EasyChair Preprint

№ 6420

Information-Based Georeferencing of Multi-Sensor-Systems by Particle Filter with Implicit Measurement Equations

Rozhin Moftizadeh, Sören Vogel, Alexander Dorndorf,
Jan Jüngerink and Hamza Alkhatib

August 27, 2021

Information-Based Georeferencing of Multi-Sensor-Systems by Particle Filter with Implicit Measurement Equations

1st Rozhin Moftizadeh
Geodetic Institute
Leibniz Universität Hannover
Hanover, Germany
moftizadeh@gih.uni-hannover.de

2nd Sören Vogel
Geodetic Institute
Leibniz Universität Hannover
Hanover, Germany
vogel@gih.uni-hannover.de

3rd Alexander Dorndorf
Geodetic Institute
Leibniz Universität Hannover
Hanover, Germany
dorndorf@gih.uni-hannover.de

4th Jan Jüngerink
Geodetic Institute
Leibniz Universität Hannover
Hanover, Germany
juengerink@stud.uni-hannover.de

5th Hamza Alkhatib
Geodetic Institute
Leibniz Universität Hannover
Hanover, Germany
alkhatib@gih.uni-hannover.de

***Abstract*—Multi-Sensor-System (MSS) georeferencing is a challenging task in engineering that should be dealt with in the most reliable way possible. The most straight forward way for localizing a MSS is to rely on the Global Navigation Satellite System (GNSS) and Inertial Measurement Unit (IMU) data. However, these data might not be always reliable enough or even available. Therefore, suitable filtering techniques are required to deal with such problems and to increase the reliability of the estimated states. In global localization and when it comes to real scenarios, particle filters are proven to deliver more realistic results than Kalman filter realizations. However, in MSS georeferencing where multiple sensors are used, different observation models are needed some of which could be of implicit type. In such a case, the likelihood estimation is challenging due to impossibility of estimating the observations by means of the generated samples. Therefore, the current paper offers a new particle filter methodology that can handle both implicit and explicit observation models. Final results of this methodology, which is applied on a simulated environment for georeferencing a MSS, are shown to be satisfactory.**

***Index Terms*—georeferencing, MSS, particle filter, implicit observation model, 6 DoF, Monte Carlo simulation**

I. Introduction

When various sensors are installed on a single platform, the resulting system is generally referred to as a Multi-Sensor-System (MSS). MSSs are of great importance and use in engineering field for measuring an environment of interest by using multiple sensors such as cameras, scanners, etc. To combine the derived measurements for further analysis purposes, it is important to have the MSS georeferenced. Georeferencing means to have the pose of the MSS with respect to a superordinate coordinate system. The MSS pose – which is also referred to as the six Degrees of Freedom (6 DoF) – consists of its three positions and three orientations in a global frame. Furthermore, usually for kinematic MSSs, the motion parameters such as velocity and acceleration are also of interest to be determined.

In real applications and when it comes to localization, the most straightforward way of georeferencing is to use the Global Navigation Satellite System (GNSS) and Inertial Measurement Unit (IMU) data. Nonetheless, using such data is not always beneficial or even reliable enough. On the one hand, as much as the GNSS measurements can be relied on in rural areas, they are not trustworthy in urban environments due to the existence of large buildings. The signal qualities of the satellites are greatly affected by such buildings and even in some cases a complete blockage of the signals is possible. Generally, a good satellite geometry results in an accuracy in meter range for the GNSS data; unless, differential approaches are applied, which can increase the accuracy up to decimeter or even centimeter level [1], [2]. On the other hand, the IMU measurements are prone to drift over time caused by different error sources and biases that are widely discussed in literature [3]–[5]. A low-cost IMU can have a heading angle accuracy of 0.8° and an accuracy of 0.1° for the roll and pitch angles [2]. Consequently, depending on the application and the required accuracy, the GNSS and IMU data cannot always be the optimal solutions to the localization problems. Instead, proper methodologies, that can well compensate for any possible errors or gaps within these data sets, should be developed.

One of the solutions to the aforementioned problem is to apply the Linear Kalman Filter (LKF) by using only the GNSS and IMU data. However, an optimal solution by means of such a methodology could not be expected in challenging areas due to a possible decreased number of observations. Therefore, as discussed and processed in [6], to increase the localization performance, different measurements derived from various

This research was funded by the German Research Foundation (DFG) as part of the Research Training Group [RTG 2159].

sensors as well as the map data could be combined. This would not only increase the accuracy of the georeferencing solutions, but also counts for their integrity, continuity and availability. Such a sensor and map fusion for the MSS georeferencing purpose is already encountered with in the developed Kalman Filter (KF) frameworks by [2], [7], [8], which enable including both implicit and explicit observation models – referred to as Gauss-Helmert-Model (GHM) and Gauss-Markov-Model (GMM), respectively – in the filtering procedure. In such proposed KF-based methodologies, a suitable initial position is assumed to be available. Having such a proper initialization is the main distinction between the "position tracking" and the "global localization" problems [9]. In the "position tracking" problems, the prototype of the algorithms is Kalman Filter [10], while in the "global localization", common frameworks such as Multi Hypotheses Localization, Histogram Filters and Particle Filters are used [11]. When it comes to real applications, having always a proper initial pose guess cannot be guaranteed. Also, making simplifying assumptions such as normally distributed noise values – which is the case in KF realizations – might lead to estimated states that are not realistic enough. Therefore developing an appropriate Particle Filter (PF) framework that enables data fusion for MSS geoereferencing seems to be a good alternative to the KF methodologies. Consequently, the current paper has focused on developing a PF framework that can handle both implicit and explicit observation models and thus can be used for sensor data fusion when it comes to MSS georeferencing.

The paper is organized as follows. In section II, the mostly relevant researches to the current paper are summarized. In section III, the mathematical models, which are the basis of the developed methodology as well as the proposed framework are explained. Section IV is dedicated to the application of the proposed algorithm on a simulated Unmanned Aerial Vehicle (UAV) environment. In section V, the paper is summarized and the highlighted conclusions and the potential future work in the same area are presented.

II. Related Work

Depending on the measurement scenario e.g. the type, number and accuracy of the installed sensors on the platform, MSS georeferencing could be done by means of different approaches [7]. In outdoor applications, georeferencing is usually dealt with directly (sensor-driven), indirectly (target-driven) or by using available reference data sets (data-driven) [12], [13]. For indoor scenarios, different methodologies are proposed that are presented by [14] in detail. The scope of the current paper is outdoor scenarios and hence the related georeferencing approaches to this kind are given in the following.

In "sensor-driven georeferencing", the GNSS [13] and IMU [15] measurements are directly used to represent the 6 DoF with respect to a superordinate coordinate system. In "target-driven georeferencing", already referenced targets with respect to a certain superordinate coordinate system are used to derive the MSS pose with respect to that specific frame. If instead of targets, referenced data sets are used for localization, the georeferencing is referred to as "data-driven". Such data sets could be 3D point clouds [16], digital surface models or 3D city models [14], [17], [18].

Moreover, to deal with possible errors of sensor measurements and hence to increase the reliability of the solutions, filtering techniques are used. Among such methods, the KF could be mentioned as one of the well-known methodologies that is widely used in engineering navigation and deformation analysis. In this filtering technique, the system state is sequentially estimated based on the information from the system model as well as the external measurements derived from different sensors [19]. If both the system and observation models are linear and the process noise is Gaussian, the filter is generally referred to as LKF, which is capable of delivering the optimal solutions [20]. However, in real applications where complicated systems equipped with multiple sensors are studied, the system and observation models are usually nonlinear and hence the LKF framework cannot be optimum. Such nonlinear measurement equations can happen due to the direct nature of the relation between the observations and state parameters or due to the transformation of the observations from a local coordinate system of the sensor to a certain superordinate coordinate system, which is often the case in navigation applications [19]. To deal with such nonlinear models, a number of approximate nonlinear filters have been proposed, e.g. by [21], [22]. Generally and not from detailed aspects, [19] has categorized these methods into two groups of "Gaussian Approximate Methods" and "Sequential Monte-Carlo Methods", which are explained in more detail in the following.

A. Gaussian Approximate Methods

To overcome the nonlinear models in the KF methodology, one of the well-known approaches is to linearize them by means of Taylor series expansion around a certain state [23], [24]. In such a case, the resulting KF realization is generally referred to as the Extended Kalman Filter (EKF). However, depending on the involved nonlinear functions, the linearization could get complex and thus it can lead to an inefficient filter. To fuse measurements of different sensors of a UAV, [25] combines the GNSS and IMU data by means of the EKF algorithm. Also, EKF is applied by [26] to increase the accuracy of the pose estimations for collaboration of several UAVs by combining multiple Simultaneous Localization And Mapping (SLAM) algorithms. However, the first-order and higher-order EKF diverges if the involved models in the filter are highly nonlinear [27]. To deal with such a problem, different approaches are proposed in literature among which the Iterated Extended Kalman Filter (IEKF) could be pointed out. In IEKF, a re-linearization around the recent updated state is applied to overcome the linearization errors, which in [28] is shown to be an application of the Gauss-Newton method for approximating a solution. So far, the IEKF has widely been used to overcome the nonlinear GMMs. In such models, the observations are explicitly related to the unknown parameters.

However, in real applications due to the existence of multiple sensors, the observations and unknowns might be related to each other implicitly, yielding the GHM. References [29] and [30] have used such models within the IEKF framework. Nonetheless, in the navigation context and for the first time, [8] has proposed an IEKF framework with implicit measurement equations for the purpose of MSS georeferencing. The proposed methodology is then further applied by [2], [31], [32] to localize various MSSs in different scenarios.

B. Sequential Monte-Carlo Methods

An alternative to "Gaussian Approximate Methods" is the Sequential Monte-Carlo (SMC) filter, which is also referred to as PF. This methodology is a sub-optimal filter for the implementation of Bayesian filter by Monte-Carlo (MC) techniques [27], [33]. Instead of approximating a Gaussian distribution for the system states, in SMC the posterior Probability Density Function (PDF) is estimated by using a set of randomly generated samples, which are also referred to as particles. Due to the capability of the PF methodology to handle highly nonlinear models, it is widely used in literature for different purposes. In [34], a good overview of the Bayesian filters is given from both theoretical as well as practical aspects. In [35] a particle filter methodology is developed to deal with Ultra-Wide-Band (UWB) measurements for the matter of robot localization. Furthermore, in [9] using different observation models in PF localization methods is explored and two PF strategies namely Sample Importance Resampling (SIR) and Auxiliary Particle Filter (APF) are investigated. In [36], two localization algorithms based on the Unscented Kalman Filter (UKF) and PF framework are proposed and compared against each other as well as against a previously developed EKF algorithm for a robot in a simulated environment. Moreover, a novel particle filter algorithm is proposed in [37] for vehicle localization based on a map-matching strategy.

As much as the PF framework is used to deal with applications that have explicit observation models, its use in conjunction with implicit measurement equations has yet not been investigated. When it comes to real navigation applications, on the one hand no proper initial guess about the MSS pose as well as a normally distributed process noise could be guaranteed. On the other hand, in order to fuse measurements of different sensors, implicit observation models might be needed. Consequently, the current work is focused on introducing a PF framework that can deal with both explicit and implicit observation models. This newly developed methodology is referred to as "Particle Filter with Implicit Measurement Equations (PFI)" in the following.

III. Methodology

A MSS covers a broad range of sensors that could be attached to a vehicle, a drone, a robot, etc. for the matter of localization. The current paper has focused on a simulated scenario for georeferencing a UAV that is equipped with a GNSS, an IMU, and a Velodnye Puck 3D laser scanner, which is explained in detail in section IV. The whole environment is simulated in such a way to represent a case study with low-cost sensors and a surrounding that represents an inner-city area. Also, it is assumed that the sensors are timely synchronized and their calibration with respect to the platform is error-free. In the following the mathematical models related to this case study as well as the proposed methodology to deal with its localization are explained.

A. Mathematical Model

The main aim is to derive the states vector $\boldsymbol{x}$ at each epoch k, by using the measurements $\boldsymbol{l}$ of different sensors in that epoch. The state and observation vectors are as follows:

$$\boldsymbol{l}_k = \left[\boldsymbol{l}_k^{LS}, \boldsymbol{l}_k^{G}, \boldsymbol{l}_k^{I}\right]^T \;,\; \boldsymbol{x}_k = \left[\boldsymbol{x}_k^{P}, \boldsymbol{x}_k^{O}, \boldsymbol{x}_k^{V}\right]^T \tag{1}$$

wherein, $\boldsymbol{l}^{LS}$, $\boldsymbol{l}^{G}$, and $\boldsymbol{l}^{I}$ are the local laser scanner, GNSS, and IMU observations, respectively. The GNSS observations $\boldsymbol{l}^{G}$ consist of the 3 MSS positions in the global frame G_X,G_Y,G_Z and the IMU measurements $\boldsymbol{l}^{I}$ are the MSS orientations ω,φ,κ. As mentioned in [7], due to discontinuity and singularity problems, the Euler angles suffer from instability and instead quaternions are better to be used. However, in the current paper to better interpret the simulation results, the quaternion representation is avoided and instead the rotation matrices are used to deal with the Euler angle challenges. $\boldsymbol{x}^{P}$, $\boldsymbol{x}^{O}$, and $\boldsymbol{x}^{V}$ are parameter vectors containing three translations t_x,t_y,t_z, three orientations ω,φ,κ, and three directional velocities V_x,V_y,V_z, respectively.

Generally, in filtering methods, the states vector with dimension n_x, $\boldsymbol{x} \in \mathbb{R}^{n_x}$ could be described by a dynamic model as follows:

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{w}_{k-1}) \quad , \quad \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{ww}}) \tag{2}$$

wherein, $f(\cdot)$ is a known and generally nonlinear function, k is the time index, $\boldsymbol{u}$ is a known deterministic vector of control variables, and $\boldsymbol{w}$ is the process noise vector that is considered to count for those environmental effects that cannot be modelled by $f(\cdot)$ and have a known PDF with Variance-Covariance-Matrix (VCM) of $\boldsymbol{Q}_{\boldsymbol{ww}}$. In the current paper, instead of a nonlinear model, a linear system model is used as follows:

$$\boldsymbol{x}_k = \boldsymbol{F}_{x,k-1}\, \boldsymbol{x}_{k-1} + \boldsymbol{\omega}_{k-1} \quad , \quad \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{ww}}) \tag{3}$$

wherein, $\mathbf{F}_x$ is the transition matrix:

$$\boldsymbol{F}_x = \begin{bmatrix} \boldsymbol{I}_{[3\times3]} & \boldsymbol{0}_{[3\times3]} & diag([\Delta\tau, \Delta\tau, \Delta\tau]) \\ \boldsymbol{0}_{[3\times3]} & \boldsymbol{I}_{[3\times3]} & \boldsymbol{0}_{[3\times3]} \\ \boldsymbol{0}_{[3\times3]} & \boldsymbol{0}_{[3\times3]} & \boldsymbol{I}_{[3\times3]} \end{bmatrix} \tag{4}$$

$\boldsymbol{I}$ is an identity matrix and $\Delta\tau$ is the time period between two consecutive epochs, which in the current paper for the simulated data is set to 1 second.

The VCM of the process noise is selected inspired by the given principle of "Continuous White Noise Acceleration Model" in [21] as follows:

$$\boldsymbol{Q}_{\boldsymbol{ww}} = \boldsymbol{Q} \cdot \tilde{\boldsymbol{q}} \tag{5}$$

$$\boldsymbol{Q} = \begin{bmatrix} \frac{(\Delta\tau)^3}{3} \cdot \boldsymbol{I}_{6\times6} & \boldsymbol{0}_{6\times3} \\ \boldsymbol{0}_{3\times6} & (\Delta\tau) \cdot \boldsymbol{I}_{3\times3} \end{bmatrix} \tag{6}$$

$$\tilde{\boldsymbol{q}} = diag(\boldsymbol{q}_T, \boldsymbol{q}_O, \boldsymbol{q}_V) \tag{7}$$

$$\boldsymbol{q}_T = [a, a, a] \quad , \quad \boldsymbol{q}_O = [b, b, b] \quad , \quad \boldsymbol{q}_V = [c, c, c] \tag{8}$$

wherein, $\mathbf{Q}$ is derived based on the particular solution of the superposition law that is applied to the system noise (cf. [21]), and $\tilde{\mathbf{q}}$ are the continuous-time process noise intensities that appear as parameters to be designed, which are assumed to be constant over time for all the states parameters as shown in (8). In Table I, the considered values for the current paper are given.

Generally, the sensor(s) measurements, with dimension n_l at epoch k and $\boldsymbol{l} \in \mathbb{R}^{n_l}$, could be related to the states vector by means of an observation model that in an explicit case can be expressed as follows:

$$\boldsymbol{l}_k + \boldsymbol{v}_k = h(\boldsymbol{x}_k) \quad , \quad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{vv}}) \tag{9}$$

wherein, h is a known and generally nonlinear function, and $\boldsymbol{v}$ is the measurement noise vector with a known PDF and VCM of $\boldsymbol{Q}_{\boldsymbol{vv}}$, that is considered to be mutually independent from the process noise vector $\boldsymbol{w}$. If the observations cannot be explicitly related to the unknowns, an implicit measurement equation holds, which could be expressed as follows:

$$h(\boldsymbol{l}_k + \boldsymbol{v}_k, \boldsymbol{x}_k) = \boldsymbol{0} \quad , \quad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{vv}}) \tag{10}$$

As stated before, in the current paper, the simulated data include the GNSS, the IMU, and the 3D scanner observations. For the GNSS and IMU, explicit observation models as (9) hold, which are:

$$\boldsymbol{l}_k^G + \boldsymbol{v}_k^G = \boldsymbol{x}_k^P \quad , \quad \boldsymbol{l}_k^I + \boldsymbol{v}_k^I = \boldsymbol{x}_k^O \tag{11}$$

wherein, $\boldsymbol{v}^G$ and $\boldsymbol{v}^I$ are the measurement noise of GNSS and IMU, respectively. To relate the simulated scanned data of the 3D scanner to the unknown parameters, the main idea of the proposed IEKF methodology of [8] is used. According to this idea, which is also applied by [2] for the matter of georeferencing, the scanned data of the scanner are related to the buildings of the surrounding environment by using the 3D city models. In such a case, an implicit observation model as of (10) could be established, which is:

$$\boldsymbol{s} = \boldsymbol{n}_{\boldsymbol{x}k} \cdot \boldsymbol{X}_k + \boldsymbol{n}_{\boldsymbol{y}_k} \cdot \boldsymbol{Y}_k + \boldsymbol{n}_{\boldsymbol{z}k} \cdot \boldsymbol{Z}_k - \boldsymbol{d}_k = \boldsymbol{0} \tag{12}$$

In this observation model, which is the Hesse normal form of a plane, $\boldsymbol{n_x}$, $\boldsymbol{n_y}$, and $\boldsymbol{n_z}$ are the planes' normal vector parameters in epoch k. $\boldsymbol{d}$ is the planes' distance parameter vector in the same epoch. These plane parameters are taken from the 3D city model as additional information, which are taken as deterministic known values. $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$ are the transformed laser scanner 3D point cloud from local to the global coordinate system. The transformation is done as follows:

$$\boldsymbol{P}_{glo,k} = \boldsymbol{t}_k + \boldsymbol{R}_k \boldsymbol{P}_{loc,k} \tag{13}$$

wherein, $\boldsymbol{P}_{glo,k}$ are the transformed scanned points, $\boldsymbol{t}_k$ and $\boldsymbol{R}_k$ are the translation parameters vector and rotation matrix, respectively, and $\boldsymbol{P}_{loc,k}$ are the local scanned data in matrix form. In this equation, $\boldsymbol{t}_k$ and $\boldsymbol{R}_k$ are arranged according to [2] and contain the 6 DoF, which are parts of the states vector as given in (1).

B. Particle Filter with Implicit Measurement Equations

The generic PF, which was introduced by [11] could be summarized as the following steps. These steps are also the basis of Algorithm 1 in the current paper and hence the corresponding lines are mentioned in the following:

1) M initial particle sets $\boldsymbol{\chi}_0^{(i)}$ should randomly be generated from a proposal PDF $\pi(\cdot)$, each a representative for the states vector of the system (line 1).
2) The generated particles should then be propagated to the next epoch by using the dynamic system model (2). This yields the so-called "predicted" particles $\boldsymbol{\chi}_{k,-}^{(i)}$ (line 6).
3) The likelihood of the predicted particles $L\left(\boldsymbol{\chi}_{(k,-)}^{(i)}\right)$ should be computed by using the observations in the current epoch k and the measurement model (line 23)
4) To each particle set a weight should be assigned (cf. [27] (line 25).
5) To avoid the "degeneracy" problem, the particles should be resampled, meaning that randomly new particle sets $\boldsymbol{\chi}_{k,+}^{(i)}$ should be generated based on the relative weights of the previous step (line 28).
6) The states vector and its covariance are calculated based on the resampled particles from step 5 and their normalized weights (lines 30 and 31).
7) The resampled particles are then propagated to the next epoch and the sequence is repeated from step 2 to 7 for each epoch.

The basis for estimating the likelihoods in the generic particle filter is to compare the estimated observations $\hat{\boldsymbol{l}}_k$ with the original measurements $\boldsymbol{l}_k$. Such a comparison is only possible if the observation model is of explicit type as (9). This way, the measurements can be estimated by substituting the predicted particles in the observation model and then the differences of these estimations to the real sensor measurements $\hat{\boldsymbol{v}}_k = \hat{\boldsymbol{l}}_k - \boldsymbol{l}_k$ are used to calculate the likelihoods. However, if the observation model is of implicit type, such a likelihood estimation based on the calculated residuals is no more possible. In such a case, substituting the particles – which are states representatives – in (10) does not yield the estimated observations to then be compared with the original measurements. Therefore, a new strategy is required to estimate the likelihoods, which is the main aim of the current paper. For that, the suggested idea is to estimate the likelihood of each particle set by directly using the implicit observation model. In the current paper, the observation model is according to (12), which could be treated as an indicator for the quality of the particle set when the laser scanner observations are taken into account. In other words, a particle set is more probable to be correct if $\boldsymbol{s}$ in equation (12) is closer to zero, which in turn means that the transformed

3D points have properly lied on their corresponding assigned building planes and thus have small so-called "misclosure" values. Therefore, it is claimed that such misclosure values in the observation model (12) carry comparable information content for likelihood estimations as the residuals $\hat{\boldsymbol{v}}_k = \hat{\boldsymbol{l}}_k - \boldsymbol{l}_k$ in an explicit observation model as (9). In this paper, $\hat{s}_j$ is used to represent the estimated misclosure value of the j^{th} scanned point, whose likelihood is assumed to be normally distributed with the expected value and standard deviation of zero and $\sigma_{\hat{s}}$, respectively. The estimated likelihoods of all the misclosure values are then multiplied, so that to each particle set one likelihood value is assigned based on the laser scanner data according to the following formulae:

$$L^{LS}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = \prod_{j=1}^{m} p\left(\hat{s}_{(j,k)} | \boldsymbol{\chi}^{(i)}_{(k,-)}, \sigma^2_{\hat{s}}\right) \tag{14}$$

wherein, m is the total number of the scanned 3D points in epoch k. For calculating the likelihoods of the particle sets based on the GNSS and IMU observations, the residuals $\hat{\boldsymbol{v}}^G_k$ and $\hat{\boldsymbol{v}}^I_k$ are used, as for these measurements an explicit observation model holds as shown in (11). Similar to (14), the likelihood of each particle set based on the GNSS and IMU observations could be computed as follows:

$$L^{G}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = \prod_{j=1}^{3} p\left(\hat{v}^G_{(j,k)} | \boldsymbol{\chi}^{(i)}_{(k,-)}, \sigma^2_{\hat{v}^G}\right) \tag{15}$$

$$L^{I}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = \prod_{j=1}^{3} p\left(\hat{v}^I_{(j,k)} | \boldsymbol{\chi}^{(i)}_{(k,-)}, \sigma^2_{\hat{v}^I}\right) \tag{16}$$

After deriving the likelihoods of each particle set based on measurements of each individual sensor, they are combined as follows:

$$L\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = L^{LS}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) \cdot L^{G}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) \cdot L^{I}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) \tag{17}$$

Since in practice due to large amount of points, such a likelihood multiplication could lead to numerical issues, in the current paper, the logarithm of the likelihoods is taken into account. Algorithm 1 shows the general framework of the suggested PFI. The selected $\sigma_{\hat{s}}$, $\sigma_{\hat{v}^G}$ and $\sigma_{\hat{v}^I}$ values for the current paper are given in Table I. These values are claimed to be among design parameters that should be selected by the user based on the application. In the current paper, a sensitivity analysis is done for that purpose. Also, for this work, the proposal distribution is always selected to be Gaussian. Furthermore, the "Residual Resampling" technique (cf. [38]) is used for resampling the particles based on their importance weights. In the given algorithm, $\boldsymbol{idx}$ in line 27 represents a vector containing the indices of those particles that are selected based on the used resampling technique.

Algorithm 1: The algorithm of Particle Filter with Implicit Measurement Equations (PFI).

1 Initialization: $\left\{\boldsymbol{\chi}^{(i)}_{(0)}\right\}_{i=1}^{M} \sim \mathcal{N}(\boldsymbol{x}_0, \boldsymbol{Q}_{xx,0})$
2 **while** $k <$ K **do**
3 *Prediction*
4 **for** $i = 1 \dots M$ **do**
5 $\boldsymbol{w}_{k-1} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{ww}})$
6 $\left\{\boldsymbol{\chi}^{(i)}_{(k,-)}\right\} = \boldsymbol{F}_x \cdot \left\{\boldsymbol{\chi}^{(i)}_{(k-1,+)}\right\} + \boldsymbol{w}_{k-1}$
7 *Likelihood estimation*
8 **for** $i = 1 \dots M$ **do**
9 Likelihood based on the scanned data:
10 Using the 6 DoF from $\left\{\boldsymbol{\chi}^{(i)}_{(k,-)}\right\}$ to derive $\boldsymbol{t}_k$ and $\boldsymbol{R}_k$
11 Calculating $\boldsymbol{P}_{glo,k}$ based on Eq. (13)
12 Calculating $\hat{\mathbf{s}}$ based on Eq. (12)
13 $L^{LS}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = \prod_{j=1}^{m} p\left(\hat{s}_j | \boldsymbol{\chi}^{(i)}_{(k,-)}, \sigma^2_{\hat{s}}\right)$
14 Likelihood based on the GNSS data:
15 Setting $\hat{\boldsymbol{l}}^G_k$ to the translation parameters in $\left\{\boldsymbol{\chi}^{(i)}_{(k,-)}\right\}$
16 $\hat{\boldsymbol{v}}^G = \hat{\boldsymbol{l}}^G_k - \boldsymbol{l}^G_k$
17 $L^{G}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = \prod_{j=1}^{3} p\left(\hat{v}^G_j | \boldsymbol{\chi}^{(i)}_{(k,-)}, \sigma^2_{\hat{v}^G}\right)$
18 Likelihood based on the IMU data:
19 Setting $\hat{\boldsymbol{l}}^I_k$ to the orientation parameters in $\left\{\boldsymbol{\chi}^{(i)}_{(k,-)}\right\}$
20 $\hat{\boldsymbol{v}}^I = \hat{\boldsymbol{l}}^I_k - \boldsymbol{l}^I_k$
21 $L^{I}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = \prod_{j=1}^{3} p\left(\hat{v}^I_j | \boldsymbol{\chi}^{(i)}_{(k,-)}, \sigma^2_{\hat{v}^I}\right)$
22 Fusing likelihoods of the sensors:
23 $L\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) = L^{LS}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) \cdot L^{G}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right) \cdot L^{I}\left(\boldsymbol{\chi}^{(i)}_{(k,-)}\right)$
24 *Assignment of the importance weights*
25 $\boldsymbol{W} = \frac{1}{M} \cdot \boldsymbol{L}$
26 *Resampling of the particles based on [38]*
27 $\boldsymbol{idx} = ResidualResampling(1:M, \boldsymbol{W})$
28 $\left\{\boldsymbol{\chi}^{(i)}_{(k,+)}\right\}_{i=1}^{M} = \boldsymbol{\chi}^{(\boldsymbol{idx})}_{(k,-)}$
29 *State vector estimation and the corresponding VCM*
30 $\hat{\boldsymbol{x}}_k = \frac{1}{M} \cdot \sum_{i=1}^{M} \boldsymbol{\chi}^{(i)}_{(k,+)}$
31 $\hat{\boldsymbol{Q}}_{\boldsymbol{xx},k} = \frac{1}{M-1} \sum_{k=1}^{k=M} \left(\boldsymbol{\chi}^{(i)}_{(k,+)} - \hat{\boldsymbol{x}}_k\right)\left(\boldsymbol{\chi}^{(i)}_{(k,+)} - \hat{\boldsymbol{x}}_k\right)^T$
32 *Particle regeneration for the next epoch*
33 $\left\{\boldsymbol{\chi}^{(i)}_{(k,+)}\right\}_{i=1}^{M} \sim \mathcal{N}(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{Q}}_{\boldsymbol{xx},k})$

IV. Application

In the current paper, the proposed particle filter algorithm is applied on a simulated environment. To estimate the states by means of this algorithm, a total amount of 1000 particle sets are generated in each epoch (index M in Algorithm 1 is 1000). Also, to better see the performance of this new methodology, both LKF and the IEKF algorithm given in [2] are also applied on the same environment and the results are compared.

A. Simulated Environment

The simulated environment for this paper is selected to be the one given in [2] and [7], in which a UAV is equipped with a GNSS, an IMU, and a 3D laser scanner. The surrounding environment, which is created by unreal building models and a ground plane, represents a 3D city model from which the additional information for the PFI and IEKF is extracted. Figure 1 shows an overview of this environment in which the UAV is assumed to cover a distance of 70 meters in the y-axis in 70 epochs while ascending for 4 meters in the z-axis. The dotted black lines in this figure show the scan lines of the UAV that moves through the environment in y direction. In

Table I, the accuracy and noise values that are considered in the simulated environment are given.

To ensure the functionality of the proposed methodology, 1000 Monte Carlo (MC) runs are performed and in each run new GNSS and IMU observations are generated by adding random noise to the true states and the three filters (PFI, IEKF and LKF) are used to estimate the states vector.

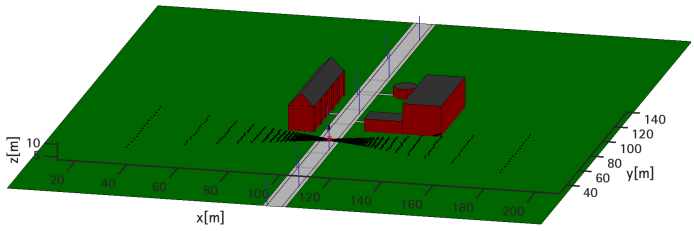

Fig. 1. 3D view of the simulated environment.

TABLE I
APPLIED ACCURACY, DESIGN PARAMETERS $\tilde{\boldsymbol{q}}$ OF THE SYSTEM NOISE VALUES AND STANDARD DEVIATIONS OF THE LIKELIHOODS.

Initial state accuracy	$\sigma_{T,0} =$	$0.5\ m$
	$\sigma_{O,0} =$	0.2°
	$\sigma_{V,0} =$	$1\ m/s$
System noise	$\Delta\tau =$	$1\ s$
	$a =$	$0.12\ m^2/s^3$
	$b =$	$2.3 \times 10^{-6}\ rad^2/s^3$
	$c =$	$0.01\ m^2/s^3$
Measurement noise	$\sigma_{LS} =$	$0.02\ m$
	$\sigma_{GNSS} =$	$0.5\ m$
	$\sigma_{IMU} =$	0.2°
Standard deviations	$\sigma_{\hat{s}} =$	$0.5\ m$
	$\sigma_{\hat{v}G} =$	$0.5\ m$
	$\sigma_{\hat{v}I} =$	0.2°

B. Results

In the following, results of the three filters after performing the MC runs are depicted, which are averaged in each epoch. Also, in each MC run, to avoid high computation times only 20% of the laser scanner observations are considered, which would mean a decrease from approximately 10000 measurements to 2000. Such a random measurement subsampling in a real case scenario could lead to a high loss of information content; however, in the simulated environment of the current paper, it is claimed not to have a significant effect.

Figures 2 and 3 show the average Root Mean Square Error (RMSE) of the estimated translation and orientation parameters, respectively in each epoch over the MC runs. The RMSE value in each MC run for the translation parameters is calculated as follows (same also holds for the orientation parameters):

$$\boldsymbol{RMSE}_k = \sqrt{\frac{1}{N}\sum_{k=1}^{k=N}(\boldsymbol{x}_k^P - \bar{\boldsymbol{x}}_k^P)^2} \tag{18}$$

wherein $\boldsymbol{x}_k^P$ and $\bar{\boldsymbol{x}}_k^P$ are the estimated translation parameters vector in each epoch and their true values, respectively. N is the number of epochs up to the current one. Average RMSE over the MC runs is calculated according to (19).

$$\overline{\boldsymbol{RMSE}_k} = \frac{1}{S}\sum \boldsymbol{RMSE}_k \tag{19}$$

wherein S is the total number of MC runs, which is 1000 in this paper.

As it could be seen from Figure 2, all the three algorithms have a decreasing RMSE pattern over time, which means that the estimations get closer to the ground truth values. Among the three solutions, LKF always delivers the least satisfactory results for both the estimated translations and orientations. The reason is due to the considerably less number of observations that is used in each epoch of this algorithm compared to the other two where the laser scanner measurements are also taken into account. Also, it could be seen that by the suggested PFI methodology, the RMSE of the translation parameters lies in the same range as of the IEKF algorithm. The maximum difference between the estimated translation parameters by IEKF and PFI in this scenario is around 1 meter. Also, the translation parameters by PFI are estimated with a maximum standard deviation of 20 centimeters, which is around 10 and 40 centimeters by the IEKF and LKF algorithms, respectively.

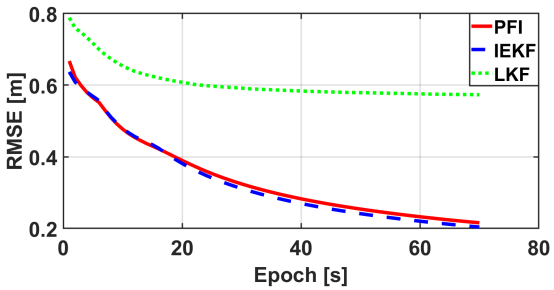

Fig. 2. Mean of RMSE for all translation parameters in all epochs over the total MC runs.

On the other hand according to Figue 3, the RMSE of the orientation parameters shows that PFI delivers solutions closer to the LKF than IEKF. A possible reason for that is claimed to lie in the fusion part of the sensors likelihoods (line 22 of Algorithm 1). Such a fusion without giving weights to the likelihoods of each sensor could lead to resampled particle sets that have an acceptable estimated mean for the translation parameters, but the mean of their orientations diverges from the true values. Moreover, the process noise values usually play a significant role in the prediction step of particle filter. Inappropriate choice of these values could lead to rejection of a large amount of the particles in each epoch and thus deviating from the true values. Therefore, finding a good tune between the weights of the likelihoods from various sensors and the process noise values is claimed to be an important aspect of the PFI algorithm. In the current paper, this aspect is appropriately fulfilled for the translation parameters only. The maximum difference between the estimated orientation parameters by IEKF and PFI in this scenario is around 0.006 radians. Also, the orientation parameters by PFI are estimated with a

maximum standard deviation of 0.001 radians, which is around 0.0008 and 0.002 radians by the IEKF and LKF algorithms, respectively. Combining the IEKF and PFI algorithms could lead to better estimations of the orientation parameters, which is going to be investigated in the future research activities.

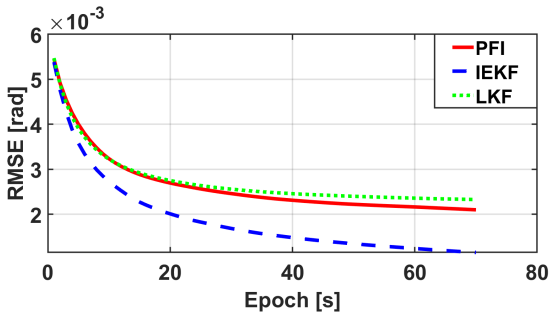

Fig. 3. Mean of RMSE for all orientation parameters in all epochs over the total MC runs.

Furthermore, the estimated parameters are also statistically evaluated over the MC runs. In Figure 4, box plots for the RMSE of translation parameters derived from the PFI are depicted. Top and bottom of each box plot show the 25^{th} and 75^{th} percentiles, respectively and the middle horizontal line of each box plot shows the median. The extended lines above and below each box are the whiskers and the red crosses are the outliers, which are more explained in [7]. It could be seen that the boxes along with their corresponding medians are large in the first epochs and they decrease over time. Having such a decreasing pattern is due to the less number of observations in the beginning, which increases as the UAV enters the space between the buildings (cf. Figure 1). Therefore, the estimated translation parameters are not accurate in the beginning, which leads to larger box plots and vice-versa as more observations are available. Moreover, having the median and mean of RMSE values in the same range and in the middle of the box plots represents a normal distribution with no skewness, which means no systematic errors within the estimations of the filtering algorithm. The same behavior is also seen for the orientation parameters, which is not presented in the current paper.

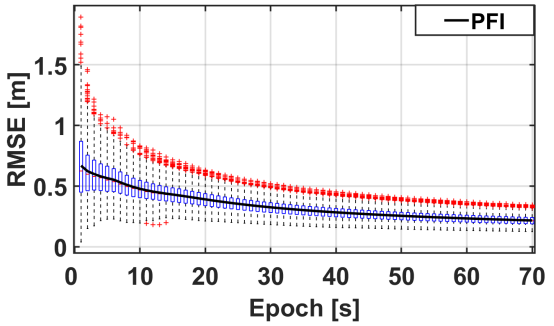

Fig. 4. Box plot of the mean of RMSE for all translation parameters in all epochs over the total MC runs.

Figure 5 shows the time it takes in each epoch for the filters to estimate the states. It could be seen that LKF has the shortest time along all the epochs, which is due to the considerably less number of observations that is used in this filter. As expected, PFI has the longest computation time in all the epochs, which is due to the state estimation principle of particle filter based on a large number of samples. For 70 epochs, the PFI takes approximately 45 seconds to run. In case of the IEKF and LKF algorithms, the total run times are 30 seconds and 0.05 seconds, respectively.

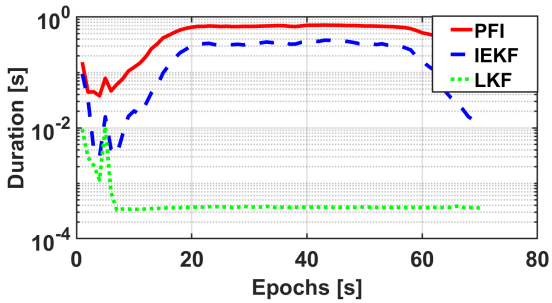

Fig. 5. Duration of the filters.

V. Conclusions and Future Work

The current paper has focused on introducing a newly developed particle filter algorithm, which can handle both implicit and explicit observation models. The introduced methodology is referred to as the PFI. The main idea of this algorithm is to overcome the likelihood estimation difficulty in case of implicit measurement functions by introducing a measure that is taken directly from the observation models, which could be used as the expected values. This new methodology is applied on a simulated environment for georeferencing a UAV that is equipped with a GNSS, an IMU and a 3D laser scanner. To better evaluate the performance of the algorithm, two KF-based methodologies are also applied on the same environment. Results of 1000 MC runs show that the PFI can estimate the translation parameters (position of the MSS) reasonably. However, the estimation of the orientation parameters are shown to have room for improvement. Moreover, the statistical behavior of the estimations of this new methodology are evaluated and shown to be satisfactory. Overall, it is shown that although this new algorithm takes a longer time to run compared to the KF realizations, it can still be beneficial for future applications due to its capability to handle implicit observation models when it comes to the PF framework.

Further research will consider improving the estimation of the orientation parameters by integrating a suitable weighting procedure between likelihood estimations of different sensors. Additionally, the computation time of the PFI is to be improved by combining the current methodology with EKF to increase the quality of the generated samples in each epoch and thus to need less number of particles. Moreover, possibility of including state constraints in the PFI is to be investigated.

Finally, the performance of the methodology in a real case scenario is to be tested and evaluated.

ACKNOWLEDGMENT

This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - NE 1453/5-1 and as part of the Research Training Group i.c.sens [RTG 2159]. The computations were performed by the compute cluster, which is funded by the Leibniz University of Hanover, the Lower Saxony Ministry of Science and Culture (MWK) and DFG.

REFERENCES

[1] L. Gang-jun, Z. Kefei, W. Falin, D. Liam, G. Retscher, "Characterisation of current and future GNSS performance in urban canyons using a high quality 3-D urban model of Melbourne, Australia," Journal of Applied Geodesy, pp. 15–24, 2009.

[2] J. Bureick, S. Vogel, I. Neumann, J. Unger, and H. Alkhatib, "Georeferencing of an unmanned aerial system by means of an iterated extended Kalman filter Using a 3D city model," PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science, vol. 87(5-6), pp. 229–47, December 2019.

[3] N. Bonnor, "Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems," Second EditionPaul D. Groves Artech House, 2013, 776 pp ISBN-13: 978-1-60807-005-3, The Journal of Navigation, vol. 67(1), pp. 191–2, January 2014.

[4] D. Titterton, J.L. Weston, and J. Weston, "Strapdown inertial navigation technology," IET, vol. 17.

[5] O.J. Woodman, "An introduction to inertial navigation," University of Cambridge, Computer Laboratory, No. UCAM-CL-TR-696, 2007.

[6] S. Schön, C. Brenner, H. Alkhatib, M. Coenen, H. Dbouk, N. Garcia-Fernandez, C. Fischer, C. Heipke, K. Lohmann, I. Neumann, U. Nguyen, J.A. Paffenholz, T. Peters, F. Rottensteiner, J. Schachtschneider, M. Sester, L. Sun, S. Vogel, R. Voges, and B. Wagner, "Integrity and collaboration in dynamic sensor networks," Sensors, vol. 18(7):2400, July 2018.

[7] R. Moftizadeh, J. Bureick, S. Vogel, I. Neumann, and H. Alkhatib, "Information-Based Georeferencing by Dual State Iterated Extended Kalman Filter with Implicit Measurement Equations and Nonlinear Geometrical Constraints," 23rd International Conference on Information Fusion (FUSION), pp. 1–9, July 2020.

[8] S. Vogel, H. Alkhatib, and I. Neumann, "Iterated extended Kalman filter with implicit measurement equation and nonlinear constraints for information-based georeferencing," IEEE 21st International Conference on Information Fusion (FUSION), pp. 1209–1216, July 2018.

[9] L. Marchetti, G. Grisetti, and L. Iocchi, "A comparative analysis of particle filter based localization methods," Robot Soccer World Cup, Springer, Berlin, Heidelberg, pp. 442–449, June 2006.

[10] J.J. Leonard and H.F. Durrant-White, "Mobile robot localization by tracking geometric beacons," IEEE Transactions on robotics and Automation, vol. 7(3), pp. 376–382, June 1991.

[11] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," Proc. of the 16th National Conference on Artificial Intelligence (AAAI99), pp. 343–349, July 1999.

[12] S. Schuhmacher, and J. Böhm, "Georeferencing of terrestrial laser scanner data for applications in architectural modeling," 3D-ARCH 2005: Virtual Reconstruction and Visualization of Complex Architectures, XXXVI, PART, vol. 5:W17, 2005.

[13] J.A. Paffenholz, "Direct geo-referencing of 3D point clouds with 3D positioning sensors," München: Verlag der Bayerischen Akademie der Wissenschaften, Reihe C, Heft Nr. 689, 2012.

[14] S. Vogel, H. Alkhatib, and I. Neumann, "Accurate indoor georeferencing with kinematic multi sensor systems," IEEE International conference on indoor positioning and indoor navigation (IPIN), pp. 1-8, October 2016.

[15] J. Talaya, R. Alamus, E. Bosch, A. Serra, W. Kornus, and A. Baron, "Integration of a terrestrial laser scanner with GPS/IMU orientation sensors," Proceedings of the XXth ISPRS Congress, Vol. 35, pp. 1049-1055, July 2004.

[16] A. Soloviev, D. Bates, and F. Van Graas, "Tight coupling of laser scanner and inertial measurements for a fully autonomous relative navigation solution," Navigation, vol. 54(3), pp. 189–205, September 2007.

[17] J. Li-Chee-Ming, and C. Armenakis, "Determination of UAS trajectory in a known environment from FPV video," Proceedings of the UAV-g (Unmanned Aerial Vehicles in Geomatics) Conference, September 2013.

[18] Y. Dehbi, L. Lucks, J. Behmann, L. Klingbeil, and L. Plümer, "Improving GPS trajectories using 3D city models and kinematic point clouds," ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, September 2019.

[19] H. Alkhatib, "Alternative Nonlinear Filtering Techniques in Geodesy for Dual State and Adaptive Parameter Estimation," The 1st International Workshop on the Quality of Geodetic Observation and Monitoring Systems (QuGOMS'11), Springer, Cham, pp. 107–113.

[20] R.E. Kalman, and R.S. Bucy, "New results in linear filtering and prediction theory," pp. 95–108, 1961.

[21] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan, "Estimation with applications to tracking and navigation: theory algorithms and software," John Wiley & Sons, April 2004.

[22] D. Simon, "Optimal state estimation: Kalman, H infinity, and nonlinear approaches," John Wiley & Sons, June 2006.

[23] W.F. Denham, and S. Pines, "Sequential estimation when measurement function nonlinearity is comparable to measurement error," AIAA journal, vol. 4(6), pp. 1071–6, June 1966.

[24] A. Gelb, editor, "Applied optimal estimation," MIT press, 1974.

[25] M. Tailanián, S. Paternain, R. Rosa, and R. Canetti, "Design and implementation of sensor data fusion for an autonomous quadrotor," IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, pp. 1431-1436, May 2014.

[26] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza, "Collaborative monocular SLAM with multiple micro aerial vehicles," IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3962-3970, November 2013.

[27] A. Doucet, N. De Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," Sequential Monte Carlo methods in practice, Springer, New York, NY, pp. 3–14, 2001.

[28] B.M. Bell, F.W. Cathey, "The iterated Kalman filter update as a Gauss-Newton method," IEEE Transactions on Automatic Control, vol. 38(2), pp. 294–7, February 1993.

[29] T. Dang, "An iterative parameter estimation method for observation models with nonlinear constraints," Metrology and Measurement Systems, vol. 15(4), pp. 421–32, 2008.

[30] R. Steffen, "A robust iterative Kalman filter based On implicit measurement equations Robuster iterativer Kalman-Filter mit implizierten Beobachtungsgleichungen," Photogrammetrie-Fernerkundung-Geoinformation, vol. 4, pp. 323–32, August 2013.

[31] S. Vogel, H. Alkhatib, J. Bureick, R. Moftizadeh, and I. Neumann, "Georeferencing of laser scanner-based kinematic multi-sensor systems in the context of iterated extended Kalman filters using geometrical constraints," Sensors, vol. 19(10):2280, January 2019.

[32] S. Vogel, "Kalman Filtering with State Constraints Applied to Multi-sensor Systems and Georeferencing," München: Verlag der Bayerischen Akademie der Wissenschaften, Reihe C, Heft Nr. 856, 2020.

[33] B. Ristic, S. Arulampalam, and N. Gordon, "Beyond the Kalman filter: Particle filters for tracking applications," Artech house, Boston, December 2003.

[34] F. Gustafsson, "Particle filter theory and practice with positioning applications," IEEE Aerospace and Electronic Systems Magazine, vol. 25(7), pp. 53–82. August 2010.

[35] J. González, J.L. Blanco, C. Galindo, A. Ortiz-de-Galisteo, J.A. Fernández-Madrigal, F.A. Moreno, and J.L. Martinez, "Mobile robot localization based on ultra-wide-band ranging: A particle filter approach," Robotics and autonomous systems, vol. 57(5), pp. 496–507, May 2009.

[36] I. Ullah, Y. Shen, X. Su, C. Esposito, and C. Choi, "A localization based on unscented Kalman filter and particle filter localization algorithms," IEEE Access, vol. (8), pp. 2233–46, December 2019.

[37] A.U. Peker, O. Tosun, and T. Acarman, "Particle filter vehicle localization and map-matching using map topology," IEEE Intelligent Vehicles Symposium (IV), pp. 248–253, June 2011.

[38] T. Li, M. Bolic, P.M. Djuric, "Resampling methods for particle filtering: classification, implementation, and strategies," IEEE Signal processing magazine, vol. 32(3), pp. 70-86, April 2015.