2017 International Conference on
Unmanned Aircraft Systems (ICUAS)
June 13-16, 2017, Miami, FL, USA

Estimation of Wind Velocities and Aerodynamic Coefficients for UAVs using standard Autopilot Sensors and a Moving Horizon Estimator

Andreas Wenz *, Tor Arne Johansen*
*Centre for Autonomous Marine Operations and Systems
Department of Engineering Cybernetics
Norwegian University of Science and Technology, Trondheim, Norway

***Abstract*— While operating any aircraft it is vital to know its current flight state. Some of the most important variables to assess the flight state are the airspeed, the angle of attack and the sideslip angle. Larger aircraft are equipped with sensors specifically designed to measure these variables. However on small unmanned aerial vehicles (UAVs) much stricter restrictions on size, weight and cost prohibit the use of such sensors. Therefore we propose a method to estimate the airflow variables utilizing only sensors that are part of a standard UAV autopilot. This includes an inertial measurement unit (IMU), a global navigation satellite system (GNSS) receiver and a pitot-static tube. These measurement together with kinematic and aerodynamic models will be fused within an estimator to estimate steady and turbulent wind velocities as well as aerodynamic coefficients. With these estimates it is possible to calculate the angle of attack, the sideslip angle and the airspeed. A main challenge is to distinguish between changes in the aerodynamic coefficients and changes in wind velocity, since pitot-static tube measurements of the relative airspeed are only available in one direction at a time and hence the system is not always observable. Therefore attitude changes have to be undertaken to achieve persistence of excitation. In this paper a Moving Horizon Estimator (MHE) is used for estimation. Simulation results show overall good estimation results and significant improvements compared to a previous Extended Kalman Filter approach. Root mean square errors (RMSE) are 0.25° for the angle of attack, $0.08m/s$ for the airspeed and 1.06° for the side slip estimates.**

I. Introduction

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. Larger aircraft are usually equipped with sensors dedicated to deliver reliable measurements of these variables. Those sensor systems usually include vanes or multi-hole pitot-static tubes. Unfortunately, on small fixed-wing unmanned aerial vehicles (UAVs), these sensor systems are difficult to install due to much stricter weight and size limitations or their costs often prove to be prohibitive. Previous work (i.e. [24], [3], [6]) has therefore focused on model-based approaches utilizing aircraft models which fully model the aerodynamic and kinematic behavior of the UAV. However these models require the knowledge of a large parameter set which in many cases is difficult or costly to obtain, since this often requires the availability of wind tunnel data. Some of these parameters might also change from mission to mission due to different payloads or individual differences between supposedly identical UAVs. Aerodynamic coefficients can even change during a mission due to structural effects like in-flight icing. Due to these circumstances it is desirable to develop estimation concepts that do not rely on the availability of a large parameter set 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, allowing the computation of the airflow variables.

Recently several methods have been proposed to estimate the airflow variables and aerodynamic coefficients. One popular methods is the Extended Kalman Filter (EKF), which has been used in [7] , [14], [16], and the Unscented Kalman Filter (UKF) which has been applied to the problem in [21], [8].

Tian et.al [25] compare the use of an EKF, Output Error Minimization and a Complimentary Filter to improve measurements of the airflow variables obtained from a multi-hole probe. [6] uses a detailed aircraft model and a nonlinear observer for wind estimation. In [15] kinematic vehicle models are used together with IMU, GNSS and multi-directional airspeed sensors to estimate the wind field. [23] propose a method using optical flow measurements. A hybrid system approach using Bayesian estimation is presented in [24], and achieving promising results. 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 previous work [28] where the above mentioned sensor suite was used together with aerodynamic and kinematic relationships and a stochastic wind model to estimate wind speeds and lift coefficients. In [28] 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, since in the EKF framework input noise can not be handled independently of process noise. To overcome

978-1-5090-4494-8/17/$31.00 ©2017 IEEE

these issues this paper proposes a Moving Horizon Estimator (MHE). 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 a active research topic in the last years. One of the earlier descriptions can be found in [22]. [20] 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. The goal of this paper is to compare the performance of the MHE with the EKF especially regarding parameter drift.

A disadvantage of the MHE is the increasing computational cost with increasing window length. We therefore apply methods described in [13], [5] and [9] to efficiently solve the NLP.

II. Modeling

A. Aerodynamic Model

As in [28], we start with a simplified model of the aerodynamics of the UAV in z-direction in body frame:

$$f_z = KV_a^2 \left(C_{L,0} + \alpha C_{L,\alpha} + \nu(q)\right) \tag{1}$$

where f_z is the specific force in z-direction in body frame, measured by an accelerometer, and $K = \frac{\rho S}{2m}$ is a constant factor consisting of the air density ρ, the wing area S and the mass of the aircraft m. $C_{L,0}$ is the constant lift coefficient and $C_{L,\alpha}$ is the linear lift coefficient. This model of the aerodynamics in the z-direction is simplified and neglects the effects of drag on the body z-acceleration and linearizes the underlying trigonometric functions. The acceleration induced by the pitch rate q is accounted for in $\nu(q)$ and will be regarded as a disturbance later on. However, for small angles of attack and non-stall conditions the model errors induced by this simplifications are small and do not degrade the estimation performance. V_a is the airspeed and α is the angle of attack which are defined as:

$$\alpha = \tan^{-1}\left(\frac{w_r^b}{u_r^b}\right) \tag{2}$$

$$\beta = \sin^{-1}\left(\frac{v_r^b}{V_a}\right) \tag{3}$$

$$V_a = \|\boldsymbol{v}_r^b\| \tag{4}$$

Here $\boldsymbol{v}_r^b = \begin{pmatrix} u_r^b & v_r^b & w_r^b \end{pmatrix}^T$ is the velocity of the aircraft relative to the surrounding airmass decomposed in the body frame which is given by the wind triangle:

$$\boldsymbol{v}_r^b = \boldsymbol{v^b} - \boldsymbol{R}_n^b \boldsymbol{v}_w^n \tag{5}$$

Where $\boldsymbol{v}^b = (u, v, w)$ is the velocity over ground in body frame, $\boldsymbol{v}_w^n$ is the local wind velocity in NED frame and $\boldsymbol{R}_n^b$ is the rotation matrix from NED frame to body frame. We assume that the variables u_r^b and $\boldsymbol{v}^b$ are available as measurements from a pitot-static tube and a GNSS receiver, furthermore we assume that $\boldsymbol{R}_n^b$ is estimated from IMU and GNSS data. The wind velocity $\boldsymbol{v}_w^n$ is unknown and needs to be estimated in order to obtain the full vector $\boldsymbol{v}_r^b$.

B. Wind Model

In the following we use a time-discrete model where k denotes the time index and $\Delta \boldsymbol{x}_k$ the difference between $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k-1}$. We assume the wind velocity $\boldsymbol{v}_w^n$ to have two parts. A low-frequent steady part $\boldsymbol{v}_s^n$ and a high-frequent turbulent part v_t^n which are governed by the following discrete time dynamics [3, pp.55]:

$$\boldsymbol{v}_{w,k}^n = \boldsymbol{v}_{s,k}^n + \boldsymbol{v}_{t,k}^n \tag{6}$$

$$\Delta \boldsymbol{v}_{s,k}^n \approx 0 \tag{7}$$

The dynamic equations for the turbulent wind velocity (8) are given by the Dryden wind model [1]:

$$\Delta \boldsymbol{v}_{t,k}^n = -\Delta T V_{a,k} \begin{pmatrix} \frac{u_t^n}{L_u} \\ \frac{v_t^n}{L_v} \\ \frac{w_t^n}{L_w} \end{pmatrix}\Bigg|_k + \begin{pmatrix} \sigma_u \sqrt{2\Delta T \frac{V_a}{L_u}} \eta_{u_t} \\ \sigma_v \sqrt{2\Delta T \frac{V_a}{L_v}} \eta_{v_t} \\ \sigma_w \sqrt{2\Delta T \frac{V_a}{L_w}} \eta_{w_t} \end{pmatrix}\Bigg|_k \tag{8}$$

Where $\boldsymbol{\eta}_{\boldsymbol{v}_t,k} = \begin{bmatrix} \eta_{u_t} & \eta_{v_t} & \eta_{w_t} \end{bmatrix}^T$ is Gaussian white noise with standard deviation $\sigma = 1$ and the spatial wavelengths are given by:

$$L_{u,k} = L_{v,k} = \frac{h_k}{(0.177 + 0.0027 \cdot h_k)^{1.2}} \tag{9}$$

$$L_{w,k} = h_k \tag{10}$$

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

$$\sigma_{u,k} = \sigma_{v,k} = V_{w,G} \frac{1}{(0.177 + 0.0027 \cdot h_k)^{0.4}} \tag{11}$$

$$\sigma_w = 0.1 \cdot V_{w,G} \tag{12}$$

where $V_{w,G}$ is the wind speed measured 6 meters above ground.

C. State-space Model

The estimation problem consists of state and parameter estimation parts. The states to be estimated are the turbulent wind velocity 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, (cf. (24)):

$$\boldsymbol{x} = \begin{bmatrix} u_t^n & v_t^n & w_t^n \end{bmatrix}^T \tag{13}$$

$$\boldsymbol{p} = \begin{bmatrix} u_s^n & v_s^n & w_s^n & KC_{L_0} & KC_{L_\alpha} & \gamma \end{bmatrix}^T \tag{14}$$

Inputs to the observer are the measurements of the velocity over ground in the body frame from GNSS, the pitot - static tube airspeed, the rotation matrix estimated by an attitude and heading reference system (AHRS) and the altitude.

$$\tilde{\boldsymbol{u}} = \begin{bmatrix} \tilde{u}^b & \tilde{v}^b & \tilde{w}^b & \tilde{u}_r^m & \boldsymbol{R}_n^b & h \end{bmatrix}^T \tag{15}$$

The measurements of the inputs $\tilde{v}^b$and $\tilde{u}_r^b$ are affected by measurement noise

$$\tilde{\boldsymbol{v}}^{\boldsymbol{b}} = \boldsymbol{v}^b + \boldsymbol{\eta}_{v^b} \tag{16}$$

$$\tilde{u}_r^b = u_r^b + \eta_{u_r^b} \tag{17}$$

which is assumed to be additive Gaussian white noise. Regarding the rotation matrix, it is assumed that biases and noise of the used IMU are accounted for in the AHRS. Popular approaches for attitude and heading estimation are the EKF and nonlinear observers with a global or semi-global region of attraction [3], [12], [10]. 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 input sensor noise. We therefore assume that the given attitude estimates have negligible noise levels and are bias free. The altitude is affected by noise on the GNSS measurements, however the influence of this noise is limited since the altitude is only used to calculate the spatial wavelength of the wind gusts in (9)-(10) 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) can be applied (i.e. [3], [10]). The system setup is shown in Figure 1.

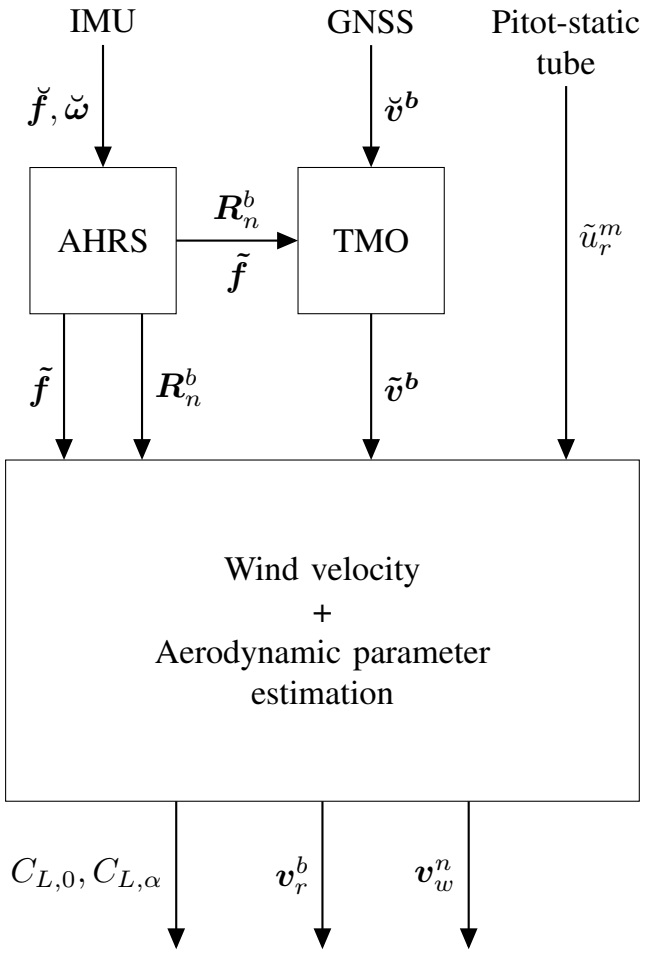

Fig. 1: System structure

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

$$\boldsymbol{u} = \tilde{\boldsymbol{u}} - \eta_{\boldsymbol{u}} \tag{18}$$

The state transition function is then given by:

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) = -\bar{V}_a \begin{pmatrix} \frac{u_t}{L_u} \\ \frac{v_t}{L_v} \\ \frac{w_t}{L_w} \end{pmatrix} \Bigg|_{\substack{\boldsymbol{u} \\ \boldsymbol{x} \\ p}} \tag{19}$$

The process noise input function is given by:

$$\boldsymbol{w}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\eta}_{\boldsymbol{v}_t,k}, \boldsymbol{p}) = \begin{pmatrix} \sigma_u \sqrt{2\Delta T \frac{\bar{V}_a}{L_u}} \eta_{u_t} \\ \sigma_v \sqrt{2\Delta T \frac{\bar{V}_a}{L_v}} \eta_{v_t} \\ \sigma_w \sqrt{2\Delta T \frac{\bar{V}_a}{L_w}} \eta_{w_t} \end{pmatrix} \Bigg|_{\substack{\boldsymbol{u}_k \\ \boldsymbol{x}_k \\ p}} \tag{20}$$

The predicted state is then given by:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \Delta T \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{p}) + \boldsymbol{w}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\eta}_{\boldsymbol{v}_t,k}, \boldsymbol{p}) \tag{21}$$

where ΔT is the sampling interval. Note that $\bar{V}_a$ is a low-frequency version of V_a which is only dependent on the velocity over ground $\boldsymbol{v}^b$ and the steady wind velocity $\boldsymbol{v}_s^n$ and independent of the turbulent wind velocity $\boldsymbol{v}_t^n$, as described in [28].

$$\bar{V}_a = \left\| \boldsymbol{v}^b - \boldsymbol{R}_n^b \boldsymbol{v}_s^n \right\| \tag{22}$$

The underlying assumption is that the shape of the low-pass filter described in (19) and (20) used to calculate the change in turbulent wind velocity $\Delta \boldsymbol{v}_t^n$ is not depending on the turbulent wind velocity itself.

Measurements used for correcting the predicted state vector are the specific force in z-direction in body frame $\tilde{f}_z$ measured by an accelerometer and the longitudinal velocity over ground in body frame $\tilde{u}^b$ which both are affected by additive white Gaussian measurement noise.

$$\boldsymbol{z}_k = \begin{bmatrix} \tilde{f}_z \\ \tilde{u}^b \end{bmatrix} = \begin{bmatrix} f_z + \eta_{f_z} \\ u^b + \eta_{u^b} \end{bmatrix} \tag{23}$$

where η_{f_z} is again assumed to be additive Gaussian white noise. The time varying measurement function is given by:

$$\boldsymbol{h}(\boldsymbol{x_k}, \boldsymbol{u_k}, \boldsymbol{p}) = \begin{bmatrix} -K\bar{V}_a^2 (C_{L_0} + C_{L_\alpha}\alpha) \\ \boldsymbol{d}_1 \boldsymbol{R}_n^b (\boldsymbol{v}_s^n + \boldsymbol{v}_t^n) + \gamma u_r^m \end{bmatrix} \tag{24}$$

The first measurement equation in (24) uses the z-accelerometer together with the aerodynamic model (1) to estimate the aerodynamic coefficients and the wind velocities via $\bar{V}_a$ and α. Here $\bar{V}_a$ is used in order to interpret fast changes in the acceleration f_z as to be induced by changes in the angle of attack α, which has higher frequency components instead of interpreting them as airspeed changes due to turbulence. This assumption provides different gradients when differentiating (24) with respect to $\boldsymbol{v}_s^n$ and $\boldsymbol{v}_t^n$ and therefore allows separate observation of these states. The second measurement equation utilizes the wind triangle (5) 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 [11].

As discussed in [28] and [11] the system is not uniformly observable at all time instants. Therefore attitude changes are necessary in order to ensure persistence of excitation.

III. Moving Horizon Estimator

The Moving Horizon Estimator (MHE) uses the sensor measurements gathered in a window of length L 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 $\boldsymbol{y}_j$ of a state space model of the system. This system is affected by process noise $\boldsymbol{\eta}_{\boldsymbol{v}_{t,j}}$ and input noise $\boldsymbol{\eta}_{\boldsymbol{u},j}$, summarized in a noise vector $\boldsymbol{\eta}_j$. 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 states, parameters and the process noise the following nonlinear program is solved in each time step:

$$\min_{\substack{\boldsymbol{x}_{k-L},\dots,\boldsymbol{x}_k \\ \boldsymbol{w}_{k-L},\dots,\boldsymbol{w}_k \\ \boldsymbol{\theta}_{k-L,0},\dots,\boldsymbol{\theta}_{k,d} \\ \boldsymbol{p}}} \left(\left\| \begin{matrix} \boldsymbol{x}_{k-L} - \hat{\boldsymbol{x}}_{k-L} \\ \boldsymbol{p} - \hat{\boldsymbol{p}}_{k-L} \end{matrix} \right\|^2_{\hat{\boldsymbol{P}}^{-1}_{k-L}} c + \cdots \right.$$
$$\left. \sum_{j=k-L}^{k} \|\boldsymbol{y}_j - \boldsymbol{h}(\boldsymbol{x}_j, \boldsymbol{u}_j, \boldsymbol{p})\|^2_{\boldsymbol{R}^{-1}} + \sum_{j=k-L}^{k-1} \|\boldsymbol{\eta}_j\|^2_{\boldsymbol{W}^{-1}} \right) \tag{25}$$

Where $\boldsymbol{u}_j = \tilde{\boldsymbol{u}}_{\boldsymbol{j}} - \boldsymbol{\eta}_{\boldsymbol{u},j}$, $\boldsymbol{y}_j = \boldsymbol{z}_{\boldsymbol{j}} - \boldsymbol{\eta}_{\boldsymbol{z},j}$ and $\boldsymbol{\eta}_j = \begin{bmatrix} \boldsymbol{\eta}_{\boldsymbol{v}_t,j} & \boldsymbol{\eta}_{\boldsymbol{u},j} & \boldsymbol{\eta}_{\boldsymbol{z}_j} \end{bmatrix}$. Note that all norms are 2-norms and are weighted by a matrix indicated in the subscript of the respected norm. 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 variable c is an additional tuning factor used to increase the weight on past data in periods of low excitation. The arrival cost is key to the stability of the MHE [13], [19] and will therefore be discussed in more detail in section III-A. The second term penalizes deviations of the output of the state space model from the measurement $\boldsymbol{z}_j$ with a weighting matrix $\boldsymbol{R}^{-1}$. The last term is penalizes the noise vector in the state transition function with a weight matrix $\boldsymbol{W}^{-1}$.

In this paper we chose a direct collocation formulation for the nonlinear program, proposed for model predictive control in [4], [5], [26]. This formulation uses 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 indicated as $\mathcal{L}_1 \dots \mathcal{L}_d$. 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 weights $\boldsymbol{\theta}$. On each collocation point the gradient condition (26) and at the end of an collocation interval the continuity condition (27) is added to the equality constraints of the NLP.

$$\sum_{r=0}^{d} \dot{\boldsymbol{\Gamma}}_r(\tau_i)\boldsymbol{\theta}_{j,r} = \boldsymbol{f}(\boldsymbol{\theta}_{j,i}, \boldsymbol{u}_j, \boldsymbol{p}) + \frac{1}{\Delta T}\boldsymbol{w}(\boldsymbol{\theta}_{j,i}, \boldsymbol{p}, \boldsymbol{u}_j, \boldsymbol{\eta}_{\boldsymbol{v}_{t,j}}) \tag{26}$$

$$\sum_{i=0}^{d} \boldsymbol{\Gamma}_i(1)\boldsymbol{\theta}_{j,i} = \boldsymbol{\theta}_{j+1,0} \tag{27}$$

$$\boldsymbol{x}_j = \boldsymbol{\theta}_{j,0}, \qquad \boldsymbol{x}_{j+1} = \boldsymbol{\theta}_{j,d}$$
$$\text{for } i = 0, \dots, d \text{ and } j = k-L, \dots, k-1$$
$$\boldsymbol{p}_{min} \leq \boldsymbol{p} \leq \boldsymbol{p}_{max}$$

where:

$$\boldsymbol{\tau} = \begin{bmatrix} 0 & \mathcal{L}_1 & \dots & \mathcal{L}_d \end{bmatrix} \tag{28}$$

$$\boldsymbol{\Gamma}_i(t) = \prod_{r=0, r\neq i}^{d} \frac{t - \tau_r}{\tau_i - \tau_r} \tag{29}$$

The result is a very large but also very sparse NLP, which can be solved efficiently by dedicated NLP solvers like IPOPT [27] which has been used in this paper.

One of the benefits of the MHE is the possibility to impose constrains on the estimated states, noise variables and parameters. In this case we will only impose constraints on the estimated lift coefficients and the pitot-static tube scaling factor since bounds for these parameters can be roughly estimated, whereas wind velocity bounds are unknown.

A. Approximation of the Arrival Cost

As mentioned before the arrival cost term is key to the stability of the estimator. Ideally one would like to use the exact arrival cost, i.e. (25) is writen as:

$$\min \sum_{j=0}^{k} \|\boldsymbol{y}_j - \boldsymbol{h}(\boldsymbol{x}_j, \boldsymbol{p}, \boldsymbol{u}_j)\|^2_{\boldsymbol{R}^{-1}} + \sum_{j=0}^{k} \|\boldsymbol{\eta}_j\|\|^2_{\boldsymbol{W}^{-1}} \tag{30}$$

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. [17] consider different Kalman filter type 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 [18]. 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 $\boldsymbol{p}^*_{k-1}$ estimates of the previous solution of the NLP (25), with $v = k - L$. The augmented state vector includes also the first moments of the process, measurement and input

noises, which are, following the previous assumptions, equal to zero.

$$\boldsymbol{x}^a_{v-1|v-1} = \begin{bmatrix} \boldsymbol{x}^*_{v-1|v-1} & \boldsymbol{p}^*_{k-1} & \mathbf{0}_{q_x} & \mathbf{0}_{q_p} & \mathbf{0}_r & \mathbf{0}_u \end{bmatrix}^T \tag{31}$$

The corresponding covariance matrix of the estimation error is:

$$\boldsymbol{P}_{v-1|v-1} = \begin{bmatrix} \boldsymbol{P}^{\boldsymbol{x}}_{v-1|v-1} & \mathbf{0}_{[9x9]} & \mathbf{0}_{[9x2]} & \mathbf{0}_{[9x4]} \\ \mathbf{0}_{[9x9]} & \boldsymbol{Q} & \mathbf{0}_{[9x2]} & \mathbf{0}_{[9x4]} \\ \mathbf{0}_{[2x9]} & \mathbf{0}_{[2x9]} & \boldsymbol{R} & \mathbf{0}_{[2x4]} \\ \mathbf{0}_{[4x9]} & \mathbf{0}_{[4x9]} & \mathbf{0}_{[4x2]} & \boldsymbol{W}_u \end{bmatrix} \tag{32}$$

Where $\boldsymbol{Q}$ is the covariance matrix of the process noise, $\boldsymbol{R}s$ is the covariance matrix of the measurement noise and $\boldsymbol{W}_u$ is the covariance matrix of the input noise. We then generate a set of sigma points:

$$\begin{aligned} \boldsymbol{\chi}^a_{v-1} &= \boldsymbol{X}^a_{v-1|v-1} + \cdots \\ &\begin{pmatrix} \mathbf{0}_{[24x1]} & \sqrt{(n^a+\kappa)\boldsymbol{P}_{v-1|v-1}} & -\sqrt{(n^a+\kappa)\boldsymbol{P}_{v-1|v-1}} \end{pmatrix} \end{aligned} \tag{33}$$

and corresponding weights:

$$\Omega_i = \begin{cases} \frac{\kappa}{n_a+\kappa} & \text{if } i = 1 \\ \frac{1}{2(n_a+\kappa)} & \text{otherwise} \end{cases} \tag{34}$$

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 [18]. $\boldsymbol{X}^a_{v-1|v-1}$ 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:

$$\begin{aligned} \boldsymbol{\chi}^x_v = \;& \boldsymbol{\chi}^x_v + \Delta T \boldsymbol{f}(\boldsymbol{\chi}^x_{v-1}, \tilde{\boldsymbol{u}}_{v-1} - \boldsymbol{\chi}^u_{v-1}, \boldsymbol{\chi}^p_{v-1}) + \cdots \\ & \boldsymbol{w}(\boldsymbol{\chi}^x_{v-1}, \tilde{\boldsymbol{u}}_{v-1} - \boldsymbol{\chi}^u_{v-1}, \boldsymbol{\chi}^{q_x}_{v-1}, \boldsymbol{\chi}^p_{v-1}) \end{aligned} \tag{35}$$

$$\boldsymbol{\chi}^p_v = \boldsymbol{\chi}^p_{v-1} + \boldsymbol{\chi}^{q_p}_{v-1} \tag{36}$$

$$\boldsymbol{\gamma}_v = \boldsymbol{h}(\boldsymbol{\chi}^x_v, \tilde{\boldsymbol{u}}_v - \boldsymbol{\chi}^u_{v-1}, \boldsymbol{\chi}^p_k) + \chi^r_{v-1} \tag{37}$$

With:

$$\boldsymbol{\chi}^a_{v-1} = \begin{bmatrix} \boldsymbol{\chi}^x_{v-1} & \boldsymbol{\chi}^p_{v-1} & \boldsymbol{\chi}^{q_x}_{v-1} & \boldsymbol{\chi}^{q_p}_{v-1} & \boldsymbol{\chi}^r_{v-1} & \boldsymbol{\chi}^u_{v-1} \end{bmatrix}^T \tag{38}$$

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

$$\boldsymbol{x}_{v|v-1} = \sum_{i=1}^{2n^a+1} \Omega_i \boldsymbol{\chi}^x_{i,v} = \hat{\boldsymbol{x}}_{k-L} \tag{39}$$

$$\boldsymbol{p}_{v|v-1} = \sum_{i=1}^{2n^a+1} \Omega_i \boldsymbol{\chi}^p_{i,v} = \hat{\boldsymbol{p}}_{k-L} \tag{40}$$

$$\hat{\boldsymbol{z}}_{\boldsymbol{v}} = \sum_{i=1}^{2n^a+1} \Omega_i \gamma_{i,v} \tag{41}$$

$$\boldsymbol{P}^{\boldsymbol{x}}_{v|v-1} = \sum_{i=1}^{2n^a+1} \Omega_i \left[\boldsymbol{\chi}^x_{i,v} - \boldsymbol{x}_{v|v-1}\right] \left[\boldsymbol{\chi}^x_{i,v} - \boldsymbol{x}_{v|v-1}\right]^T \tag{42}$$

$$\boldsymbol{P}_{y,v} = \sum_{i=1}^{2n^a+1} \Omega_i \left[\boldsymbol{\gamma}_{i,v} - \hat{\boldsymbol{y}}_{\boldsymbol{v}}\right] \left[\boldsymbol{\gamma}_{i,v} - \hat{y}_v\right]^T \tag{43}$$

$$\boldsymbol{P}_{yz,v} = \sum_{i=1}^{2n^a+1} \Omega_i \left[\boldsymbol{\chi}^x_{i,v} - \boldsymbol{x}_{v|v-1}\right] \left[\boldsymbol{\gamma}_{i,v} - \hat{y}_v\right]^T \tag{44}$$

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

$$\boldsymbol{K}_v = \boldsymbol{P}_{xy,v} \boldsymbol{P}^{-1}_{y,v} \tag{45}$$

$$\boldsymbol{P}^{\boldsymbol{x}}_{v|v} = \boldsymbol{P}_{v|v-1} - \boldsymbol{K}_v \boldsymbol{P}_{y,v} \boldsymbol{K}^T_v = \hat{\boldsymbol{P}}_{k-L} \tag{46}$$

IV. Simulation Setup

A. UAV Simulation

All simulations were performed using a simulation of the X8 flying wing and a simulated autopilot. The simulator and the autopilot are based on Beard and McLain [3, Chap. 4 and 6] and implemented in Matlab / Simulink. In the aircraft simulation a more complex non-linear model of the aerodynamics is used (ssee [3, pp.44]). The aerodynamic forces are thus given by:

$$\zeta = \frac{1 + e^{-M(\alpha-\alpha_0)} + e^{M(\alpha+\alpha_0)}}{\left(1 + e^{-M(\alpha-\alpha_0)}\right)\left(1 + e^{M(\alpha+\alpha_0)}\right)} \tag{47}$$

$$\begin{aligned} C_L(\alpha) =& (1-\zeta)(C_{L,0} + C_{L,\alpha}\alpha) + \\ & \zeta(2\,\mathrm{sign}(\alpha)\sin(\alpha)^2\cos(\alpha)) \end{aligned} \tag{48}$$

$$F_L = \frac{1}{2}\rho V_a^2 S(C_L(\alpha) + C_{L,q} c/(2V_a) q + C_{L,\delta_e}\delta_e) \tag{49}$$

$$\begin{aligned} C_{D,\alpha} =& C_{D,p} + (1-\zeta)\frac{(C_{L,0} + C_{L,\alpha}\alpha)^2}{\pi e(b^2/S)} + \\ & \zeta 2\,\mathrm{sign}(\alpha)\sin(\alpha)^3 \end{aligned} \tag{50}$$

$$F_D = \frac{1}{2}\rho V_a^2 S\left(C_{D,\alpha} + C_{D,q}\frac{c}{2V_a}q + C_{D,\delta_e}\left|\delta_e\right|\right) \tag{51}$$

$$\begin{pmatrix} f_x \\ f_z \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} -F_D/m \\ -F_L/m \end{pmatrix} \tag{52}$$

where M and α_0 are positive constants, c is the mean aerodynamic chord of the wing, q is the pitch-rate and δ_e is the elevator deflection and $C_{L,q}, C_{L,\delta_e}, C_{D,q}, C_{D,\delta_e}$ are their respective lift and drag coefficients. Figure 2 shows the resulting lift coefficient for different angles of attack.

Wind velocity is simulated following Beard and McLain [3,

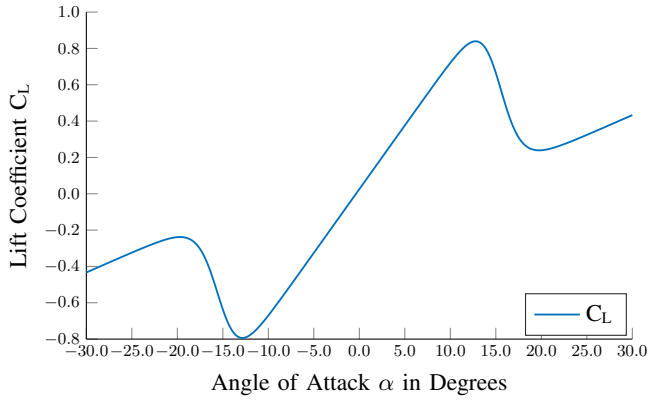

Fig. 2: Nonlinear model of the lift coefficient

pp.55] as the sum of a steady and a turbulent wind velocity component, where the turbulence is generated by passing white noise through a low pass filter. The filter is designed in the way described in section II-B according to the Dryden model. The steady wind velocity and the wind velocity at $6m$ above ground, needed for the turbulence generation, were set to $6m/s$, the wind direction was set to $90°$.

B. Moving Horizon Estimator

As a simulation and computation environment Matlab/ Simulink was used. Within Matlab, Casadi [2] 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. Limits for the coefficient estimates were set to:

$$\begin{bmatrix} -0.1 \\ 0 \\ 0 \end{bmatrix} \leq \begin{bmatrix} KC_{L,0} \\ KC_{L,\alpha} \\ \gamma \end{bmatrix} \leq \begin{bmatrix} 0.2 \\ 1 \\ 2 \end{bmatrix} \tag{53}$$

C. Initial Conditions and Tuning

The initial condition for the state and parameter vectors were set to:

$$\begin{aligned} \boldsymbol{x}_0 &= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \\ \boldsymbol{p}_0 &= \begin{bmatrix} 0 & 0 & 0 & 0.01 & 0 & 1 \end{bmatrix}^T \end{aligned} \tag{54}$$

The inital condition for the error covariance matrix of the arrival cost was set to:

$$\begin{aligned} \boldsymbol{P}_0 = diag\,\big[&10^{-5} \quad 10^{-5} \quad 10^{-5} \quad 10^{-2} \quad 10^{-2} \\ &10^{-5} \quad 10^{-5} \quad 3 \cdot 10^{-3} \quad 10^{-8} \quad \big] \end{aligned} \tag{55}$$

The process noise covariance matrix $\boldsymbol{Q}$ for the arrival cost approximation was set to:

$$\begin{aligned} \boldsymbol{Q} = diag\,\big[&10^{-2} \quad 10^{-2} \quad 1 \quad 10^{-6} \quad 10^{-6} \\ &10^{-12} \quad 10^{-12} \quad 10^{-20} \quad 10^{-20} \quad \big]\,\Delta T \end{aligned} \tag{56}$$

The first 3 parameters govern the noise power of the inputs to the noise transfer function (20). A finding during tuning was that a somewhat higher noise power for the vertical turbulent wind velocity in comparison to the horizontal component yielded better performance of the estimator. The other noise parameters describe the uncertainty in the parameters. It was assumed that the steady wind velocities in the horizontal plane are somewhat more likely to change than in the vertical direction, resulting in higher covariances for these parameters. The linear lift coefficient and the pitot-static tube scaling factor are unlikely to change over time and therefore lower covariances were chosen. The estimator was tuned so that model mismatch in (1) results in a change in the constant lift coefficient estimate, therefore the process noise covariance was set somewhat higher for this parameter estimate. The covariance matrix of the measurement noise is chosen to:

$$\boldsymbol{R} = diag(1, 10^{-1}) \tag{57}$$

in normal flight and to

$$\boldsymbol{R} = diag(1.2, 10^{-1}) \tag{58}$$

if $\|q_j\| \geq 0.2\frac{rad}{s}$ for $j = k - L, \ldots, k$. This covariance matrix is important in tuning the MHE. It quantifies the expected uncertainties in the measurement equation induced by noise in the sensors and model mismatch. For the measurement noise was modeled as zero-mean Gaussian white noise. For the pitot-static tube the measurement the noise standard deviation was set to $0.001m/s$, for the z- accelerometer measurement the standard deviation was set to $0.1m/s^2$ and for the ground velocity measurement to $0.001m/s$. It was assumed in the simulation that the AHRS system supplies the estimator with accurate attitude angles with negligible noise levels. The weighting matrix of the process and input noise $\boldsymbol{W}$ was chosen to:

$$\boldsymbol{W} = diag\,\begin{bmatrix} 10^{-2} & 10^{-2} & 1 & 10^{-3} & 10^{-3} & 10^{-3} & 10^{-3} \end{bmatrix} \tag{59}$$

The first 3 noise parameters are set as for the process noise covariance matrix $\boldsymbol{Q}$ and the input noise covariances were set to the actual covariances of the simulated sensor noise.

D. Autopilot Maneuver

We simulate a normal take off and some circles, which are typically performed by UAV pilots before handing control over to an autopilot. The UAV performs one circle per minute. The UAV is commanded to fly at an altitude of $150m$, with a course angle of $60°$. Later the UAV changes altitude to $250m$ and course $120°$ at two different time instants. The input signals can be seen in Figure 3 and Figure 4. As discussed before, attitude changes are necessary to excite the system and ensure observability. The autopilot maneuvers were chosen so that is observability is satisfied in the first part of the flight. Afterwards a steady course and altitude comments results in poor observability and allows assessment of parameter estimate drift in these circumstances.

V. Simulation Results

The simulations were done for different window lengths L where a window length of $L = 20$ provided a good

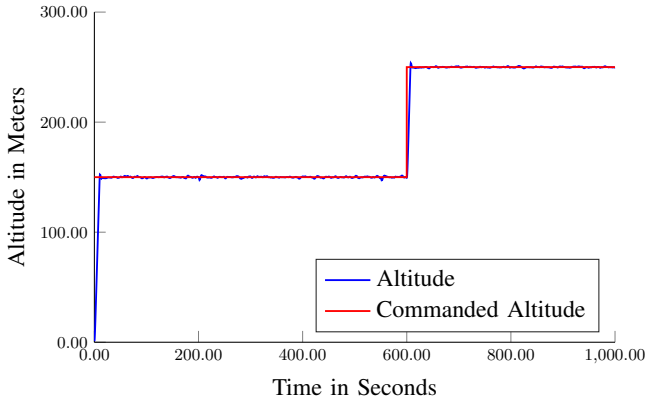

Fig. 3: Commanded and resulting altitude of the UAV

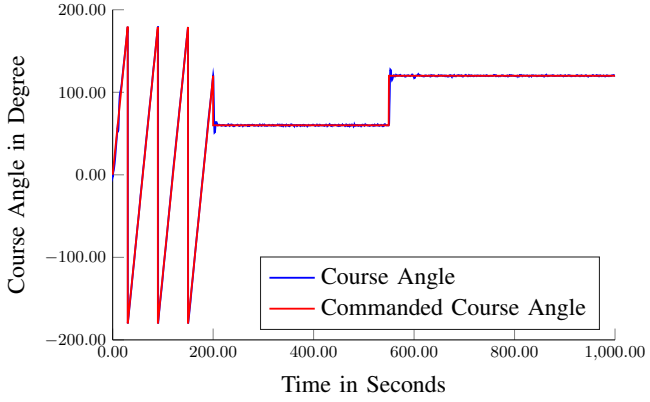

Fig. 4: Commanded and resulting course angle of the UAV

trade off between computation time and estimation performance. Computation time for this window length was 0.0520 s/iteration on a 2.5 GHz Intel Core I7 CPU, allowing real-time computation up to a sampling frequency of 19.24 Hz. The following simulation results were achieved with a sampling frequency of 10 Hz. Real-time computation was not a focus of this paper and we expect that with more efficient implementation the runtime can be improved considerably.

A. Lift Coefficient Estimates

In Section III we introduced a factor c in order to weigh the arrival cost differently in regions of low excitations. Figure 5 shows the estimates of the linear lift coefficient for different c. The coefficient estimates converge for all values of c other than $c = 1$, where the constraint is met. The other trajectories converge quickly to an interval around the reference and stay there for the entire simulation run. The best results could be achieved with a factor $c = L = 20$, which will be used subsequently.

Figure 6 shows the estimates for both lift coefficients with confidence intervals. The linear lift coefficient estimate $C_{L,\alpha}$ converges quickly to an estimate close to the reference value. Afterwards it stays there and shows very little drift. Overall the estimation error is quite low. The constant lift coefficient estimates $C_{L,0}$ converges also very fast towards its reference value. It remains within an interval for the entire

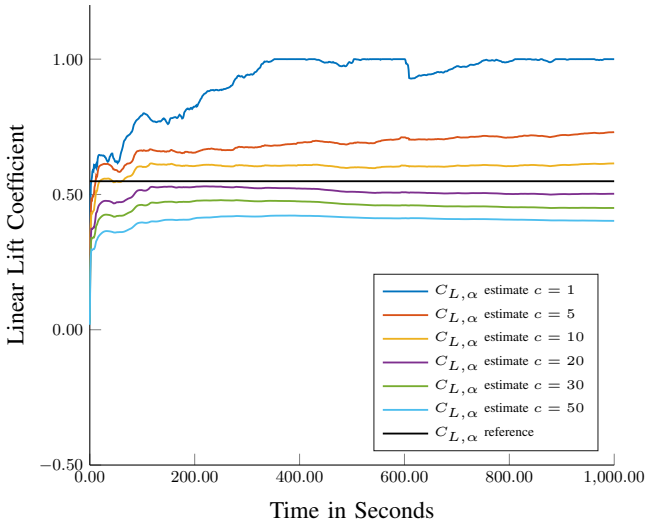

Fig. 5: Estimates of linear lift coefficient for different values of c

simulation run. Some small errors occur which are due to model mismatch and compensation for errors in the linear lift coefficient estimate.

B. Wind Velocity Estimates

Figure 7 shows the estimation error in the wind velocity estimates in x, y and z direction in body frame. The results show lower estimation error for the wind velocity estimate in x-direction which is expected due to the pitot-static tube measurement in that direction. For the y-direction the performance is not as good since estimates only rely on previous data since no measurement in that direction is available. Notably the estimate of the wind velocity estimate in z-direction which is mainly dependent on the z-acceleration measurement and the aerodynamic model (1) performs very well after a short convergence phase.

C. Airspeed, Angle of Attack and Sideslip Angle

In the following we show the estimates for Airspeed, Angle of Attack and Sideslip Angle and compare them to the estimates obtained from an Extended Kalman Filter using the same tuning as in [28]. Note that the EKF results presented here are not directly comparable to [28] since this paper uses more realistic simulation of the input noise, resulting in a decreased performance of the EKF. For the root mean square error calculation the first $100s$ were omitted, since otherwise larger errors occurring during the convergence phase distort the overall result.

Using the wind velocity estimates and the GNSS velocity, the airspeed V_a can be calculated using equation (4). The resulting estimation error is shown in Figure 8. The figure shows, that after a short convergence phase the error stays below $0.4m/s$.

To calculate the angle of attack two methods are available. One is the calculation of the angle of attack using the GNSS

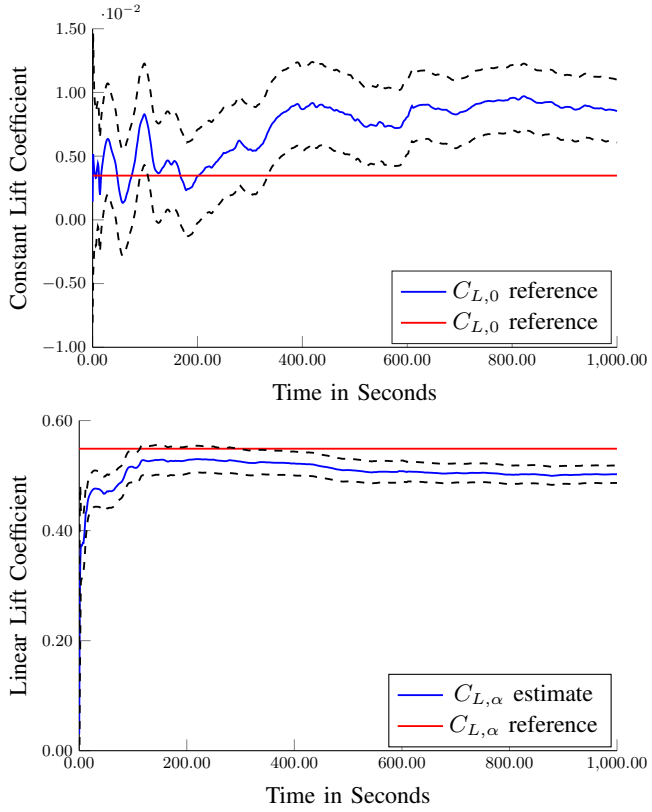

Fig. 6: Estimates of static and linear lift coefficient for $c = 20$, dashed black lines indicate 3 σ interval

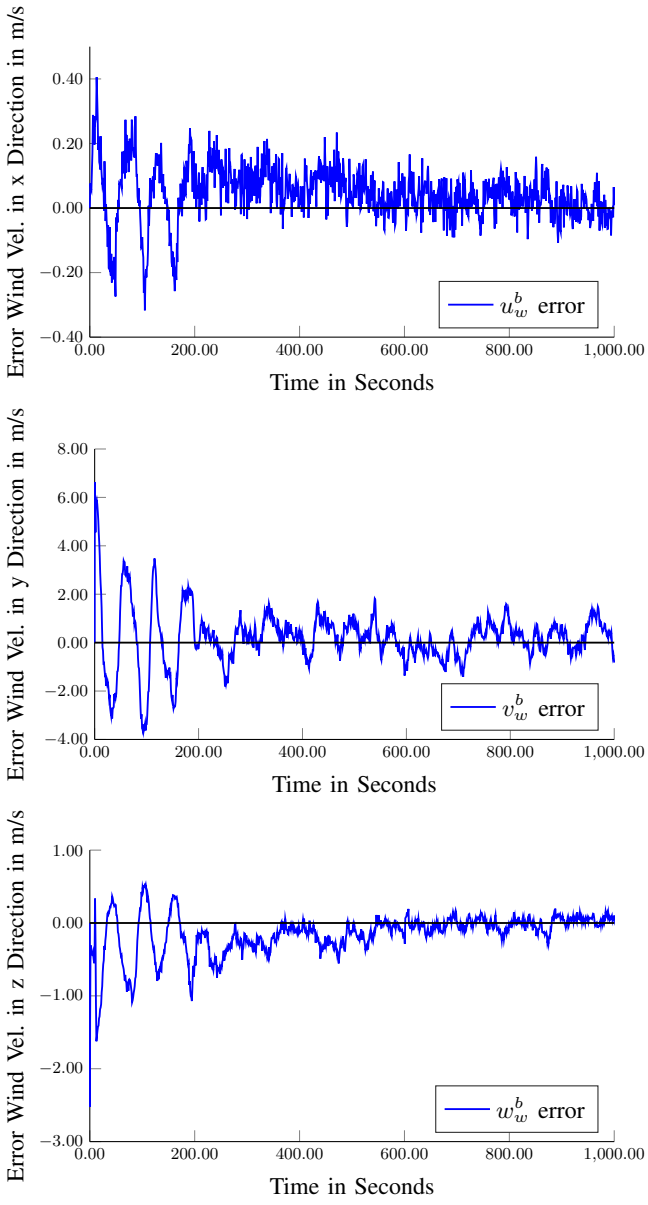

Fig. 7: Wind velocity estimation error in body frame

velocity measurement, the wind velocity estimate and equation (2). The other possibility to calculate the angle of attack is by inverting the aerodynamic model (1). The estimation errors from both methods are shown in Figure 9. In this case the performance for the calculation via the aerodynamic model showed better results than the calculation via the relative wind velocities. This is due to the good airspeed and coefficient estimates. If the wind velocity in y direction in body frame is changing very frequently the quality of the airspeed estimate might deteriorate, in these cases the calculation via (2) would have a superior performance due to its independence of the lateral wind velocity. Both estimates show clearly superior performance compared to the estimate obtained via the EKF as in [28].

For completeness we also show the estimation error of the sideslip angle estimate in figure 10. As discussed before no information about the lateral wind is available in this system setup. Therefore the estimator can only rely on prior data. This lack of information explains the larger sideslip angle estimation error compared to the angle of attack estimate. However the sideslip angle estimate also shows clear performance advantages in comparison to the EKF estimate.

The performance advantage of the MHE compared to the EKF proved to be consistent for different wind scenarios and autopilot maneuvers. The MHE also showed less parameter drift, generally coping better with regions of low observability resulting in less need for excitations. When increasing simulated sensor noise levels, the MHE outperforms the EKF significantly due to the possibility to incorporate input noise into the system model.

VI. Conclusion

This paper presents a method to estimate wind velocities and aerodynamic coefficients in order to calculate angle of attack, sideslip angle and the airspeed. This is done by utilizing measurements of a standard sensor suite together with aerodynamic, kinematic and wind models. As an estimator 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 estimator was tested using a UAV simulator steered by an autopilot model. The simulation results showed generally

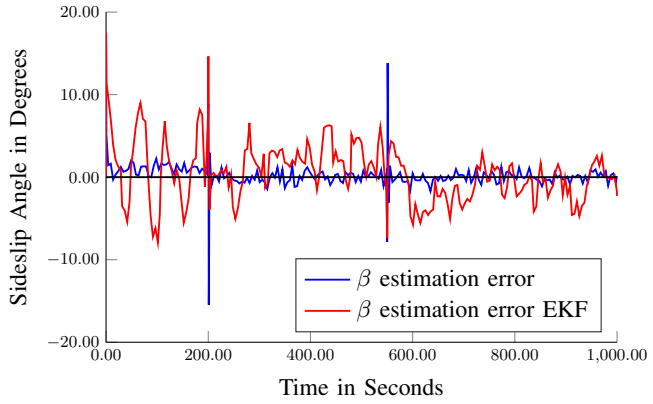

Fig. 10: Sideslip angle estimation error $RMSE = 1.06°, RMSE_{EKF} = 9.34°$

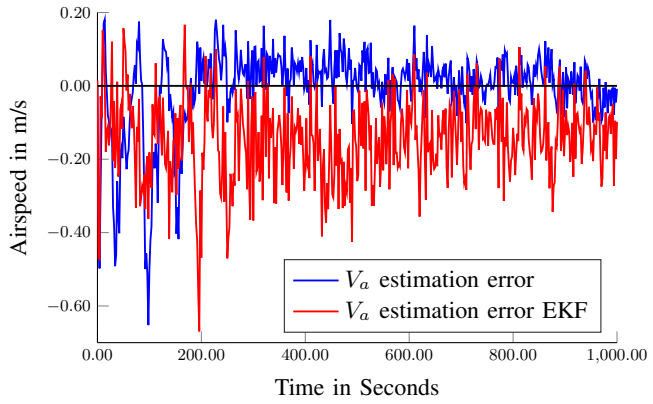

Fig. 8: Airspeed estimation error, $RMSE = 0.08m/s, RMSE_{EKF} = 0.18m/s$

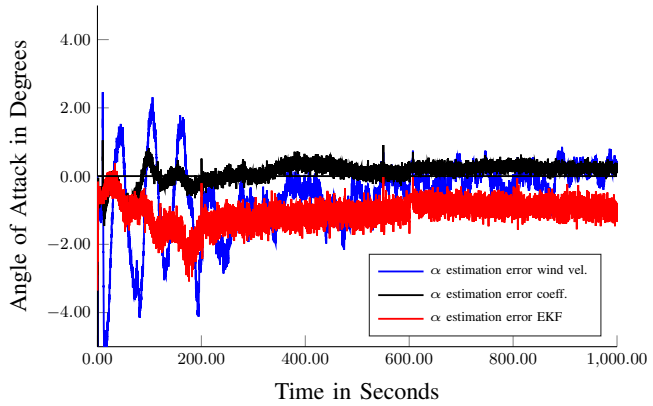

Fig. 9: Comparison of Angle of Attack calculation methods, $RMSE_w = 0.93°, RMSE_c = 0.25°, RMSE_{EKF} = 1.16°$

good performance of the wind velocity estimates and the lift coefficient estimates. This allows computation of the angle of attack and airspeed estimates which both show very low estimation errors. For the angle of attack calculation two different methods were proposed which are applicable in different scenarios. An estimate of the sideslip angle is also available, however due to the lack of a lateral wind speed measurement performance is limited. All airflow variable estimates from the MHE showed lower estimation error compared to the EKF estimates, clearly showing the advantages of the MHE formulation.

VII. Acknowledgments

This project has received funding from the European Unions Horizon 2020 research and innovation programme 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 Kristoffer Gryte for the development of the UAV simulator.

References

[1] MIL-STD-1797A: Flying Qualities of Piloted Aircraft. Technical report, 2004.
[2] Joel Andersson. *A General Purpose Software Framework for Dynamic Optimization*. Phd thesis, KU Leuven, 2013.
[3] Randal W. Beard and Timothy W. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012.
[4] Lorenz T Biegler. Solution of Dynamic Optimization Problems and Orthogonal Collocation. 8(3):0–4, 1984.
[5] 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.
[6] K. T. Borup, T. I. Fossen, and T. A. Johansen. A Nonlinear Model-Based Wind Velocity Observer for Unmanned Aerial Vehicles. In *10th IFAC Symp. Nonlinear Control Syst.*, Monterrey, CA, USA, 2016.
[7] 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.
[8] Jean-philippe Condomines, Murat Bronz, and Gautier Hattenberger. Experimental Wind Field Estimation and Aircraft Identification Experimental Wind Field Estimation and Aircraft Identification. In *IMAV 2015 Int. Micro Air Veh. Conf. Flight Compet.*, 2015.
[9] Moritz Diehl, H Ferreau, and Niels Haverbeke. Efficient numerical methods for nonlinear MPC and moving horizon estimation. *Nonlinear Model Predict. Control*, pages 391–417, 2009.
[10] Havard 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.
[11] Tor A Johansen, Andrea Cristofaro, Kim Sørensen, Jakob M Hansen, and Thor I Fossen. On estimation of wind velocity , angle-of-attack and sideslip angle of small UAVs using standard sensors. In *Int. Conf. Unmanned Aircr. Syst.*, Denver, 2015.
[12] 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.
[13] 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.
[14] 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.*, volume 2005, pages 1896–1901. IEEE, 2005.
[15] Jack W Langelaan, Nicholas Alley, and James Neidhoefer. Wind Field Estimation for Small Unmanned Aerial Vehicles. *J. Guid. Control Dyn.*, 34:1016–1030, 2011.
[16] 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.
[17] 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.

[18] 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.

[19] 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.

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

[21] 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.*, number 5201, pages 1–11, 2013.

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

[23] 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. American Institute of Aeronautics and Astronautics.

[24] 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. ICUAS 2015*, pages 723–732, 2015.

[25] 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.*, number 0645, 2016.

[26] 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.

[27] 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.

[28] 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 *2016 Int. Conf. Unmanned Aircr. Syst. ICUAS 2016*, pages 624–632, Arlington, VA, 2016.