Dead Reckoning of a Fixed-Wing UAV with Inertial Navigation Aided by Optical Flow

Lorenzo Fusini, Tor A. Johansen and Thor I. Fossen
Norwegian University of Science and Technology
Centre for Autonomous Operations and Systems
Department of Engineering Cybernetics
Trondheim, Norway
E-mail: {lorenzo.fusini, tor.arne.johansen, thor.fossen}@ntnu.no

***Abstract*—This paper provides experimental results for dead reckoning of a fixed-wing UAV using a non-linear observer (NLO) and a more recent tool called eXogenous Kalman Filter (XKF), which uses the NLO itself as a first-stage filter. The sensors used are an IMU (accelerometers, inclinometers, and rate gyros), a camera, and an altimeter; the observed states are position, velocity, and attitude. A machine vision system provides the body-fixed velocity of the UAV. Although the calculated velocity results affected by a bias, it is necessary both for estimating the attitude and for bounding the rate of divergence of the position during dead reckoning. Gyro, accelerometer, and optical flow (OF) velocity biases are estimated, but only as long as GNSS is available. When dead reckoning begins, they are frozen at their last calculated value. The experimental results show that the position error grows at a bounded rate with the proposed estimators.**

I. Introduction

In the field of navigation, the most popular methods for estimating position, velocity, and attitude of a vehicle have been the Kalman filter and its variants [1]–[3]: they are characterized by good performance at the cost of a relatively challenging tuning process, heavy computational footprint, and unclear stability properties in the non-linear case. More recently, non-linear observers have been developed to overcome the drawbacks of solutions based on the Kalman filter, offering computationally light algorithms with well-defined stability properties, often global, that make them robust to disturbances and uncertain initialization [4]–[10]. A recent addition to this set of tools is the exogenous Kalman filter [11]–[13], a two-stage filter made of a non-linear observer in cascade with a linearized time-varying Kalman filter that combines the advantages of each, while also overcoming their invididual weaknesses.

When direct information about the current position of the vehicle is not available, one way to obtain an estimate of it is through dead reckoning. At its core the method requires to integrate twice a measured acceleration, usually available via an IMU, and/or once a measured velocity: in either case, the method suffers from inaccuracies caused by the integration of noisy and biased measurements and by the approximation of the calculations themselves, leading to inevitable drifts in the calculated position. Moreover the IMU measurements must be rotated from body to Earth-fixed frame, which requires attitude estimates. For these reasons it is important to have a good dead reckoning algorithm, in order to reduce as much as possible the displacement between two successive position fixes.

A camera can provide invaluable help in this context. It allows to obtain an estimate of the motion of the vehicle, a process known as visual odometry. There exist many contributions to such topic, especially in the field of vision-aided navigation [14]–[17]. Machine vision can be use to calculate the OF from recorded images and to integrate it with other sensors to obtain the body-fixed velocity [9], [18].

The authors have previously proposed a globally exponentially stable (GES) non-linear observer for vision-aided UAV dead reckoning [19],

demonstrating that the use of a velocity aiding helps to bound the rate of divergence of the calculated position.

A. Contribution of this paper

In this paper an NLO and an XKF for dead reckoning are tested on experimental data and their results compared. Both estimators use the same set of sensors, described in Sec. II-A. The NLO is the GES observer proposed in [19]. Estimators for accelerometer bias and body-fixed velocity bias, together with compensation for the rotation of Earth, are included in the design to increase the accuracy of the estimates and reduce the position drift. The XKF is designed using the same NLO as the first-stage filter. The results help understand the problems related to each estimator in this context, and how to possibly overcome them. The method is experimentally validated by 10 minutes of dead reckoning of a fixed-wing UAV.

II. Notation and Preliminaries

Vectors and matrices are represented by lowercase and uppercase letters, respectively. X^{-1}, X^{+}, and $\text{tr}(X)$ denote the inverse, pseudo-inverse, and trace of a matrix, respectively, and X^T and x^T the transpose of a matrix and vector, respectively. An estimated value of x is represented as $\hat{x}$ and the estimation error is defined as $\tilde{x} = x - \hat{x}$; similarly, for a matrix, $\tilde{X} = X - \hat{X}$. The operator $\|\cdot\|$ denotes the Euclidean norm for vectors, and I_n is the identity matrix of order n. The function $\text{sat}(\cdot)$ performs a component-wise saturation of its vector or matrix argument to the interval $[-1, 1]$. The operator $S(x)$ transforms the vector x into the corresponding skew-symmetric matrix, such that $S(x)y = x \times y$. The inverse operation is denoted as $\text{vex}(\cdot)$, such that $\text{vex}(S(x)) = x$. For a square matrix A, its skew-symmetric part is represented by $\mathbb{P}_a(A) = \frac{1}{2}(A - A^T)$.

The reference frames considered in the paper are the body-fixed frame $\{\text{B}\}$ and the North-East-Down (NED) frame $\{\text{N}\}$ (Earth-fixed, considered inertial). The rotation from frame $\{\text{B}\}$ to $\{\text{N}\}$ is represented by the matrix $R_b^n \equiv R \in SO(3)$, where $SO(3)$ represents the Special Orthogonal group.

A vector decomposed in $\{\text{B}\}$ and $\{\text{N}\}$ has superscript ^{b} and ^{n} respectively. The gravity vector is defined as $g^n = [0, 0, g]$, with g being the local gravitational acceleration. The position vector decomposed in $\{\text{N}\}$ is $p^n = [p_N^n, p_E^n, p_D^n]^T$. The greek letters ϕ, θ, and ψ represent the roll, pitch, and yaw angles respectively, defined according to the zyx convention for principal rotations [20]. Subscript $_m$ indicates a quantity measured by a sensor. The symbol b_y identifies the bias on the measurement y. The Earth's rotation rate is $\omega_e = 7.292115 \cdot 10^{-5}$ rad/s; decomposed in the NED frame it assumes a value that depends on the latitude μ, that is $\omega_e^n = [\cos\mu, 0, -\sin\mu]^T \omega_e$.

A. Measurements and Sensors

The sensor system consists of an IMU (accelerometers, inclinometers, and rate gyros), a camera, an altimeter, and a GNSS receiver, providing the following information:

- *IMU*: biased angular velocity $\omega_m^b = \omega^b + b_\omega$; biased specific force $f_m^b = f^b + b_f$, which includes the effect of gravity; roll ϕ and pitch θ angles;
- *altimeter*: altitude p_D^n;
- *camera*: biased body-fixed velocity $v_m^b = v^b + b_v$ (depending on the machine vision system implemented, additional sensors might be required to be able to calculate v_m^b without bias);
- *GNSS receiver*: NED position p^n and velocity v^n.

Further information on the actual sensors used in the experiment is presented in Section VI.

B. Problem formulation

The kinematic system to observe is

$$\dot{R} = RS(\omega_m^b - b_\omega - R^T\omega_e^n) \tag{1a}$$
$$\dot{b}_\omega = 0 \tag{1b}$$
$$\dot{p}^n = v^n \tag{1c}$$
$$\dot{v}^n = R(f_m^b - b_f) + g^n - 2S(\omega_e^n)v^n \tag{1d}$$
$$\dot{b}_f = 0 \tag{1e}$$

with the output map

$$y_1 = p_D^n \tag{2a}$$
$$y_2 = R^T v^n + b_v \tag{2b}$$
$$y_3 = -R^T g^n + (\omega^b - b_\omega - R^T \omega_e^n) \times v^b \tag{2c}$$
$$y_4 = p^n \tag{2d}$$
$$y_5 = v^n \tag{2e}$$

where y_1 is the measurement from the altimeter, y_2 the body-fixed velocity from machine vision, y_3 the specific force from the IMU, y_4 the NED position from the GNSS, and y_5 the NED velocity from the GNSS. Although not explicitly defined in the equations, all sensors are assumed to be affected by Gaussian white noise, which is nevertheless considered when designing and tuning the XKF. The presence of the cross-product term in y_3 is justified in [13]. The OF velocity bias b_v is not included as a state, but is obtained as described in Sec. IV. The GNSS is used in an initial phase to let the estimators converge, after which the GNSS is disabled, y_4 and y_5 are removed from the output map, all biases are frozen at their last known value and not estimated any more, and the only available position fix is on the Down component p_D^n from the altimeter, such that dead reckoning becomes necessary and the North and East components have to be calculated by direct integration of the estimated linear velocity.

III. Optical Flow

The method to obtain the optical flow velocity from camera images was already presented in [9], only a summary is reported here.

There exist several methods for computing the OF. The ones chosen for the experiment presented in Section VI are SIFT [21] and region matching [22], for they can jointly provide a sufficient amount of reliable OF vectors. For the OF computations to be useful in the estimators, a transformation to body-fixed velocity is necessary. The transformation is motivated by [23], and the pinhole camera model is used [24]. The camera-fixed coordinate system is related to the body-fixed coordinate system as illustrated in Fig. 1, where O^{rs} is the centre of the image plane, and O^b the origin of the $\{\mathrm{B}\}$. The optical axis of

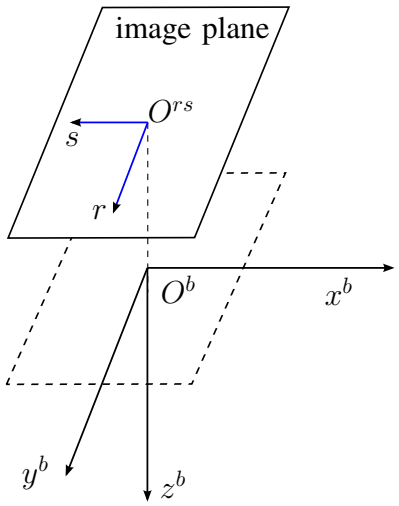

Fig. 1. Pinhole camera model. The camera is oriented downwards, while x^b is the direction of flight. O^{rs} and O^b are the origins of the image plane and body frame, respectively.

the downward-looking camera is aligned with the body z-axis, and the focal point of the camera is assumed to coincide with the origin of $\{\mathrm{B}\}$.

All features tracked by the camera are assumed to be stationary with respect to $\{\mathrm{N}\}$, hence the UAV's linear and angular velocities, v_m^b and ω_F^b (the underscript $_F$ is used to distinguish it from the ω_m^b measured by the rate gyros), relative to a feature tracked by the OF algorithm, will be equal for every tracked feature at a given instant in time. Furthermore, it is assumed that the terrain is flat, such that every feature is located at the same altitude: this assumption can be restrictive for applications that cover an uneven surface, but it can be satisfactory for localized flights, depending on the elevation profile over which the UAV operates; it does, however, greatly simplify the analysis and calculations, such that it is worth to keep it and evaluate the performance in experiments, in Sec. VI.

The angular and linear velocities can be computed by

$$\begin{bmatrix} v_m^b \\ \omega_F^b \end{bmatrix} = -M^+(l, r_{1..k}, s_{1..k}, \phi, \theta, c_z^n) \begin{bmatrix} \dot{r}_1 \\ \dot{s}_1 \\ \vdots \\ \dot{r}_k \\ \dot{s}_k \end{bmatrix} \tag{3}$$

where each pair $[r_j, s_j]$, $j = 1...k$, represents an

image feature, $[\dot{r}_j, \dot{s}_j]$ its corresponding OF, l is the focal length of the camera, c_z^n is the distance of O^b from the ground, and $M \in \mathbb{R}^{2k\times 6}$ is a rectangular matrix. The matrix M is built by stacking k sub-matrices M_j, $j = 1...k$ [9], where

$$M_j = \frac{l}{z_j^b}\begin{bmatrix} 0 & 1 & -\frac{y_j^b}{z_j^b} & -\frac{{y_j^b}^2}{z_j^b} - z_j^b & \frac{y_j^b x_j^b}{z_j^b} & x_j^b \\ -1 & 0 & \frac{x_j^b}{z_j^b} & \frac{x_j^b y_j^b}{z_j^b} & -\frac{{x_j^b}^2}{z_j^b} - z_j^b & y_j^b \end{bmatrix} \tag{4}$$

whose elements depend on

$$\begin{bmatrix} x_j^b \\ y_j^b \\ z_j^b \end{bmatrix} = \begin{bmatrix} \frac{s_j c_z^n}{s_j \sin(\theta) + \cos(\theta)(l\cos(\phi) + r_j \sin(\phi))} \\ -\frac{r_j c_z^n}{s_j \sin(\theta) + \cos(\theta)(l\cos(\phi) + r_j \sin(\phi))} \\ -\frac{f c_z^n}{s_j \sin(\theta) + \cos(\theta)(l\cos(\phi) + r_j \sin(\phi))} \end{bmatrix} \tag{5}$$

Each point $[x_j^b, y_j^b, z_j^b]^T$ is the representation in body-fixed coordinates of the corresponding point from the world. The solution to (3) exists only if $M^T M$ has full rank, such that it can be expressed as $M^+ = (M^T M)^{-1} M^T$. This can only happen if the number of flow vectors are greater than or equal to three. Since rate gyros are part of the sensor payload and can provide measurements of ω^b more accurately than the OF method, only v_m^b will be considered from now on.

This method can calculate the body-fixed velocity of a vehicle, but presents additional challenges. The matrix M often results ill-conditioned and its pseudo-inversion produces computational errors that will be included in v_m^b. Moreover, the accuracy of the computation of OF vectors is strongly dependent on the frame rate of the camera used: the higher the frame rate, the more accurate the results. For the present work, the maximum frame rate was limited to 10 because of the need to have good synchronization between all the sensors. This proved to be not high enough for accurate results and led to a large bias b_v in the measured body-fixed velocity, which the estimators presented next must estimate.

IV. Non-linear Observers

The NLO considered when GNSS is active is an extension of the one in [9], described by

$$\dot{\hat{R}} = \hat{R}S(\omega_m^b - \hat{b}_\omega - \hat{R}^T\omega_e^n) + \sigma K_P \hat{J} \tag{6a}$$

$$\dot{\hat{b}}_\omega = \text{Proj}(\hat{b}_\omega, -k_I \text{vex}(\mathbb{P}_a(\hat{R}_s^T K_P \hat{J}))) \tag{6b}$$

$$\dot{\hat{p}}^n = \hat{v}^n + K_{pp}(p^n - \hat{p}^n) + K_{pv}(v^n - \hat{v}^n) \tag{7a}$$

$$\dot{\hat{v}}^n = \hat{f}^n + g^n + K_{vp}(p^n - \hat{p}^n) + K_{vv}(v^n - \hat{v}^n) - 2S(\omega_e^n)v^n \tag{7b}$$

$$\dot{\xi} = -\sigma K_P \hat{J}(f_m^b - \hat{b}_f) + K_{\xi p}(p^n - \hat{p}^n) \tag{7c}$$

$$+ K_{\xi v}(v^n - \hat{v}^n) \tag{7d}$$

$$\hat{f}^n = \hat{R}(f_m^b - \hat{b}_f) + \xi \tag{7e}$$

$$\dot{\hat{b}}_{f1} = K_f \hat{R}^T(p^n - \hat{p}^n) \tag{7f}$$

$$\hat{b}_f = b_{f0} + \hat{b}_{f1} \tag{7g}$$

where the injection term $\hat{J}$ is

$$\hat{J} := \hat{A}_n A_b^T - \hat{R} A_b A_b^T \tag{8a}$$

$$A_b := \begin{bmatrix} \frac{\hat{f}^b}{\|\hat{f}^b\|}, & \frac{\hat{f}^b \times \hat{v}^b}{\|\hat{f}^b \times \hat{v}^b\|}, & \frac{\hat{f}^b \times (\hat{f}^b \times \hat{v}^b)}{\|\hat{f}^b \times (\hat{f}^b \times \hat{v}^b)\|} \end{bmatrix} \tag{8b}$$

$$\hat{A}_n := \begin{bmatrix} \frac{\hat{f}^n}{\|\hat{f}^n\|}, & \frac{\hat{f}^n \times \hat{v}^n}{\|\hat{f}^n \times \hat{v}^n\|}, & \frac{\hat{f}^n \times (\hat{f}^n \times \hat{v}^n)}{\|\hat{f}^n \times (\hat{f}^n \times \hat{v}^n)\|} \end{bmatrix} \tag{8c}$$

with the vectors $\hat{f}^b = f_m^b - \hat{b}_f$ and $\hat{v}^b = v_m^b - \hat{b}_v$. K_P is a symmetric positive definite gain matrix, $\hat{R}_s = \text{sat}(\hat{R})$, $\sigma \geq 1$ is a scaling factor tuned to achieve stability, $k_I > 0$ is a scalar gain, $\text{Proj}(\cdot,\cdot)$ represents a parameter projection that ensures that $\|\hat{b}_\omega\|$ not exceed a design constant $L_{\hat{b}} > L_b$ (see Appendix). The estimator for the accelerometer bias $\hat{b}_f$ is (7f)–(7g) and is taken from [25]: b_{f0} is a static value for the bias obtained by reading the measurements of the accelerometers at rest, and $\dot{\hat{b}}_{f1}$ represents how the bias varies around this value. The OF velocity bias $\hat{b}_v$ is obtained by calculating $v^n - \hat{R}v_m^b$ and filtering it with a low-pass filter. It could also be added to the NLO as a state and estimated, but it is not considered in the present work.

When the GNSS is disabled and dead reckoning begins, some parts of the observer change slightly

and become

$$\dot{\hat{p}}^n = \hat{v}^n + K'_{pp}(p_D^n - \hat{p}_D^n) \tag{9a}$$

$$\dot{\hat{v}}^n = \hat{f}^n + g^n + K'_{vp}(p_D^n - \hat{p}_D^n) + K'_{vv}(\hat{R}(v_m^b - \hat{b}_v) - \hat{v}^n) - 2S(\omega_e^n)\hat{v}^n \tag{9b}$$

$$\dot{\xi} = -\sigma K_P \hat{J}(f_m^b - \hat{b}_f) + K'_{\xi pR}(p_D^n - \hat{p}_D^n) \tag{9c}$$

$$+ K'_{\xi v}(\hat{R}(v_m^b - \hat{b}_v) - \hat{v}^n) \tag{9d}$$

$$\hat{f}^n = \hat{R}(f_m^b - \hat{b}_f) + \xi \tag{9e}$$

The three biases are not estimated any more, since the reduced set of measurements is not sufficient to make them observable. Consequently, the values used during dead reckoning are constant and correspond to the last values estimated when GNSS was available. Only the forward and lateral components of v_m^b are used in (9), since the vertical component is typically significantly less accurate than the other two. This also explains why (9a) has no velocity correction term.

A. Assumptions

Three conditions are sufficient to ensure a correct functioning of both versions of the NLO.

Assumption 1: the machine vision system is able to provide v_m^b.

Assumption 2: the gyro biases b_ω, b_f, and b_v are constant.

Assumption 3: there exists a constant $c_{obs} > 0$ such that, $\forall t \geq 0,\ \|f^b \times v^b\| \geq c_{obs}$.

Assumption 1 can be satisfied by implementing one of the many methods to obtain body-fixed velocity from camera images, for example [9]. This assumption is also necessary. Assumption 2 can be relaxed to slowly time-varying, in particular when GNSS is available. Assumption 3 is a condition of non-collinearity for the vectors v^b and f^b, i.e. the angle between them is non-zero and none of them can be identically zero (see, e.g., [26], [5]). For a fixed-wing UAV this means that the observer cannot work while the vehicle stands still on the ground, but presents no problems during flight.

As the focus of this paper is the experimental application of the proposed estimators, proofs of stability are not presented here. The NLOs, however, are built from [9], [19] by adding the biases and the compensation for the rotation of Earth, about which a few arguments can be advanced. The accelerometer bias is estimated in a way that was already proved to be GES [20], [25], so it certainly improves the NLO. The velocity bias depends on $\hat{R}$: the NLO with uncorrected body-fixed velocity could be let converge to avoid possible instabilites caused by uncertain initialization of $\hat{R}$, and only afterwards include the bias for improved performance. The same can be said about the compensation for Earth's rotation, which depends on $\hat{v}^n$ and $\hat{R}$. Both NLOs are therefore guaranteed to converge (clearly with the exception of p_N^n and p_E^n in dead reckoning).

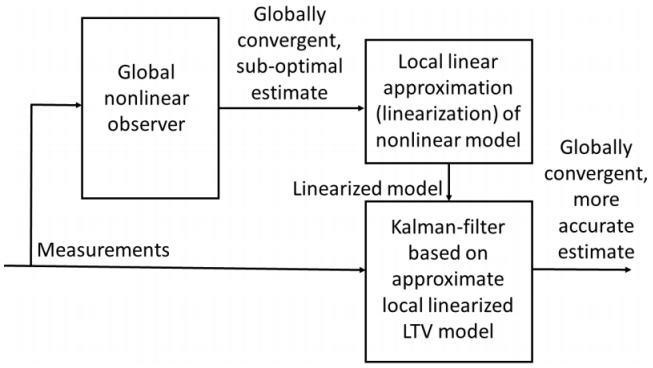

Fig. 2. Block diagram showing the cascade interconnection between an NLO and an LKF, compactly called XKF [12].

V. eXOGENOUS KALMAN FILTER

The XKF is a novel tool that combines the advantages of non-linear observers and Kalman filters while overcoming their individual shortcomings [12]. The first-stage filter of the XKF is either of the NLOs described in Section IV, depending on whether GNSS is available or not, whereas the second-stage filter is a linear time-varying Kalman filter (LKF) based on a linear approximation of the state and output equations about the trajectory generated by the NLO. In order to reduce the number of states, the kinematic system considered for this second-stage filter is represented differently from (1), as

$$\dot{\Theta} = T(\Theta)(\omega_m^b - b_\omega + n_\omega) \tag{10a}$$

$$\dot{p}^n = v^n \tag{10b}$$

$$\dot{v}^n = R(f_m^b - b_f + n_f) + g^n - 2S(\omega_e^n)v^n \tag{10c}$$

The attitude is now represented by the vector of Euler angles $\Theta = [\phi, \theta, \psi]^T$ with its dynamic equation, where $T(\Theta) \in \mathbb{R}^{3\times3}$ is the state-dependent transformation matrix [20]. n_ω and n_f are noise on rate gyros and accelerometers, respectively. The output map is the same as (2), with Gaussian white noise added to all measurements. A block diagram of a generic XKF is represented in Fig. 2. The steps necessary to obtain the XKF equations are omitted, as they were already presented in [12], [13].

The following assumptions are standard conditions that ensure that the XKF inherits the GES property of the NLO [27], [28].

Assumption 4: The LKF tunable parameters (process noise covariance matrix, measurement noise covariance matrix, and initial estimation error covariance matrix) are positive definite and symmetric.

Assumption 5: The system (10), (2) linearized about the trajectory generated by the NLO is uniformly completely observable and controllable.

Assumption 4 can be satisfied by design. The requirements of Assumption 5 are hard to verify analytically *a priori*. However, it is possible to monitor the LKF covariance matrix, and since it is always bounded it can be inferred that Assumption 5 is satisfied.

VI. Experimental Setup

The UAV employed is a UAV Factory Penguin-B, equipped with a custom payload that includes all the necessary sensors. The IMU is a Sensonor STIM300, a low-weight, tactical grade, high-performance sensor that includes gyroscopes, accelerometers, and inclinometers, all recorded at a frequency of 300 Hz. The chosen GPS receiver is a uBlox LEA-6T, which gives measurements at 5 Hz. The video camera is an IDS GigE uEye 5250CP provided with an 8mm lens. The camera is configured for a hardware-triggered capture at 10 Hz: the uBlox sends a digital pulse-per-second signal whose rising edge is accurately synchronized with the time of validity of the recorded GPS position, which guarantees that the image capture is synchronized with the position measurements. The experiment was carried out on 6 February 2015 at the Eggemoen Aviation and Technology Park, Norway, in a sunny day with good visibility, very little wind, an air temperature of about -8°C. The terrain is covered with snow and flat enough to let all features be considered as lying at zero altitude relative to altitude measurements.

The NLOs are implemented using forward Euler discretization with a time-varying step depending on the interval of data acquisition of the fastest sensor, namely the STIM300, and it is typically around 0.003 seconds. The various parameters and gains are chosen as $L_{b^b} = 2^\circ/\text{s}$, $L_{\hat{b}^b} = 2.1^\circ/\text{s}$, $\sigma = 1$, $K_P = \text{diag}[0.08, 0.04, 0.06]$, $k_I = 0.02$, $K_{pp} = 30I_3$, $K_{pv} = 2I_3$, $K_{vp} = 0.01I_3$, $K_{vv} = 20I_3$, $K_{\xi p} = I_3$, and $K_{\xi v} = 50I_3$. During dead reckoning some gains are different, namely $K'_{pp} = \text{diag}(0, 0, 300)$, $K'_{vp} = (0, 0, 200)^T$, $K'_{vv} = \text{diag}(30, 30, 0)$, $K'_{\xi p} = (0, 0, 200)^T$, and $K'_{\xi v} = \text{diag}(100, 100, 0)$. The gains for the NLO with GNSS are obtained by running the observer several times and correcting the gains until a satisfactory performance was achieved. For dead reckoning, the gain relative to the altimeter is high because it is a very accurate sensor. The OF velocity is less accurate than the GNSS velocity for the reasons explained in Sec. III, so it would be reasonable to use lower gains for it, but the presence of some residual bias requires to use a higher gain instead. This provides some additional filtering of the OF velocity, at the cost of worse noise rejection in the estimated position and velocity. The covariance matrices for the XKF are first tuned based on previous experience with the same sensors and system model, and then more finely tuned with trial and error. For both the NLO and XKF, the gyro bias and accelerometer bias estimates are initialized with the standstill values, the other states with zero.

The reference provided for the position, velocity, and attitude is the output of the EKF of the autopilot mounted on the Penguin-B; the autopilot uses a different set of sensors than the one presented here. References for the gyro bias and accelerometer bias are not available, but approximations of the real values are calculated

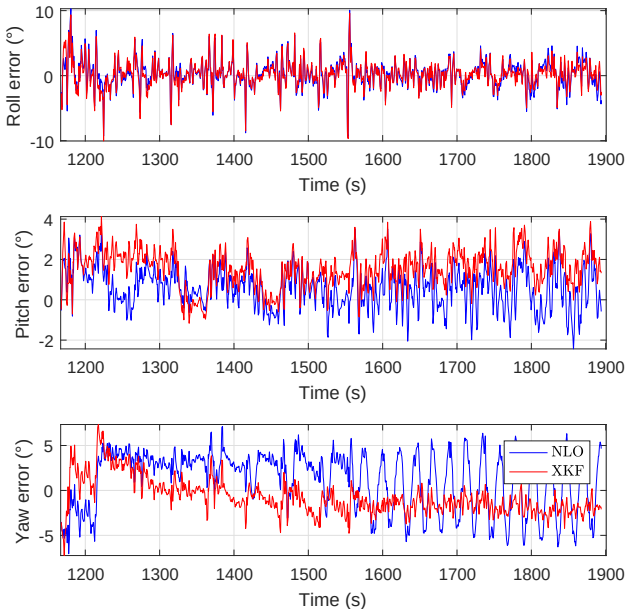

Fig. 3. Euler angles estimation error of the two methods with respect to the autopilot EKF.

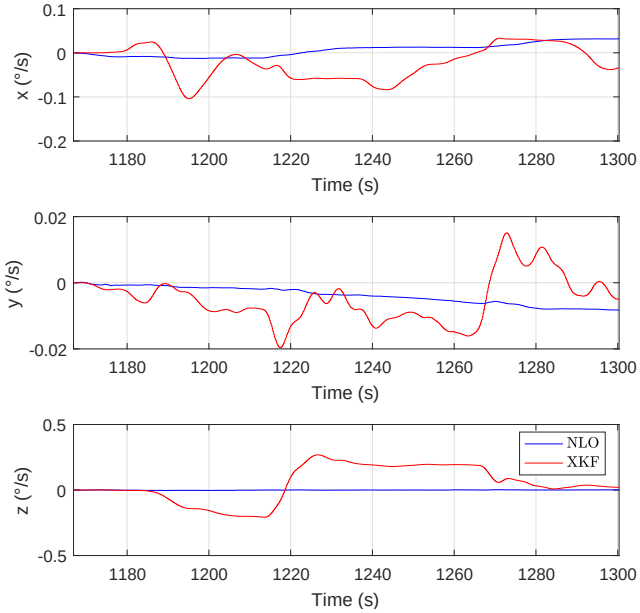

Fig. 4. Gyro bias estimation error with respect to the value measured at standstill. The estimates are used to pre-condition the measured angular velocity ω_m^b.

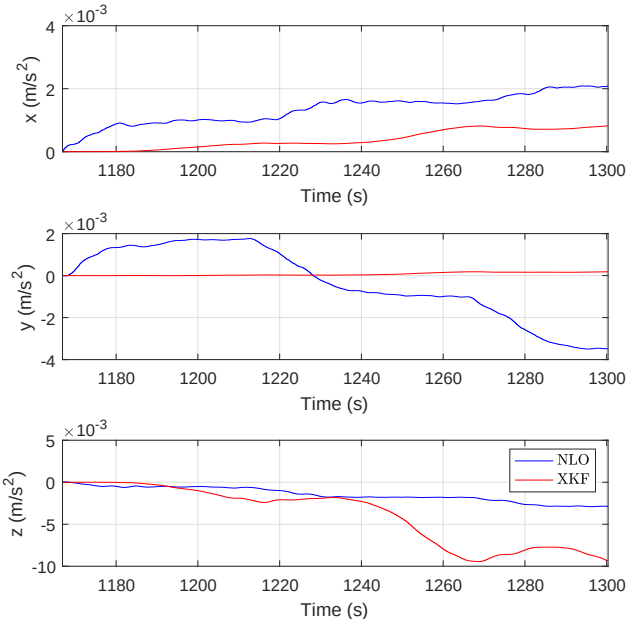

Fig. 5. Accelerometer bias estimation error with respect to the value measured at standstill. The estimates are used to pre-condition the measured specific force f_m^b.

by averaging the measurements of the rate gyros and accelerometers at standstill before and after the flight. It is not possible to obtain even an approximation of a reference for the OF velocity bias, hence the relative results show the actual estimates, not the estimation error.

The time on the x-axes of the figures is the time elapsed since the computer onboard the UAV was turned on. For all tests, the GNSS is used by the estimators until $t = 1300$ s, after which it is dropped and the estimators run in dead reckoning, with all biases frozen at their last estimated value.

A. *Results*

The results are presented in Fig. 3–9. The yaw angle estimated by the XKF is more stable than the one from the NLO; the pitch angle from the XKF seems less accurate, but the difference between NLO and XKF is between $0°-2°$, so it is legit to wonder whether the reference value is really the closest to the real value or not. The gyro, accelerometer, and OF velocity biases are in Fig. 4–6. The plots end at $t = 1300$ s because at that point dead reckoning begins and the biases are frozen at their last estimated values. The XKF provides slightly better filtering for the OF velocity bias, but it is evident that there are inconsistencies between the outputs of the two estimators for the gyro and accelerometer biases. This is most probably owing to inaccurate tuning, and a better tuning is expected to lead to better bias estimates that would reflect in even better position and velocity estimates. In Fig. 7 the position errors are rep-

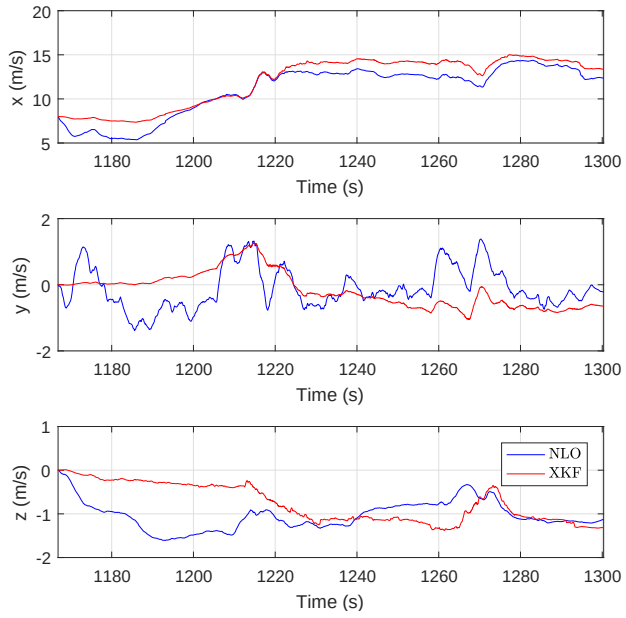

Fig. 6. OF velocity bias. This is not a plot of the estimation error, for a reference is not available. The estimates are used to pre-condition the measured OF velocity v_m^b.

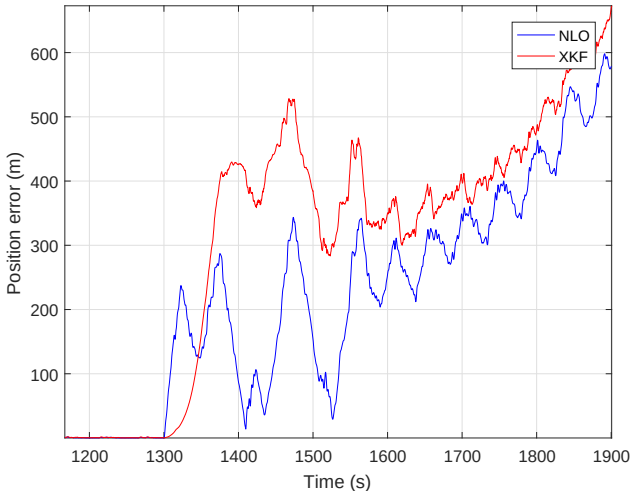

Fig. 7. Position estimation error of the two methods compared with the one from the autopilot EKF.

resented. The NLO error shows an improvement with respect to the results in [19] from the same authors, although the oscillations are still present: these are caused by the bias in the body-fixed velocity combined with the circular motion of the UAV, such that when the error increases in one direction, it decreases in the opposite direction. Since the estimated bias is kept constant during dead reckoning, a residual bias will still be present and affect the results, but evidently to a lesser

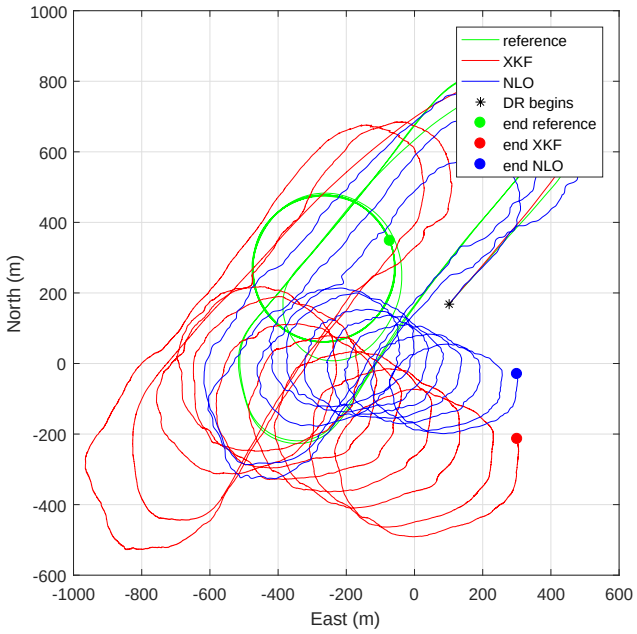

Fig. 8. Trajectories on the North-East plane estimated by the NLO and XKF compared with the one from the autopilot EKF.

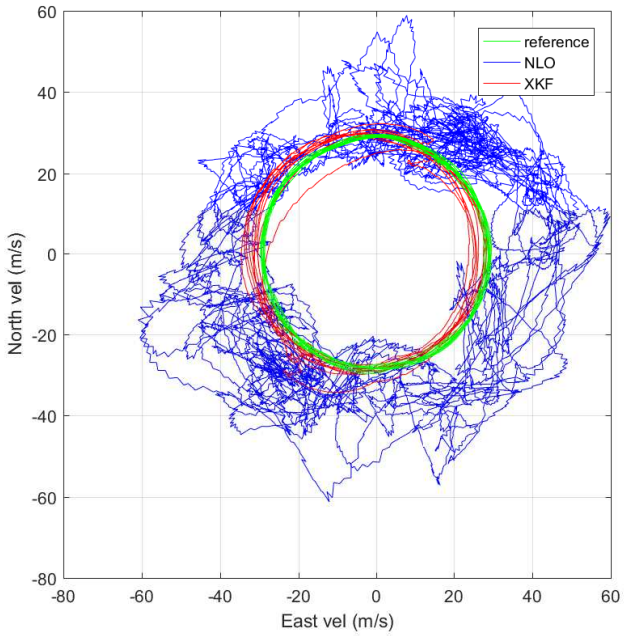

Fig. 9. Velocities on the North-East plane estimated by the NLO and XKF compared with the one from the autopilot EKF.

degree than in [19]. The XKF position error, on the other hand, presents significantly smaller variations and is perhaps more suitable than the NLO in a dead reckoning scenario: if the UAV flew in a straight line instead of following circular pattern, the position error of the NLO would be expected

to increase faster than the one of the XKF. The differences in position error are reflected in the differences in velocity error of Fig. 9: the velocity estimated by the NLO is more erratic, more noisy than the one from the XKF, which in turn seems to manifest just a sistematic error that explains the lack of oscillations in the position error.

VII. Conclusions

In this paper an NLO and an XKF for dead reckoning have been tested on experimental data obtained with a UAV equipped with an custom payload of sensors. A machine vision system has been employed to calculate the body-fixed linear velocity of the UAV using optical flow. This velocity has been used both as a vector for the injection term of the NLO, and as a correction term, combined with the estimated attitude, for the estimated NED velocity. In comparison to previous work, estimators for accelerometer bias and body-fixed velocity bias have been proposed, together with a term that compensates for the rotation of Earth. The horizontal position was not estimated, but calculated via direct integration of the estimated velocity. The results show that the inclusion of compensation for the additional biases and for the rotation of Earth help reduce the position error of the NLO, and that the XKF can reduce the error even further by providing a better estimate of the velocity that is less dependent than the NLO on the optical flow velocity bias.

Acknowledgements

This work has been carried out at the NTNU Centre for Autonomous Marine Operations and Systems (NTNU AMOS), supported by the Research Council of Norway, grants no. 223254 and 221666. The authors are grateful for the assistance provided by the UAV engineers at NTNU and Maritime Robotics AS, in particular Lars Semb and Carl Erik Stephansen. Significant contributions were made to the construction of the UAV payload by the rest of the navigation team at NTNU, in particular Sigurd M. Albrektsen, Jakob M. Hansen and Kasper T. Borup, and to the development of machine vision by Jesper Hosen and Håkon H. Helgesen.

References

[1] M. D. Shuster and S. D. Oh, "Three-axis attitude determination from vector observations," *Journal of Guidance, Control and Dynamics*, vol. 4, no. 1, pp. 70–77, 1981.

[2] E. Lefferts, F. Markley, and M. Shuster, "Kalman filtering for spacecraft attitude," *Journal of Guidance Control and Dynamics*, vol. 5, no. 5, pp. 417–429, 1982.

[3] J. L. Crassidis, F. L. Markley, and Y. Cheng, "Survey of nonlinear attitude estimation methods," *Journal of guidance, control, and dynamics*, vol. 30, no. 1, pp. 12–28, 2007.

[4] S. Salcudean, "A globally convergent angular velocity observer for rigid body motion," *IEEE Transactions on Automatic Control*, vol. 36, no. 12, pp. 1493–1497, 1991.

[5] R. Mahony, T. Hamel, and J. M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, 2008.

[6] P. Batista, C. Silvestre, and P. Oliveira, "A GES attitude observer with single vector observations," *Automatica*, vol. 48, no. 2, pp. 388–395, 2012.

[7] M. D. Hua, P. Martin, and T. Hamel, "Stability analysis of velocity-aided attitude observers for accelerated vehicles," *Automatica*, vol. 63, pp. 11–15, 2016.

[8] H. F. Grip, T. I. Fossen, T. A. Johansen, and A. Saberi, "Globally exponentially stable attitude and gyro bias estimation with application to GNSS/INS integration," *Automatica*, vol. 51, pp. 158–166, 2015.

[9] L. Fusini, T. A. Johansen, and T. I. Fossen, "Experimental validation of a uniformly semi-globally exponentially stable non-linear observer for GNSS- and camera-aided inertial navigation for fixed-wing UAVs," *Proc. Int. Conf. Unmanned Aircraft Syst.*, 2015, doi: 10.1109/ICUAS.2015.7152371.

[10] P. Martin, I. Sarras, M. Hua, and T. Hamel, "A global exponential observer for velocity-aided attitude estimation," *Proc. IEEE Conf. Decision and Control*, pp. 2047–2053, 2016.

[11] T. Johansen and T. Fossen, "Nonlinear filtering with exogenous Kalman filter and double Kalman filter," *Proc. European Control Conf.*, 2016.

[12] ——, "The eXogenous Kalman filter (XKF)," *Int. J. Control*, vol. 60, pp. 161–167, 2017, doi: 10.1080/00207179.2016.1172390.

[13] L. Fusini, T. I. Fossen, and T. A. Johansen, "Nonlinear camera- and GNSS-aided INS for fixed-wing UAV using the eXogenous Kalman Filter," in *Sensing and Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles*. Springer (T. I. Fossen, K. Y. Pettersen and H. Nijmeijer, Eds.), 2017, ch. 1.2.

[14] D. Scaramuzza and F. Fraundorfer, "Visual odometry – part i: The first 30 years and fundamentals," *IEEE Robotics & Automation Magazine*, vol. 18, no. 4, pp. 80–92, 2011.

[15] J. Zhang and S. Singh, "INS assisted monocular visual odometry for aerial vehicles," *9th International Conference on Field and Service Robotics (FSR)*, 2013.

[16] S. Zhao, F. Lin, K. Peng, X. Dong, B. M. Chen, and T. H. Lee, "Vision-aided estimation of attitude, velocity, and inertial measurement bias for UAV stabilization," *Journal of Intelligent and Robotic Systems*, 2015.

[17] D. Dusha and L. Mejias, "Error analysis and attitude observability of a monocular GPS/visual odometry integrated navigation filter," *International Journal of Robotics Research*, vol. 31, no. 6, pp. 714–737, 2012.

[18] M. Mammarella, G. Campa, M. L. Fravolini, and M. R. Napolitano, "Comparing optical flow algorithms using 6-dof motion of real-world rigid objects," *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 1752 – 1762, 2012.

[19] L. Fusini, T. A. Johansen, and T. I. Fossen, "A globally exponentially stable non-linear velocity observer for vision-aided UAV dead reckoning," *Proc. 2016 IEEE Aerospace Conf.*, 2016, doi: 10.1109/AERO.2016.7500606.

[20] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Chichester, UK: John Wiley & Sons, Ltd, 2011.

[21] D. G. Lowe, "Object recognition from local scale-invariant features," *Proc. Int. Conf. Computer Vision*, pp. 1150–1157, 1999.

[22] C. S. Fuh and P. Maragos, "Region-based optical flow estimation," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Proceedings CVPR.*, pp. 130–135, 1989.

[23] L. Fusini, T. I. Fossen, and T. A. Johansen, "A uniformly semiglobally exponentially stable nonlinear observer for GNSS- and camera-aided inertial navigation," *22nd Mediterranean Conf. Control and Automation*, pp. 1031–1036, 2014, doi: 10.1109/MED.2014.6961510.

[24] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, 1996.

[25] R. Rogne, T. Bryne, T. Fossen, and T. Johansen, "MEMS-based inertial navigation on dynamically positioned ships: Dead reckoning," *IFAC Conference on Control Applications in Marine Systems*, pp. 139–146, 2016.

[26] M. D. Hua, "Attitude estimation for accelerated vehicles using GPS/INS measurements," *Control Engineering Practice*, vol. 18, no. 7, pp. 723–732, 2010.

[27] R. Kalman and R. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME, Ser. D, J. Basic Eng.*, pp. 95–109, 1961.

[28] B. Anderson, "Stability properties of Kalman-Bucy filters," *J. Franklin Institute*, vol. 291, pp. 137–144, 1971.

[29] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*. New York: Wiley, 1995.

APPENDIX

The parameter projection Proj$(\cdot,\cdot)$ is defined as:

$$\text{Proj}(\hat{b}^b,\tau)=\begin{cases}\left(I-\frac{c(\hat{b}^b)}{\|\hat{b}^b\|^2}\hat{b}^b\hat{b}^{bT}\right)\tau, & \|\hat{b}^b\|\geq L_b,\ \hat{b}^{bT}\tau>0\\ \tau, & \text{otherwise}\end{cases}$$

where $c(\hat{b}^b)=\min\{1,(\|\hat{b}^b\|^2-L_b^2)/(L_{\hat{b}}^2-L_b^2)\}$. This operator is a special case of that from Appendix E of [29].