinear Observer Design for Aided Inertial Navigation of Ships*. PhD thesis, Norwegian University of Science and Technology, 2017.

[10] Torleiv Håland Bryne, Jakob M Hansen, Robert H Rogne, and Thor I Fossen. Nonlinear Observers for Integrated INS/GNSS Naviagation - Implementation Aspects. *IEEE Control Syst. Mag.*, 37(3):59–86, 2017.

[11] Am Cho, Jihoon Kim, Sanghyo Lee, and Changdon Kee. Wind estimation and airspeed calibration using a UAV with a single-antenna GPS receiver and pitot tube. *IEEE Trans. Aerosp. Electron. Syst.*, 47(1):109–117, 2011.

[12] Jean-Philippe Condomines, Murat Bronz, and Gautier Hattenberger. Experimental Wind Field Estimation and Aircraft Identification. In *IMAV 2015 Int. Micro Air Veh. Conf. Flight Compet.*, 2015.

[13] Moritz Diehl, Hans Joachim Ferreau, and Niels Haverbeke. Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation. In Lalo Magni, Davide Martino Raimondo, and Frank Allgöwer, editors, *Nonlinear Model Predict. Control Towar. New Challenging Appl.*, pages 391–417. Springer, Berlin, Heidelberg, 2009.

[14] William Gracy. Measurement of Aircraft Speed and Altitude. Technical report, NASA, 1980.

[15] Håvard Fjær Grip, Thor I. Fossen, Tor A. Johansen, and Ali Saberi. Attitude estimation based on time-varying reference vectors with biased gyro and vector measurements. *IFAC Proc. Vol.*, 18(PART 1):8497–8502, 2011.

[16] 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 *Am. Control Conf.*, pages 272–279, jun 2013.

[17] Richard Hann, Andreas Wenz, Kristoffer Gryte, and Tor Arne Johansen. Impact of Atmospheric Icing on UAV Aerodynamic Performance. In *Int. Work. Res. Educ. Dev. Unmanned Aer. Syst.*, Linköping, Sweden, 2017.

[18] Tor Arne 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 *Int. Conf. Unmanned Aircr. Syst.*, Denver, 2015.

[19] Derek B Kingston and Randal W. Beard. Real-time Attitude and Position Estimation for Small UAVs using Low-cost Sensors. *Proc. AIAA Unmanned Unltd. Tech. Conf. Work. Exhib.*, pages 2004–6488, 2004.

[20] Peter Kühl, Moritz Diehl, Tom Kraus, Johannes P. Schlöder, and Hans Georg Bock. A real-time algorithm for moving horizon state and parameter estimation. *Comput. Chem. Eng.*, 35(1):71–83, 2011.

[21] Makoto Kumon, Ikuro Mizumoto, Zenta Iwai, and Masanobu Nagata. Wind Estimation by Unmanned Air Vehicle with Delta Wing. In *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 1896–1901. IEEE, 2005.

[22] Jack W. Langelaan. Long Distance / Duration Trajectory Optimization for Small UAV's. In *AIAA Guid. Navig. Control Conf.*, number August, pages 1–14, Hilton Head, South Carolina, 2007.

[23] Jack W Langelaan, Nicholas Alley, and James Neidhoefer. Wind Field Estimation for Small Unmanned Aerial Vehicles. *J. Guid. Control Dyn.*, 34:1016–1030, 2011.

[24] Hao Long and Shujie Song. Method of Estimating Angle-of-Attack and Sideslip Angel Based on Data Fusion. In *2009 Second Int. Conf. Intell. Comput. Technol. Autom.*, volume 1, pages 641–644. IEEE, 2009.

[25] Rodrigo López-Negrete, Sachin C. Patwardhan, and Lorenz T. Biegler. Constrained particle filter approach to approximate the arrival cost in Moving Horizon Estimation. *J. Process Control*, 21(6):909–919, 2011.

[26] Robert Mahony, Tarek Hamel, Pascal Morin, and Ezio Malis. Nonlinear complementary filters on the special linear group. *Int. J. Control*, 85(10):1557–1573, 2012.

[27] Henk Nijmeijer and Arjan Van Der Schaft. *Nonlinear Dynamical Control Systems*, volume 464. Springer, 1990.

[28] U.S. Department of Defence. MIL-STD-1797A: Flying Qualities of Piloted Aircraft. Technical report, U.S. Department of Defence, 2004.

[29] William Premerlani. IMU WInd Estimation. 2009.

[30] Cheryl C. Qu and Juergen Hahn. Computation of arrival cost for moving horizon estimation via unscented Kalman filtering. *J. Process Control*, 19(2):358–363, 2009.

[31] Christopher V. Rao, James B. Rawlings, and David Q. Mayne. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE Trans. Automat. Contr.*, 48(2):246–258, 2003.

[32] James B. Rawlings and Bhavik R. Bakshi. Particle filtering and moving horizon estimation. *Comput. Chem. Eng.*, 30(10-12):1529–1541, 2006.

[33] Matthew B. Rhudy, Trenton Larrabee, Haiyang Chao, Yu Gu, and Marcello Napolitano. UAV Attitude, Heading, and Wind Estimation Using GPS/INS and an Air Data System. In *AIAA Guid. Navig. Control Conf.*, pages 1–11, Boston, 2013.

[34] Douglas Robertson, Jay Lee, and James Rawlings. A moving horizon-based approach for least-squares estimation. *AIChE J.*, 42(8):2209–2224, 1996.

[35] Andres Rodriguez, Evan Andersen, Justin Bradley, and Clark Taylor. Wind Estimation Using an Optical Flow Sensor on a Miniature Air Vehicle. In *AIAA Guid. Navig. Control Conf. Exhib.*, volume 6614, pages 1–8, Reston, Virigina, aug 2007. AIAA.

[36] Leopoldo Rodriguez Salazar, Jose A. Cobano, and Anibal Ollero. Small UAS-based wind feature identification system Part 1: Integration and validation. *Sensors*, 17(1):1–28, 2017.

[37] Mohammad Shaqura and Christian Claudel. A hybrid system approach to airspeed, angle of attack and sideslip estimation in Unmanned Aerial Vehicles. In *Int. Conf. Unmanned Aircr. Syst.*, pages 723–732, 2015.

[38] Kim Lynge Sørensen and Tor Arne Johansen. Flight Test Results for Autonomous Icing Protection Solution for Small Unmanned Aircraft. In *Int. Conf. Unmanned Aircr. Syst. (ICUAS 2017)*, Miami, 2017.

[39] Tomoji Takasu and Akio Yasuda. Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB. *Int. Symp. GPS/GNSS*, pages 4–6, 2009.

[40] Pengzhi Tian and Haiyang Chao. UAV Flight Test Evaluation of Fusion Algorithms for Angle of Attack and Sideslip Angle. In *AIAA Guid. Navig. Control Conf.*, San Diego, CA, 2016. AIAA.

[41] T. H. Tsang, D. M. Himmelblau, and T. F. Edgar. Optimal control via collocation and non-linear programming. *Int. J. Control*, 21(5):763–768, may 1975.

[42] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.*, 106(1):25–57, 2006.

[43] Andreas Wenz and Tor Arne Johansen. Estimation of Wind Velocities and Aerodynamic Coefficients for UAVs using standard Autopilot Sensors and a Moving Horizon Estimator. In *Int. Conf. Unmanned Aircr. Syst.*, pages 1267–1276, 2017.

[44] 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 *Int. Conf. Unmanned Aircr. Syst.*, pages 624–632, Arlington, VA, 2016.