

Article

Optimization-Based Sensor Fusion of GNSS and IMU Using a Moving Horizon Approach

Fabian Girrbach ^{1,2,*}, Jeroen D. Hol ¹, Giovanni Bellusci ¹ and Moritz Diehl ²

¹ Xsens Technologies B.V., Enschede 7521 PR, The Netherlands; jeroen.hol@xsens.com (J.D.H.); giovanni.bellusci@xsens.com (G.B.)

² Department of Microsystems Engineering (IMTEK), University of Freiburg, 79110 Freiburg, Germany; moritz.diehl@imtek.uni-freiburg.de

* Correspondence: fabian.girrbach@xsens.com; Tel.: +31-88-973-6700

Academic Editors: Cheng Wang, Julian Smit, Ayman F. Habib and Michael Ying Yang

Received: 31 March 2017; Accepted: 10 May 2017; Published: 19 May 2017

Abstract: The rise of autonomous systems operating close to humans imposes new challenges in terms of robustness and precision on the estimation and control algorithms. Approaches based on nonlinear optimization, such as moving horizon estimation, have been shown to improve the accuracy of the estimated solution compared to traditional filter techniques. This paper introduces an optimization-based framework for multi-sensor fusion following a moving horizon scheme. The framework is applied to the often occurring estimation problem of motion tracking by fusing measurements of a global navigation satellite system receiver and an inertial measurement unit. The resulting algorithm is used to estimate position, velocity, and orientation of a maneuvering airplane and is evaluated against an accurate reference trajectory. A detailed study of the influence of the horizon length on the quality of the solution is presented and evaluated against filter-like and batch solutions of the problem. The versatile configuration possibilities of the framework are finally used to analyze the estimated solutions at different evaluation times exposing a nearly linear behavior of the sensor fusion problem.

Keywords: multi-sensor fusion; state estimation; moving horizon estimation; nonlinear optimization; inertial navigation

1. Introduction

During the last few years, fully autonomous systems have been a highly active research field, which pushed product development towards the commercialization of such systems. For applications such as autonomous driving and *unmanned aerial vehicles* (UAVs) more semi-autonomous features become available every day. Besides the future of self-driving cars, autonomous drones will soon take over tasks in transportation, agriculture, maintenance, surveillance or energy generation. The first prototypes have already proven to be feasible and have successfully delivered small sized goods, simplified inspection processes in rough terrain or harvested wind energy at previously unreachable altitudes. These recent developments allow the prediction of an increasing number of UAVs applications (see Figure 1), which will result in a more crowded airspace, representing a paradigm shift in comparison to traditional airborne applications where the airspace is heavily secured and supervised.

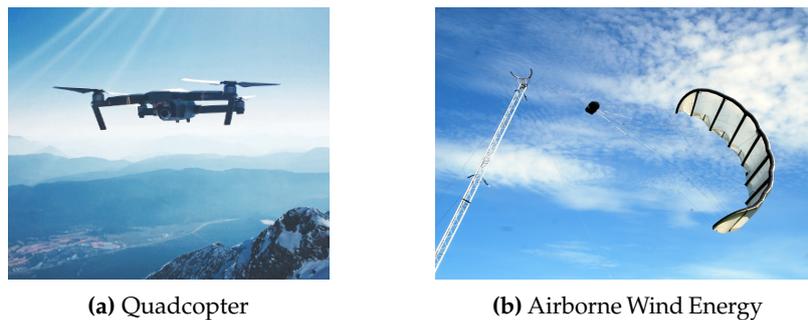


Figure 1. Examples for autonomous airborne applications: (a) shows a flying drone (available at <https://unsplash.com/photos/ZlkRrzJl20Q>, published under CC0, photo by imKirk); (b) shows a prototype of an airborne wind energy system developed at TU Delft (<http://www.kitepower.eu>). Courtesy of Roland Schmehl, TU Delft.

UAVs operating in a more occupied airspace close to humans require increased robustness to avoid fatal incidents, which translates to a strict set of requirements imposing new challenges for the estimation and control algorithms. *Moving horizon estimation* (MHE) and *model predictive control* (MPC) are promising strategies that use numerical optimization methods over a window of data to increase stability and accuracy of the system's motion. MPC has been successfully applied to challenging control problems with nonlinear dynamics and difficult environment conditions [1–3]. The counterpart to the optimal control theory for state estimation is represented by MHE which has already been used for decades to estimate the state of nonlinear chemical processes [4] or more recently, to increase fault tolerance of relative navigation problems [5]. The computational burden of solving an optimization problem has become less restrictive due to advances in computer technology and the development of tailored optimization algorithms [6,7] that nowadays allow the usage of efficient optimization-based methods on embedded systems. These recent developments make these strategies attractive for a broad range of applications including autonomous systems.

To estimate the motion of a system exposing a high degree of freedom with high accuracy, it often becomes necessary to fuse information from different sensors. Sensor fusion is a well known strategy to reduce the impact of measurement errors on the state estimate [8] and estimate not directly observable system states. One traditional way to solve multi-sensor fusion problems for time critical applications is the famous *Kalman filter* (KF) algorithm [9] and its derivatives for nonlinear systems *extended Kalman filter* (EKF) and *unscented Kalman filter* (UKF). To achieve high estimation accuracy, KF-based algorithms require, in practice, a procedure to tune the filter parameters, which is, for complex systems, non-trivial yet crucial. To overcome this problem, adaptive methods have been proposed [10,11]; however, filter stability can become an issue under certain conditions. Optimization-based approaches allow for a more elegant formulation respecting system constraints [12]. Typical Newton-type optimization algorithms use an iterative approach and are therefore known to better capture the nonlinearity of the problem yielding a more accurate and robust solution.

In this paper, we present an optimization-based sensor fusion framework for state estimation following an MHE approach, extending [13]. The framework is applied to the well-known sensor fusion problem for inertial navigation of a *global navigation satellite system* (GNSS) receiver measuring position and an *inertial measurement unit* (IMU) measuring linear acceleration and angular velocity. The measurement data is used to estimate the 3D-pose and velocity of a maneuvering object, such as an aircraft or satellite. There are several methods to integrate the inertial data and GNSS, such as loosely coupled and tightly coupled integration. The integration is typically achieved by adapting a nonlinear KF [14–16]. By preprocessing the nonlinearities of the problem, linear KFs represent a further option for the sensor fusion problem [17].

The sensor fusion problem discussed in this paper arises from the use of consumer-grade IMUs for which sensor errors introduce a drift that needs to be compensated for by using different measurement updates such as those given by vision based sensors [18] or GNSS receivers. The resulting sensor fusion problem for inertial navigation has been addressed using traditional and advanced approaches for Kalman filtering [17,19]. The observability of sensor parameters, such as biases of the IMU, is strongly coupled to the motions of the system. Therefore, a unique solution to the estimation problem cannot be guaranteed [20]. By considering a time window of measurement data, the estimated solution is suggested to achieve improved accuracy and robustness [21]. Several MHE formulations for sensor fusion in the context of inertial navigation have been published in the recent past and have been shown to outperform traditional EKF approaches for the integration of GNSS and IMU [22,23] and online-identification of IMU parameters [24]. Despite the different sets of sensors, models and optimization methods, all authors present an accuracy improvement for a specific horizon length compared to Kalman filtering. This paper contributes to the existing research by presenting a general sensor fusion framework capable of analyzing moving horizon estimators over the spectrum of filter-like configurations to batch processing. Furthermore, the estimated results are evaluated against an accurate reference of a maneuvering aircraft that was obtained using high-grade sensors. A detailed study of the influence of the horizon length on the quality of the solution is given and critically analyzed.

This paper is organized as follows. First, in Section 2, we describe the relevant system and sensor models to address the sensor fusion problem by the definition of an equality-constrained optimization problem, which components are explained in detail. In Section 3, the presented approach is used to estimate the trajectory of an aircraft, and the results are compared against an accurate reference trajectory. Finally, the conclusions of this work are presented in Section 4.

2. Methods

Section 2 introduces the necessary coordinate frames to define the system and sensor models, which will finally be used to derive the optimization problem of the MHE state estimator.

2.1. Coordinate Frames

The sensor fusion problem contains measured and estimated quantities expressed in several coordinate frames (see Figure 2). The position measurements are obtained by the GNSS sensor in the *earth-centered, earth-fixed* (ECEF) frame and often expressed in *latitude, longitude, altitude* (LLA). The measurements are transformed to a locally-fixed and non-moving frame L following the *east, north, up* (ENU) convention with its origin located at a reference location. Since the transformation between global and local frame is constant over time, the measurements are converted to the local frame in a preprocessing step. The measurements of the IMU are obtained in the sensor coordinate frame S , which is moving with respect to the local frame L . Throughout this document, the notation \cdot^L or \cdot^S is adopted to indicate measured or estimated variables according to the local or sensor frame, respectively.

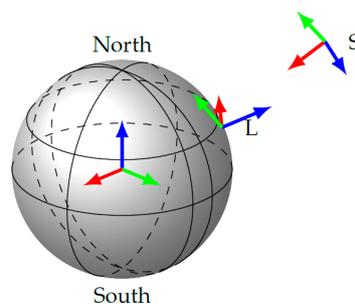


Figure 2. The relevant navigation frames for the fusion of GNSS and IMU. The figure shows the fixed frames (ECEF and L) and the free moving sensor frame S .

2.2. System Model

A generic piecewise constant linear acceleration and angular velocity model are used to model translational and rotational dynamics of the moving system. These models are widely used in the target tracking community and can be reviewed in detail in, e.g., the survey of [25]. The *ordinary differential equations* (ODEs) of the navigational states are defined by

$$\dot{p}^L(t) = v^L(t), \quad (1a)$$

$$\dot{v}^L(t) = a^L(t), \quad (1b)$$

$$\dot{q}^{LS}(t) = \frac{1}{2}q^{LS}(t) \odot \omega_{LS}^S(t), \quad (1c)$$

where the position $p^L(t) \in \mathbb{R}^3$ and velocity $v^L(t) \in \mathbb{R}^3$ are obtained by integration of the acceleration $a^L(t) \in \mathbb{R}^3$. The angular velocity $\omega_{LS}^S \in \mathbb{R}^3$ from the S -frame to the L -frame expressed in the S -frame drives the ODE of the orientation, which is parametrized by a unit quaternion $q^{LS}(t) \in \mathbb{Q}_1 = \{\mathbb{R}^4 : \|q\|_2 = 1\}$ describing the orientation between the S and L -frame. The \odot operator is introduced

$$q \odot r := [-q_v \cdot r, q_0 r + q_v \times r] \quad (2)$$

for the product of a vector $r \in \mathbb{R}^3$ and a quaternion $q = (q_0, q_v) \in \mathbb{Q}_1$ with $q_0 \in \mathbb{R}$ and $q_v \in \mathbb{R}^3$ being the scalar and vector part, respectively. The ODE Equations (1) of the system model lead to the definition of the concatenated vector

$$\mathbf{x}_M(t) = [p^L(t)^T, v^L(t)^T, q^{LS}(t)^T]^T, \quad (3)$$

which contains the navigation states of the model and is therefore denoted by the index M . The navigational states are driven by the piecewise constant control inputs

$$\mathbf{u}(t) = [a^L(t)^T, \omega_{LS}^S(t)^T]^T. \quad (4)$$

2.3. Sensor Models

The sensors used for this estimation problem are a single GNSS receiver and an IMU which are both rigidly attached to the moving system. The described sensor models relate the measured quantities to the state and control variables defined accordingly in Equations (3) and (4). Section 2.2 defines a continuous-time model; however, the sensors acquire the measurements at discrete times t_k with a sensor-specific measurement frequency f_s . We define output functions for each sensor measurement, which evaluate the continuous variables at the sampling times t_k and establish the relation between discrete measurements and estimated continuous state and control variables.

2.3.1. Inertial Measurement Unit

An IMU combines a *three-axis accelerometer* and a *three-axis gyroscope* in a single package. The sensors can be based on different principles that define implicitly the accuracy of the sensor. Due to the advances in *micro-electro-mechanical system* (MEMS) technology, the sensor modules can be produced very cost efficiently on a single silicon chip. The MEMS accelerometer and gyroscope are modeled in this paper using an additive bias term δ^S to compensate the time-varying offsets in the average signal output of the sensor [26]. These biases $\delta_a^S \in \mathbb{R}^3$ and $\delta_\omega^S \in \mathbb{R}^3$ are modeled as a random walk using the device specific Allan variance [27]. To include the estimation of the bias terms, we extend the state vector $\mathbf{x}_M(t)$ as defined in Equation (3) to

$$\mathbf{x}(t) = [\mathbf{x}_M(t)^T, \delta_a^S(t)^T, \delta_\omega^S(t)^T]^T = [p^L(t)^T, v^L(t)^T, q^{LS}(t)^T, \delta_a^S(t)^T, \delta_\omega^S(t)^T]^T. \quad (5)$$

The bias terms are modeled as constants between consecutive sampling times t_k and t_{k+1} of the sensor exposing the lowest sampling rate f_S leading to

$$\delta_a^S(t) = \text{const}, \quad t_k \geq t > t_{k+1}, \quad (6a)$$

$$\delta_\omega^S(t) = \text{const}, \quad t_k \geq t > t_{k+1}, \quad (6b)$$

and the actual random walk is embedded inside the optimization problem by allowing discontinuities according to the device specific random walk process noise.

The inertial quantities acceleration $y_a^S \in \mathbb{R}^3$ and angular velocity $y_\omega^S \in \mathbb{R}^3$ are acquired by the IMU at sampling rate $f_{\text{IMU}} \gg 100$ Hz. The sampled IMU data is integrated between two consecutive measurements of the slowest sensor, which is, in this specific problem, the GNSS receiver, and expressed as motion increments [28]. The increments are converted to an average inertial measurement over the interval $[t_k, t_{k+1}]$, which directly translates to the piecewise constant control inputs defined in Equation (4). The following output functions define the IMU measurements in terms of state and control variables:

$$y_a^S(\mathbf{x}, \mathbf{u}, t_k) = R(q^{LS}(t_k))^{-1}, a^L(t_k) - g^L + \delta_a^S(t_k), \quad (7a)$$

$$y_\omega^S(\mathbf{x}, \mathbf{u}, t_k) = \omega_{LS}^S(t_k) + \delta_\omega^S(t_k), \quad (7b)$$

where g^L stands for the constant gravity vector in the L frame and $R(q, r)$ denotes the rotation of a vector $r \in \mathbb{R}^3$ by a unit quaternion $q \in \mathbb{Q}_1$.

2.3.2. GNSS Receiver

A GNSS receiver is used to retrieve position measurements based on pseudo ranges estimated from satellite signals. The interested reader is referred to [29] for more information. A typical GNSS receiver has an output frequency between 1 Hz and 10 Hz, at which the position measurements are made available with respect to an earth fixed coordinate frame and converted to the local L frame as described before in Section 2.1. The output function for position measurements at t_k is defined by

$$y_p^L(\mathbf{x}, \mathbf{u}, t_k) = p^L(t_k), \quad (8)$$

which relates the GNSS measurements directly to the state $\mathbf{x}(t)$. The position accuracy is estimated and reported by the GNSS receiver and strongly depends on the signal and ambient conditions.

2.4. Optimization Problem

To define the optimization problem for the nonlinear sensor fusion of GNSS and IMU, we first define the components of the cost function, the implicit integration method using direct collocation, and the necessary equality constraints.

2.4.1. Measurement Residuals

The cost function of the optimal estimation problem is defined by the squared weighted sum of residuals between the estimated output variables \mathbf{y}_k and the measurements $\bar{\mathbf{y}}_k$ at sampling times t_k . The cost terms for the position and inertial measurements are defined by

$$c_p(\mathbf{x}, \mathbf{u}, t_k) = \frac{1}{2} \|r_p(\mathbf{x}, \mathbf{u}, t_k)\|_{Q_p^{-1}}^2 = \frac{1}{2} \|y_p^L(\mathbf{x}, \mathbf{u}, t_k) - \bar{y}_{p,k}^L\|_{Q_{p,k}^{-1}}^2, \quad (9a)$$

$$c_a(\mathbf{x}, \mathbf{u}, t_k) = \frac{1}{2} \|r_a(\mathbf{x}, \mathbf{u}, t_k)\|_{Q_a^{-1}}^2 = \frac{1}{2} \|y_a^S(\mathbf{x}, \mathbf{u}, t_k) - \bar{y}_{a,k}^S\|_{Q_a^{-1}}^2, \quad (9b)$$

$$c_\omega(\mathbf{x}, \mathbf{u}, t_k) = \frac{1}{2} \|r_\omega(\mathbf{x}, \mathbf{u}, t_k)\|_{Q_\omega^{-1}}^2 = \frac{1}{2} \|y_\omega^S(\mathbf{x}, \mathbf{u}, t_k) - \bar{y}_{\omega,k}^S\|_{Q_\omega^{-1}}^2, \quad (9c)$$

and evaluate the continuous output functions of IMU (7) and GNSS (8) at time t_k . The residuals are weighted according to the measurement noise variances using the diagonal matrices $Q(\cdot)$.

2.4.2. Direct Collocation

We use a direct collocation approach to embed the integration of the nonlinear dynamics inside the optimization problem. Direct collocation is a strategy known from the field of direct optimal control [30] that is based on the discretization of control and state trajectory, therefore lifting the problem to higher dimensional space in comparison to traditional single shooting methods [31]. By embedding the numerical integration inside the optimization problem using collocation variables, the need of an explicit call to an integrator vanishes. We decided to adopt direct collocation for the presented sensor fusion problem because of the potential to decrease the number of iterations by initializing the collocation variables well. The discretization of $\mathbf{x}(t)$ and $\mathbf{u}(t)$ at times $t_{0:N}$ leads to the discrete sets of states $\mathcal{X} = \{\mathbf{x}_0 \dots \mathbf{x}_N\}$ and controls $\mathcal{U} = \{\mathbf{u}_0 \dots \mathbf{u}_{N-1}\}$. While the control trajectory is defined to be piecewise constant, the state trajectory is approximated using Lagrange polynomials $P_{0:D}$ of degree D at Gauss–Radau collocation points $\tau \in \mathbb{R}^D$ [32]. Hence, the state trajectory is represented by the sum of the weighted polynomials

$$\mathbf{x}(t_{k+1}) \approx C(\mathbf{x}_k, \mathbf{c}_k, \tau) = \sum_{m=0}^M \sum_{n=0}^{N_x} c_{k,mn} e_n P_m(\tau), \quad \mathbf{c}_{k,0} = \mathbf{x}_k, \quad 0 > \tau_{0:D} > 1,$$

where the polynomial is scaled by the collocation variables $\mathbf{c}_k \in \mathbb{R}^{N_x \times D}$. The additional collocation variables enter the optimization problem over the collocation states $\mathcal{C} = \{\mathbf{c}_0 \dots \mathbf{c}_{N-1}\}$ and increase the number of decision variables. To retrieve a physically meaningful state trajectory, the collocation variables require being constrained using the system dynamics. This is enforced by constraining the derivative of the time-scaled polynomial at the collocation points $\tau_{1:D}$ to the ODE of the system $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$ evaluated at the same point in time using equality constraints. We obtain D equality constraints for each time interval $[t_k, t_{k+1}]$ of duration $T = t_{k+1} - t_k$ defined by

$$\frac{\partial}{\partial \tau} C_k(\mathbf{x}_k, \mathbf{c}_k, \tau) |_{\tau_d T^{-1}} = f(\mathbf{x}(t_k + \tau_d T), \mathbf{u}_k), \quad d = 1, \dots, D, \quad k = 0, \dots, N-1. \quad (10)$$

The discretization of the state trajectory $\mathcal{X} = \{\mathbf{x}_0 \dots \mathbf{x}_N\}$ requires further equality constraints to obtain a closed state trajectory. The continuity constraints,

$$Z_M \mathbf{x}_{k+1} = Z_M C(\mathbf{x}_k, \mathbf{c}_k, \tau_D), \quad Z_M \mathbf{x}(t) = \mathbf{x}_M(t), \quad \tau_D = 1.0, \quad k = 0, \dots, N-1, \quad (11)$$

constrain the next state to correspond to the propagated previous state. $Z_M \in \mathbb{R}^{N_x \times N_x}$ is defined as a selection matrix for the system model states $\mathbf{x}_M(t)$. The continuity constraint is exclusively applied to the system states $\mathbf{x}_M(t)$ as defined in Equation (3).

2.4.3. Random Walk

As described in Section 2.3, the biases are modeled by a random walk process. This behavior is introduced by relaxing the continuity constraint

$$\delta_{k+1} \approx \delta_k, \quad k = 0, \dots, N-1, \quad (12)$$

and defining additional cost terms

$$c_\delta(\mathbf{x}_k, \mathbf{u}_k, t_k) = \frac{1}{2} \|Z_\delta \mathbf{x}(t_k) - Z_\delta \mathbf{x}(t_{k-1})\|_{Q_\delta^{-1}}^2, \quad k = 1, \dots, N, \quad (13)$$

where the difference between previous and current state is weighted according to the sensor's random walk process noise and minimized accordingly. The selection matrix $Z_\delta \in \mathbb{R}^{3 \times N_x}$ is used to extract the bias states from the state vector $\mathbf{x}(t)$.

2.5. MHE Estimator

The components of the optimization problem introduced in Section 2.4 are leading to the definition of the following minimization problem for the MHE estimator

$$\begin{aligned} \underset{\mathcal{X}, \mathcal{C}, \mathcal{U}}{\text{minimize}} \quad & c_A(\mathbf{x}_j) + \sum_{k=j}^{j+N} [c_p(\mathbf{x}_k, \mathbf{u}_k, t_k) + c_a(\mathbf{x}_k, \mathbf{u}_k, t_k) + c_\omega(\mathbf{x}_k, \mathbf{u}_k, t_k)] \\ & + \sum_{k=j}^{j+N-1} [c_{\delta_a}(\mathbf{x}_k, \mathbf{u}_k, t_k) + c_{\delta_\omega}(\mathbf{x}_k, \mathbf{u}_k, t_k)], \end{aligned} \quad (14a)$$

subject to

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \quad (14b)$$

$$Z_q \mathbf{x}_{j+N} (Z_q \mathbf{x}_{j+N})^T = \mathbf{1}, \quad (14c)$$

$$Z_M \mathbf{x}_{k+1} = Z_M C(\mathbf{x}_k, \mathbf{c}_k, \tau) \quad k = j, \dots, N-1, \quad (14d)$$

$$\frac{\partial}{\partial \tau} C_k(\mathbf{x}_k, \mathbf{c}_k, \tau)|_{\tau_d} T^{-1} = f(\mathbf{x}(t_k + \tau_d T), \mathbf{u}_k) \quad d = 1, \dots, D, k = j, \dots, N-1, \quad (14e)$$

where $N \in \mathbb{N}$ and $T = t_{k+1} - t_k$ define the horizon length and the sampling time. Since the quaternion is an over-parametrization of a rotation, we have to guarantee that the estimated quaternions satisfy the unit norm condition. The quaternion ODE of an orientation as defined in Equation (1c) preserves the unit norm, allowing for adding a single unit norm constraint at the end of an estimation horizon. The constraint (14c) is expressed using the selection matrix Z_q for the quaternion entries of the state vector \mathbf{x}_{j+N} .

The additional cost term in the objective function (14a) defines the arrival cost $c_A(\mathbf{x}_j)$, which summarizes the history of measurements by penalizing deviations to previous estimates of $\mathbf{x}(t_j)$. The arrival cost is defined as a quadratic approximation of the Schur complement correction and is calculated while marginalizing the system before shifting the horizon [33]. Depending on the nonlinearity of the system, the approximation of the past by the arrival cost might be more or less accurate. In an MHE approach, longer horizons can increase the information about the past by considering more measurement data. The benefit of having a more information about the recent past becomes negligible in cases where the approximation of the past by the arrival cost is already accurate, and vice versa.

3. Experimental Results

3.1. Dataset

The MHE estimator described in Section 2 is used in the following to study the influence of the horizon length N on the accuracy of the solution. The dataset used for evaluation has been recorded during a flight with a small Socata single propulsion aircraft (see Figure 3a) flying different maneuvers in hazy weather conditions. In the 100 s long data selection, the airplane completes two sharp turns (see Figure 4), while reaching roll angles up to 60 deg.



Figure 3. Socata airplane and sensors used for data collection.

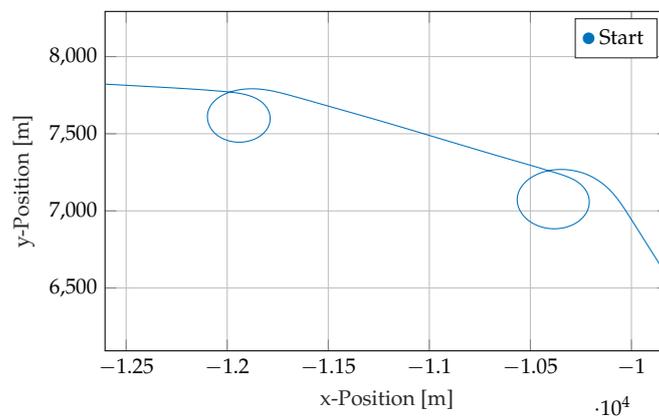


Figure 4. Flight trajectory of airplane used for evaluation (recorded near Enschede, The Netherlands).

3.2. Sensor Setup

The complete sensor setup used for data collection contains two independent sets of sensors: one for the estimation results and one to obtain the reference trajectory. Both sets use a GNSS receiver and an IMU, which differ in their specifications. While the estimation setup includes exclusively consumer-grade sensors, the reference setup uses higher-grade sensors.

To collect the data used for estimation, we used a Xsens MTi-G-700 [34] motion tracker which combines an IMU and GNSS receiver in a single package. By sharing the same signal pipeline, the risk of time synchronization errors between the independent sensors is reduced. The Allan variance curves of the contained MEMS accelerometers and MEMS gyroscopes are plotted in Figure 5 and define the noise density and bias stability of the sensor. The noise density is used to fill the weighting matrices Q_{ω}^{-1} and Q_a^{-1} used in the cost functions (9b) and (9c). The GNSS receiver measures the position and its accuracy at time t_k . The latter is further used to fill the weighting matrix $Q_{p,k}^{-1}$ in Equation (9a). The GNSS data $\bar{y}_{p,k}^L$ was acquired at the maximum rate of 4 Hz and the IMU was configured to stream motion increments, as described in Section 2.3 at an output frequency of 4 Hz. The further available velocity measurements acquired by the GNSS receiver are not considered in this document.

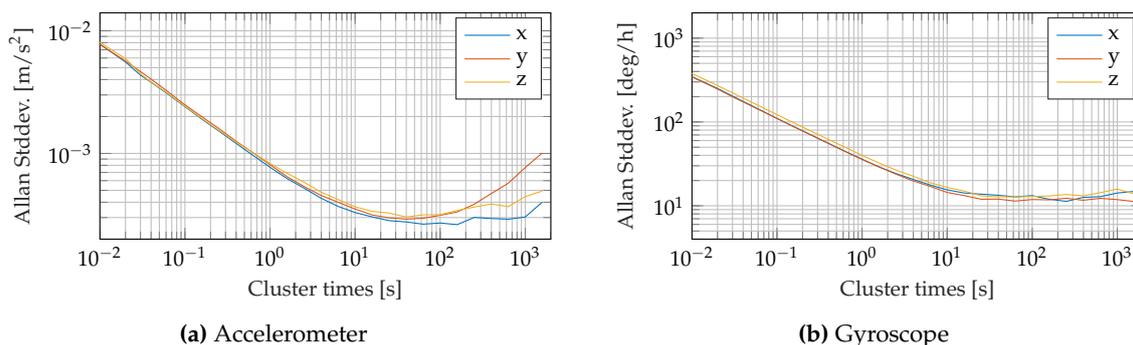


Figure 5. Allan variance curves for accelerometer and gyroscope of the Xsens MTi-700 used for estimation.

Table 1 allows a direct comparison between the noise parameters of the sensors used for estimation and reference. The iMAR-FSAS IMU [35] uses a tactical-grade *fibre optic gyro* (FOG) with lower measurement noise compared to the consumer-grade MEMS gyroscopes used for estimation. Furthermore accurate GNSS receivers were used in a *differential GPS* (DGPS) configuration allowing for very precise positioning with errors below 0.1 m. To improve the accuracy of the ground truth trajectory further, the acquired data of the reference sensors was fused using a batch processing approach [36].

Table 1. Sensor specifications for estimation and reference setup.

		Estimation	Reference
IMU		MTi-G-700	iMAR-FSAS
Gyro Technology		MEMS	FOG
Gyro Rate Bias Repeatability	[deg/s]	0.2	0.21×10^{-3}
Gyro Rate Bias Stability	[deg/h]	10	< 0.1
Angular Random Walk	[deg/ \sqrt{h}]	0.6	0.16
Accelerometer Technology		MEMS	Servo
Accelerometer Bias Repeatability	[mg]	< 5	1.0
Accelerometer Bias Stability	[μ g]	40	< 10
Accelerometer Random Walk	[μ g/ \sqrt{Hz}]	80	< 50
GNSS		MTi-G-700	FlexPak-V2-RT2
Constellation		GPS	GPS + GLONASS
Bands		L1	L1 + L2 (DGPS)
Accuracy	[m]	2–4	0.1
max. Rate	[Hz]	4	40

All sensor devices (Figure 3b) were rigidly attached to the airplane (Figure 3a). The sensor frame S of the IMU was aligned with the body frame of the aircraft, which implies that no further transformations were required to retrieve physically meaningful output for the navigation states defined in Equation (3).

3.3. Algorithm Configuration

Crucial for every kind of nonlinear optimization is the initialization of the decision variables. To initialize the states \mathcal{X} and controls \mathcal{U} , we first compute the initial orientation q_0^{LS} using the velocity vector estimated from consecutive GNSS measurements, assuming the system obeys holonomic

constraints and has sufficient speed. The measurements of GNSS receiver and IMU are used to initialize position p_0^L and controls $\mathbf{u}(t_0)$, respectively. The accelerometer and gyroscope biases are initialized to zero. The resulting initial guess $\mathbf{x}(t_0)$ (as defined in Equation (3)) and its corresponding standard deviations are summarized in Table 2.

Table 2. Parameters of MHE algorithm defining the initial conditions.

Parameter	Unit	Value	Standard Deviation
Position	[m]	$\bar{y}_{p,0}^L$	10
Velocity	[m/s]	$\frac{d}{dt}\bar{y}_{p,0}^L$	5
Orientation	[deg]	$q_0^{LS}(p_0^L, v_0^L)$	8
Accelerometer Bias	[m/s ²]	0	0.05
Gyroscope Bias	[deg/s ⁻¹]	0	0.5

3.4. Horizon Length Evaluation

To analyze the estimation accuracy depending on the horizon length N , the estimation is repeated for different values $N = \{1, 2, 4, 8, 16, 20\}$. The horizon length N directly defines the number of measurements $N_M = N$ and the number of decision variables

$$N_{DV} = N(N_x + N_u + N_{x_M}D) + N_x \quad (15)$$

contained in each MHE optimization problem, where N_x , N_u and N_{x_M} represent the dimensions of the state and control vectors defined in Equations (3) and (4). The degree of the collocation polynomials D defines implicitly the number of collocation variables used for the integration of the navigation states and is set to $D = 3$ for this evaluation.

Figure 6 shows the estimated values for velocity and orientation as well as the calculated reference values. For all configurations, the MHE estimator output follows the reference trajectory and no major differences are observable. The plots reveal, however, that the estimation results recover faster from an incorrect initialization with longer horizons and that, in general, larger horizons show an improved tracking behavior of the reference trajectory. The evolution of the standard deviations of position, velocity, and orientation for the evaluated horizons is plotted in Figure 7. The standard deviations converge after the first turn to a steady state, which is determined by the measurement noise. The turn maneuver results in sufficient excitation of the system to observe the orientation, which is the only navigational state that is not directly related to the measurements of the sensors. In Figure 8, we observe that the bias estimates are converging towards the same steady-state for all evaluated horizons. Furthermore, the evolution of the bias standard deviations confirms the necessity of motion changes to identify sensor parameters. The estimated biases converge only after finishing the first turn at around 40 s to their corresponding steady state.

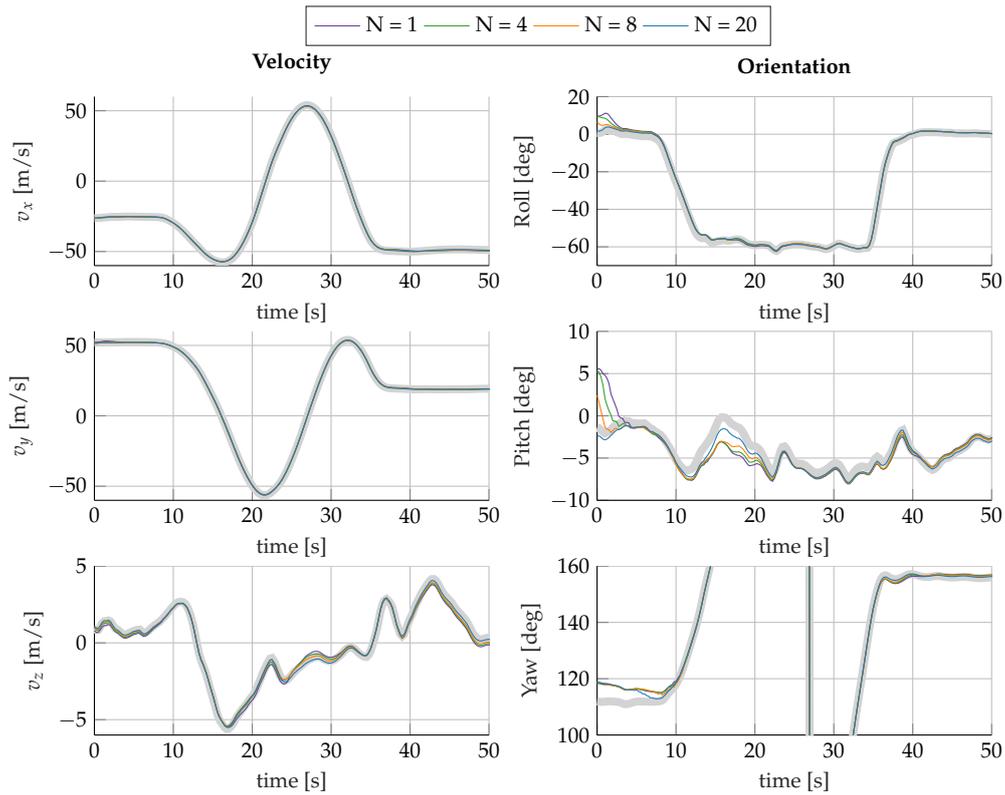


Figure 6. Estimated state trajectories for the first 50 s of velocity and orientation for increasing horizons N , compared to the reference trajectory (—).

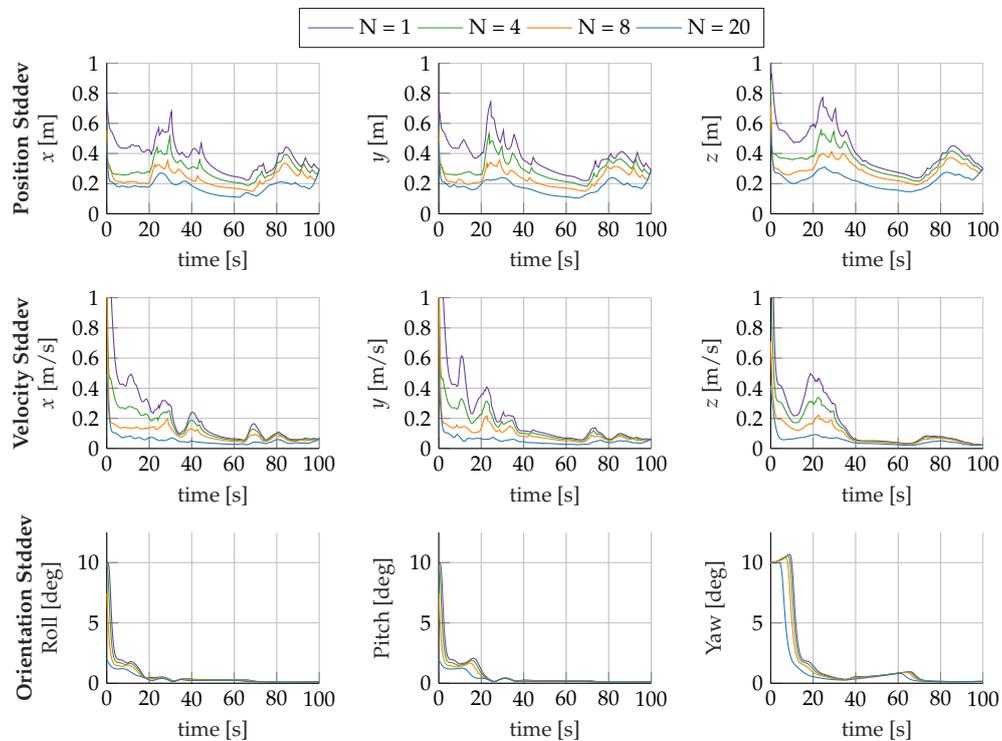


Figure 7. Evolution of standard deviations of the time-lagged estimate of the navigation states: position, velocity, and orientation with increasing horizons N .

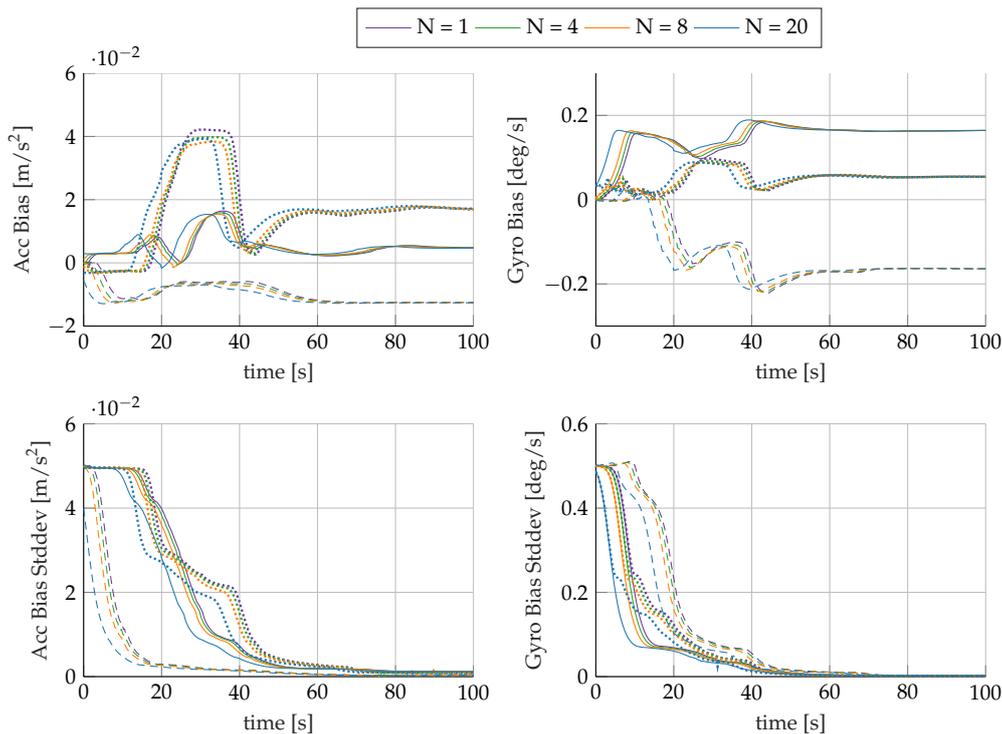


Figure 8. Time-lagged solution for bias values (x —, y , z ---) of accelerometer and gyroscope and their corresponding standard deviations for different horizons N .

Before taking a closer look at the results for different horizons, we define two types of solutions: the real-time solution represents the estimate of the most recent state in the horizon, whereas the lagged solution represents the oldest state in the current window. In the latter case, the value of the state estimate gets updated until it is not contained anymore in the window. Considering a dataset with current data index $j = 1, \dots, J$ of total length J , the lagged solution can be compared to fixed-lag smoothing, maximizing the probability of $p(\mathbf{x}_{j-N}|\mathbf{y}_{1:j})$, whereas the real-time solution corresponds to filtering approach estimating $p(\mathbf{x}_j|\mathbf{y}_{1:j})$. For completeness, we remind readers that a typical batch processing approach estimates $p(\mathbf{x}_{0:j}|\mathbf{y}_{1:j})$. Even though both real-time and lagged solutions contain more measurement data with increasing horizons, there are differences in their interpretation. Both estimators contain an approximation of the past that is defined by the arrival cost $c_A(\mathbf{x}_j)$ in Equation (14a). While the lagged version evaluates the first state in the window, the real-time version estimates the last state in the window, which time is defined by the latest measurement at time t_j . From a time perspective, this means that the lagged version contains information about its past and future, whereas the real-time problem only contains information about its past.

A comparison between real-time and lagged solutions for different horizon lengths is given in Figure 9. A noticeable decrease in terms *root mean square error* (RMSE) with increasing horizons can only be observed for the lagged solution. The accuracy of the lagged solution converges towards the solution of the batch estimator (dotted lines), which can be interpreted as a locally optimal solution since all measurement data is contained in the optimization problem, allowing the removal of start-up effects due to the initialization of the algorithm.

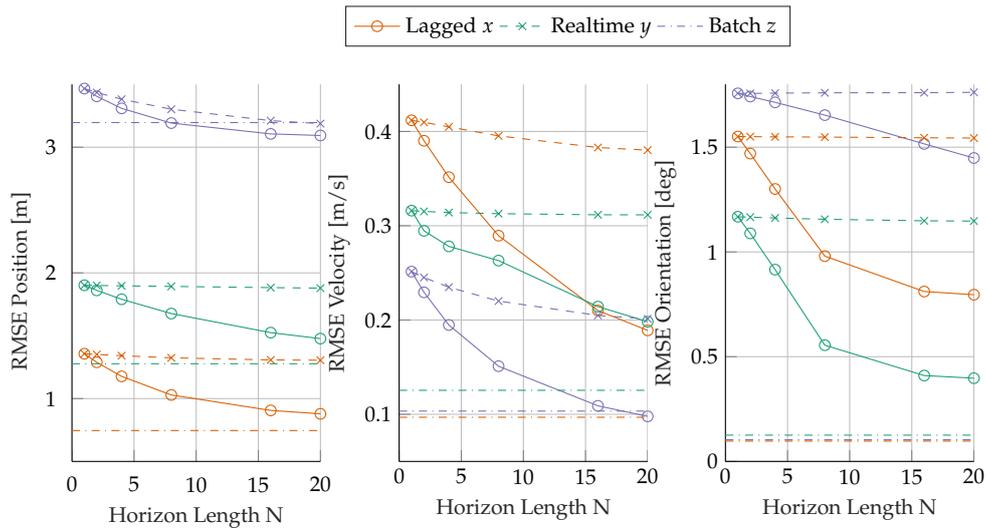


Figure 9. Evaluation of RMSEs for real-time ($-\circ-$) and lagged ($- \times -$) MHE estimators over an increasing horizon length N compared to the batch estimation results ($- \cdot \cdot -$).

In terms of error quantities, we observe relatively large RMSEs for the position estimate, ranging up to 3.5 m in the z -direction. The batch solution confirms this large error of 3.1 m, which allows the conclusion that the error is not coupled to start-up effects or initialization errors but rather a systematic offset between the sensor setups. Table 1 shows that the GNSS receiver, used in the reference setup, operates on the $L1/L2$ -band, which improves position accuracy and especially the altitude estimate compared to a single $L1$ -band GNSS receiver as used for estimation.

The orientation estimate reveals error reductions for roll ϕ and pitch θ angles with increasing horizon length. The yaw angle ψ reveals the largest RMSE, which is an expected behavior due to limited system dynamics contained in the dataset. The 100 s long flight is not subject to larger changes in pitch or strong accelerations in the xy -plane, which contributes to the challenging conditions for the estimation of the yaw angle in the presence of sensor biases. The consequence of a lack of excitation can be further observed in the bias estimates and their standard deviations in Figure 8. The initial jumps of the bias estimates reveal that the biases are not observable under constant motion. After completion of the first turn, the biases in x - and y -directions become observable and the values converge to their corresponding steady state.

Coming back to Figure 9, we notice that only the lagged MHE solution (solid lines) achieves major RMSE improvements in position, velocity, and orientation with an increasing horizon length N . The real-time solution does hardly improve by considering a bigger horizon in comparison to the filter solution, where $N = 1$. The evaluated sensor fusion problem exposes a close to linear behavior, which can be approximated well by an arrival cost. Therefore, only small improvements can be achieved by considering a longer window and, therefore, a more detailed representation of the recent past for the real-time solution.

The nearly linear behavior of the regarded problem is further proven by restricting the optimization algorithm to a maximum number of one iteration [37]. This configuration achieves a KF-like algorithm where only a single linearization of the system is calculated. Analyzing the solution in Figure 10 for the RMSEs in orientation, we notice no visible difference between the single-iteration and the iterated solution that satisfies the exit condition, even though the converged solutions will naturally preserve constraints more accurately.

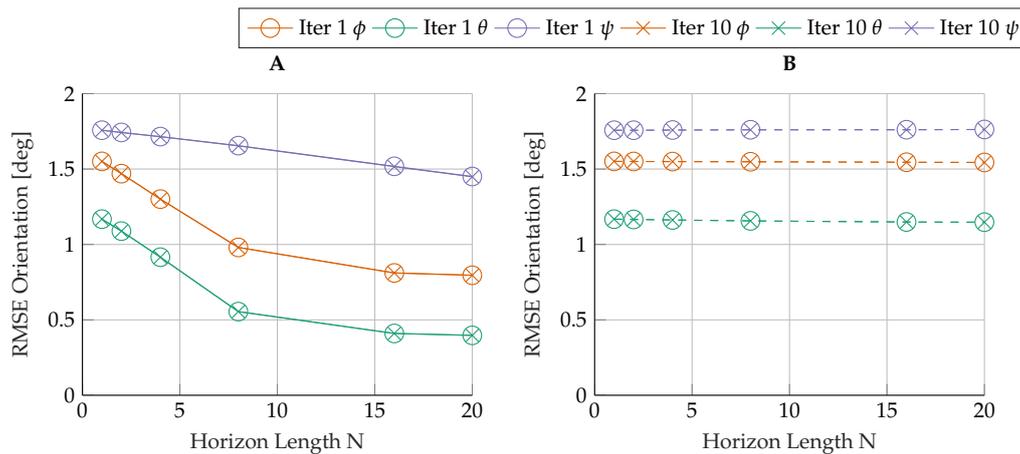


Figure 10. RMSE error in orientation for a different number of iterations $I = \{1, 10\}$. The lagged solutions are plotted on the left (A) and the real-time solutions on the right (B).

4. Discussion

The presented MHE approach for multi-sensor fusion allows a mathematically elegant formulation in a single optimization problem where cost functions and constraints are recursively adapted with the availability of new measurement data. We introduced the modules of the framework by applying it to the commonly known sensor fusion problem of GNSS receiver and IMU and evaluated the accuracy of the solution on a real flight dataset in comparison to an accurate reference trajectory for different horizon lengths $N = \{1, 2, 4, 8, 16, 20\}$. Major improvements for the navigation states position, velocity, and orientation in terms of RMSEs can be achieved when considering either the time-lagged or the batch estimates. With an increasing horizon, the RMSEs decrease and errors approach the results of the post-processed batch solution. For the real-time solution, which estimates the state at the most recent measurement, the performance is only slightly better than the filter solution, corresponding to $N = 1$.

These findings coincide with relevant research [21], where MHE is suggested to improve consistently the accuracy of the estimates at the price of a higher computational cost. In general, it can be said that if applications allow for a time-lagged estimate, the results will be more accurate considering larger horizons. This makes the time-lagged implementation of a MHE-based estimator a versatile approach for systems with slow dynamics and represents an alternative for memory-restrictive offline processing. For real-time critical applications which are governed by nearly linear equations, the accuracy of the state estimates does hardly benefit from the nonlinear optimization approach resulting in a comparable accuracy to filter solutions. The nearly linear behavior of the specific sensor fusion problem was confirmed by evaluating an EKF-like configuration of the MHE estimator using a real-time iteration scheme [37] with a single iteration and a horizon of $N = 1$ yielding similar results in terms of RMSEs. Notice, the considered sensor fusion problem of IMU and GNSS receiver can be considered a linear estimation problem when conditioned on orientation, i.e., the optimization problem reduces to linear system if the orientation trajectory is known a priori.

The presented optimization-based framework for sensor fusion allows a seamless and detailed evaluation of filter, MHE, and batch solutions by adapting the horizon length N . The modular structure contains several key features which we aim to exploit in further research. By encoding the system dynamics in a single set of ODEs, the framework can be adjusted to consider more complex system kinematics in a straightforward manner. Additional measurement updates from further sensors can be included by manipulation of the cost function of the optimization problem. When increasing the number of sensors, the calculation of a common time-grid becomes nontrivial. The presented framework is prepared to account for asynchronous measurements by estimating time-continuous state trajectories using a direct collocation approach. The adaptivity of a MHE-based approach for sensor

fusion is one major advantage compared to traditional filter schemes and combined with efficient numerical solvers MHE is likely to be beneficial for many challenging applications.

The problem of motion tracking does not classify as nearly-linear problem in the general case. By considering measurements of relative distance sensors, such as *ultra-wideband* (UWB), *time of flight* (TOF) or camera sensors, the sensor models become strongly nonlinear. Therefore an MHE approach with the appropriate optimization strategy is expected to improve the real-time estimates as well. For the future, we plan to extend the presented sensor fusion approach of GNSS and IMU to include the full calibration of the sensors, which will increase the nonlinearity of the optimization problem and MHE is expected to outperform conventional filter techniques. The identification of parameters like scale factors, misalignment or time delays will improve the accuracy when using uncalibrated consumer-grade sensors.

Acknowledgments: This work is supported by the EU ITN project AWESCO, which has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie Grant No. 642682.

Author Contributions: Fabian Girrbach, Jeroen Hol, Giovanni Bellusci and Moritz Diehl defined the scope of the paper; Fabian Girrbach developed the algorithms used in this work based on ideas of Jeroen Hol; Fabian Girrbach analyzed the results and drafted the paper, which has been reviewed by all co-authors.

Conflicts of Interest: This document reflects only the author's view. The Commission of the European Union is not responsible for any use that may be made of the information it contains.

References

1. Verschueren, R.; Zanon, M.; Quirynen, R.; Diehl, M. Time-optimal race car driving using an online exact hessian based nonlinear MPC algorithm. In Proceedings of the European Control Conference (ECC), Aalborg, Denmark, 29 June–1 July 2016; pp. 141–147.
2. Gros, S.; Zanon, M.; Vukov, M.; Diehl, M. Nonlinear MPC and MHE for Mechanical Multi-Body Systems with Application to Fast Tethered Airplanes. In Proceedings of the IFAC Conference on Nonlinear Model Predictive Control, Leeuwenhorst, The Netherlands, 23–27 August 2012; pp. 86–93.
3. Chao, Z.; Ming, L.; Shaolei, Z.; Wenguang, Z. Collision-free UAV formation flight control based on nonlinear MPC. In Proceedings of the International Conference on Electronics, Communications and Control (ICECC), Ningbo, China, 9–11 September 2011; pp. 1951–1956.
4. Russo, L.P.; Young, R.E. Moving-Horizon State Estimation Applied to an Industrial Polymerization Process. In Proceedings of the American Control Conference, San Diego, CA, USA, 2–4 June 1999; pp. 1129–1133.
5. Hoffmann, G.; Gorinevsky, D.; Mah, R.; Tomlin, C.; Mitchell, J. Fault Tolerant Relative Navigation Using Inertial and Relative Sensors. In Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, SC, USA, 20–23 August 2007.
6. Quirynen, R.; Vukov, M.; Diehl, M. Auto Generation of Implicit Integrators for Embedded NMPC with Microsecond Sampling Times. In Proceedings of the IFAC Nonlinear Model Predictive Control Conference, Noordwijkerhout, The Netherlands, 23–27 August 2012; pp. 175–180.
7. Frison, G.; Kufoalor, D.K.M.; Imsland, L.; Jorgensen, J.B. Efficient implementation of solvers for linear model predictive control on embedded devices. In Proceedings of the IEEE Conference on Control Applications (CCA), Juan Les Antibes, France, 8–10 October 2014; pp. 1954–1959.
8. Hall, D.L.; Llinas, J. An introduction to multisensor data fusion. In the Proceedings of the Circuits and Systems, Monterey, CA, USA, 31 May–3 June 1998.
9. Sun, S.-L.; Deng, Z.-L. Multi-sensor optimal information fusion Kalman filter. *Automatica* **2004**, *40*, 1017–1023.
10. Mohamed, A.H.; Schwarz, K.P. Adaptive Kalman Filtering for INS/GPS. *J. Geod.* **1999**, *73*, 193–203.
11. Jwo, D.J.; Weng, T.P. An Adaptive Sensor Fusion Method with Applications in Integrated Navigation. In Proceedings of the International Federation of Automatic Control, Seoul, Korea, 6–11 July 2008; pp. 9002–9007.

12. Rao, C.V.; Rawlings, J.B.; Lee, J.H. Constrained Linear State Estimation—A Moving Horizon Approach. *Automatica* **2001**, *37*, 1619–1628.
13. Grrrbach, F.; Hol, J.D.; Bellusci, G.; Diehl, M. Towards Robust Sensor Fusion for State Estimation in Airborne Applications Using GNSS and IMU. *IFAC-PapersOnLine* **2017**, in press.
14. Cox, D.B. Integration of GPS with Inertial Navigation Systems. *Navigation* **1978**, *25*, 236–245.
15. Crassidis, J. Sigma-point Kalman filtering for integrated GPS and inertial navigation. *IEEE Trans. Aerosp. Electron. Syst.* **2006**, *42*, 750–756.
16. Panzneri, S.; Pascucci, F.; Ulivi, G. An outdoor navigation system using GPS and inertial platform. *IEEE/ASME Trans. Mech.* **2002**, *7*, 134–142.
17. Honghui Qi.; Moore, J. Direct Kalman filtering approach for GPS/INS integration. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 687–693.
18. Jones, E.S.; Soatto, S. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Int. J. Robot. Res.* **2011**, *30*, 407–430.
19. Crassidis, J.L.; Markley, F.L.; Cheng, Y. Survey of Nonlinear Attitude Estimation Methods. *J. Guid. Control Dyn.* **2007**, *30*, 12–28.
20. Gros, S.; Diehl, M. Attitude estimation based on inertial and position measurements. In Proceedings of the 51st IEEE Conference on Decision and Control (CDC), Maui, HI, USA, 10–13 December 2012; pp. 1758–1763.
21. Haseltine, E.L.; Rawlings, J.B. Critical Evaluation of Extended Kalman Filtering and Moving-Horizon Estimation. *Ind. Eng. Chem. Res.* **2005**, *44*, 2451–2460.
22. Polóni, T.; Rohal-Ilkiv, B.; Arne Johansen, T. Moving Horizon Estimation for Integrated Navigation Filtering. *IFAC-PapersOnLine* **2015**, *48*, 519–526.
23. Gul, H.U.; Kai, D.Y. An Optimal Moving Horizon Estimation for Aerial Vehicular Navigation Application. In Proceedings of the IOP Conference Series: Materials Science and Engineering, Réunion Island, France, 10–13 October 2016.
24. Vandersteen, J.; Diehl, M.; Aerts, C.; Swevers, J. Spacecraft Attitude Estimation and Sensor Calibration Using Moving Horizon Estimation. *J. Guid. Control Dyn.* **2013**, *36*, 734–742.
25. Rong Li, X.; Jilkov, V. Survey of maneuvering target tracking. Part I: Dynamic models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364.
26. Titterton, D.; Weston, J.L. *Strapdown Inertial Navigation Technology*; AIAA: Reston, VA, USA, 2004; p. 558.
27. El-Sheimy, N.; Hou, H.; Niu, X. Analysis and Modeling of Inertial Sensors Using Allan Variance. *IEEE Trans. Instrum. Meas.* **2008**, *57*, 140–149.
28. Savage, P.G. Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms. *J. Guid. Control Dyn.* **1998**, *21*, 19–28.
29. Kaplan, E.D.; Hegarty, C.J. *Understanding GPS : Principles and Applications*; Artech House: Norwood, MA, USA, 2006; p. 703.
30. Von Stryk, O. Numerical Solution of Optimal Control Problems by Direct Collocation. In *Optimal Control*; Birkhäuser Basel: Basel, Switzerland, 1993; pp. 129–143.
31. Bock, H.G.; Diehl, M.M.; Leineweber, D.B.; Schlöder, J.P. A Direct Multiple Shooting Method for Real-Time Optimization of Nonlinear DAE Processes. In *Nonlinear Model Predictive Control*; Birkhäuser Basel: Basel, Switzerland, 2000; pp. 245–267.
32. Butcher, J.C. Integration Processes Based on Radau Quadrature Formulas. *Math. Comput.* **1964**, *18*, 233.
33. Wang, S.; Chen, L.; Gu, D.; Hu, H. An Optimization Based Moving Horizon Estimation with Application to Localization of Autonomous Underwater Vehicles. *Robot. Auton. Syst.* **2014**, *62*, 1581–1596.
34. Xsens Technologies B.V. Xsens MTi-G-710 Motion Tracker. 2017. Available online: <https://www.xsens.com/products/mti-g-700/> (accessed on 12 May 2017).
35. iMAR Navigation GmbH. iIMU-FSAS: IMU with Odometer Interface and Integrated Power Regulation. Available online: http://www.imar-navigation.de/downloads/IMU_FSAS.pdf (accessed on 12 May 2017).

36. Vydhyathan, A.; Bellusci, G.; Luinge, H.; Slycke, P. The Next Generation Xsens Motion Trackers for Industrial Applications. 2015. Available online: http://https://www.xsens.com/download/pdf/documentation/mti/mti_white_paper.pdf (accessed on 12 May 2017).
37. Diehl, M.; Bock, H.G.; Schlöder, J.P. A Real-Time Iteration Scheme for Nonlinear Optimization in Optimal Feedback Control. *SIAM J. Control Optim.* **2005**, *43*, 1714–1736.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).