Accepted Publication to the 2018 21st International Conference on Intelligent Transportation Systems (ITSC)
November 4-7, 2018, Maui, Hawaii, USA
DOI: 10.1109/ITSC.2018.8569456

Increasing Accuracy in Train Localization Exploiting Track-Geometry Constraints

Hanno Winter*, Volker Willert*, Jürgen Adamy*

***Abstract*— Train-borne localization systems as a key component of future signalling systems are expected to offer huge economic and operational advances for the railway transportation sector. However, the reliable provision of a track-selective and constantly available location information is still unsolved and prevents the introduction of such systems so far. A contribution to overcome this issue is presented here. We show a recursive multistage filtering approach with an increased cross-track positioning accuracy, which is decisive to ensure track-selectivity. This is achieved by exploiting track-geometry constraints known in advance, as there are strict rules for the construction of railway tracks. Additionally, compact geometric track-maps can be extracted during the filtering process which are beneficial for existing train localization approaches. The filter was derived applying approximate Bayesian inference. The geometry constraints are directly incorporated in the filter design, utilizing an interacting multiple model (IMM) filter and extended Kalman filters (EKF). Throughout simulations the performance of the filter is analyzed and discussed thereafter.**

I. Introduction

A reliable location information is indispensable for the safe execution of train rides. As rail guidance and low friction are inherently existent in the railway system it is not safe to operate trains on sight. Instead, the control relies on central instances, taking the location information of all trains into account. For that, train localization is a safety-critical issue within the overall signalling system. At the moment, this localization heavily relies on track-side infrastructure that provides information on whether or not a specific section on the track, which may be several kilometers long, is occupied by a train. This localization principle has proven to be successful, although, by design it suffers either from high costs or a low track capacity [1]. For this reason and because of advances in sensor technology, especially in the development of micro-electro-mechanical systems (MEMS) inertial measurement units (IMU) and the further deployment of global navigation satellite systems (GNSS), the development of train-borne localization systems has gained more interest in recent years [2]. One of the main challenges for train-borne localization systems is to ensure a track selective localization result on parallel tracks, which are typically installed in a distance of $4\,\mathrm{m}$. In addition, the functionality of such a system has to be demonstrated in the sense of EN 50126, as it is highly safety-critical. Therefore, it has to fulfill the highest standards in terms of reliability, availability, maintainability and safety (RAMS) [3]. These two circumstances ultimately lead to the reason why there is no market-ready solution available so far.

*Control Methods and Robotics, TU Darmstadt, Germany

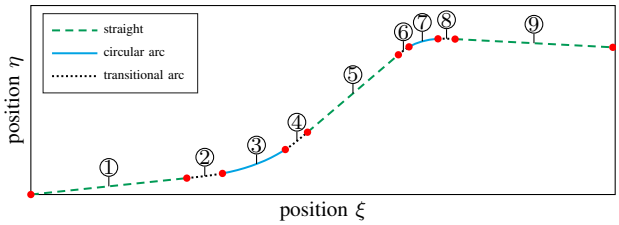

Fig. 1. Exemplary track, consisting of the three standard track geometries: straight, transitional arc and circular arc. The individual parameters for the track elements $1-9$ are listed in Tab. I.

Here we aim at presenting a new train localization approach exploiting track-geometry constraints to increase the positioning accuracy. We utilize the fact that railway tracks can only consist of a sequence of the following three geometric shapes: straight, transitional arc and circular arc as demonstrated in Fig. 1 [4]. By continuously identifying the current track geometry and incorporating this knowledge in the estimation process of the train's position, we will demonstrate that especially the cross-track positioning accuracy can be increased significantly. In addition, the identified track-geometry parameters can be used as a compact geometric track-map which can be utilized by several already existing train localization algorithms. According to our best knowledge, this is the first approach which explicitly takes the sequence and shape of different track geometries into account, although the idea of exploiting special railway features to achieve better positioning results has been known before. The feature probably taken advantage of most often, is the fact that the position of a railway vehicle is constrained to its track. Thus, if a track map is available, the 2-D localization problem can be reduced to a 1-D problem. For example, in [5] a map-matching algorithm solely based on GNSS measurements is presented. However, as shown in [6] better performances can be achieved by explicitly considering the track constraints in the filter design, as it is done in [7]–[9] in different ways. The track constraints also can be exploited to improve GNSS performance, both, in the sense of the absolute positioning accuracy and in terms of integrity monitoring [10]–[12].

It is important to notice the different meanings of *track-constraint*, used in previous literature, and *track-geometry constraint* that we apply here. The former relates to exploiting a given track map to constrain possible positioning solutions to fixed tracks, while the latter is related to our idea of going one step further by directly incorporating track geometry constraints into the position calculation.

TABLE I

PARAMETERS OF THE TRACK SHOWN IN FIG. 1.

Track element	①	②	③	④	⑤	⑥	⑦	⑧	⑨
Shape[a]	st	ta	ca	ta	st	ta	ca	ta	st
Length L in m	1000	231	476	231	1000	108	206	108	1000
Length L_{start} in m	0	1000	1231	1707	1938	2938	3046	3252	3360
Radius r in m	∞	—	900	—	∞	—	300	—	∞
Speed v_{max} in km/h	160	130	130	130	160	75	75	75	160

[a] st: straight, ta: transitional arc, ca: circular arc

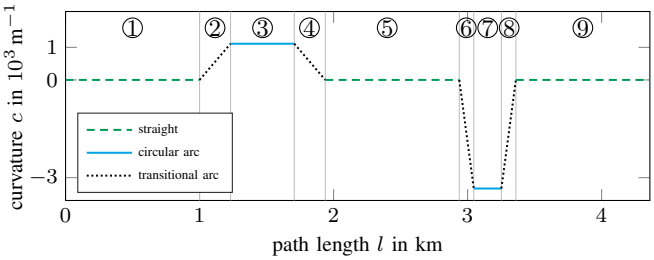

Fig. 2. Curvature profile c of the track depicted in Fig. 1 plotted against the track's path length l.

II. CHARACTERISTICS OF RAILWAY TRACKS

In order to better understand the train localization algorithm presented later, we start with a brief survey of the most important features of railway tracks, based on [4].

A. Construction Principles

Railway tracks should be constructed in a way to ensure a safe and comfortable journey at the highest speed possible. Therefore, one of the main objectives is to minimize the occurring accelerations, as they are perceived as unpleasant by the passengers very quickly. Great changes in acceleration occur e. g. when changing from a straight line to a circular arc where the radius changes abruptly. To mitigate such unwanted effects there are clear rules that regulate the sequence and the realization of certain track geometries.

B. Standard Track Geometries

Typical railway tracks only consist of the following three geometric shapes: straight, transitional arc and circular arc. This sequence is fixed when entering a curve and repeated the other way round when the curve is left again. The transitional arc ensures a smooth transition from a straight line with radius $r_{\text{st}} = \infty$ to the circular arc's radius $r_{\text{ca}} > 0$ to reduce the before mentioned impact when entering or leaving a curve. By additionally elevating the outer rail in curves the comfort and maximum curve speed can be further increased.

C. Curvature Profile

In the railway domain it is convenient to use the curvature $c = \frac{1}{r}$ instead of the radius r as design parameter, as the radius often appears in the denominator throughout many calculations.

In Fig. 2 the curvature profile corresponding to the track shown in Fig. 1 is drawn against the path length l of the track. The track consists of a total of nine elements, as numbered in the figures, realizing two turns with different lengths and radii, according to the scheme described above. The exact track parameters can be found in Tab. I. From the linear curvature transitions in the sections 2, 4, 6 and 8, it can be seen that the transitional curve is realized by a clothoid, which is the most common case.

III. PROBABILISTIC PROBLEM FORMULATION

Train-borne localization requires a strict probabilistic treatment of all the underlying data to be able to appropriately cope with uncertainties [6]. For that a dynamic Bayesian network (DBN) will be specified to gain more insight into the localization problem at hand. Based on the DBN, a general posterior can be formulated, serving as basis for the development of a Bayesian filter. However, here we want to propose an alternative filtering approach by slicing up the original DBN into three successive DBNs, inspired by the structure of the underlying model. Thus, our filter is derived by approximately applying Bayesian inference. The consequences will be discussed in due course.

A. Dynamic Bayesian Network

A DBN visualizes the dependencies of multiple random variables over time. Thereby, it represents the preliminary assumptions on how a considered joint probability density function (pdf) can be factorized into multiple conditional distributions [13]. The DBN related to the problem at hand is drawn on the left in Fig. 3. It represents a first-order hidden Markov model, with the hidden train states $\boldsymbol{x}$ and the observable measurements $\boldsymbol{z}$ extended by a mode switching variable $\boldsymbol{M}$ and the track-geometry parameters $\boldsymbol{g}$. The mode switching variable $\boldsymbol{M} \in \{\boldsymbol{M}_i\}_{i=1}^{r}$ allows different models to be applied, each matched to a special track geometry, for the transition of the train states and the geometry parameters. By separately modeling the train states and the track-geometry parameters the connection between the two becomes obvious. As the track geometry can be seen as the given source of the train's motional behavior, we choose $\boldsymbol{x}$ to be dependent on $\boldsymbol{g}$. It is also conceivable to model this dependability the other way round, although, the physical explanation then becomes less intuitive. In fact, our modeling represents the main idea of our localization algorithm, that is, to improve the position estimate by taking knowledge of the track geometry into account explicitly.

B. General Posterior

The inference task that needs to be solved is to estimate the joint pdf $p(\boldsymbol{x}_k, \boldsymbol{M}_k, \boldsymbol{g}_k | \boldsymbol{z}_{1:k})$[1]. Through marginalization

[1] The subscripts indicate the discrete time or time span we refer to, e. g. by writing $\boldsymbol{z}_{1:k}$ we refer to all measurements in the time span from $t = T$ to $t = kT$, with sampling time T and $k \in \mathbb{Z}$.

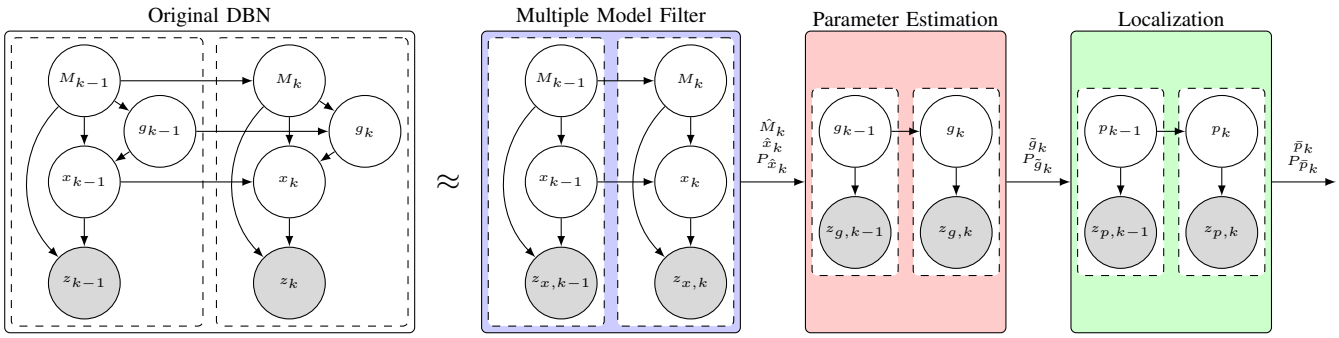

Fig. 3. Approximation scheme for the posterior (1) by slicing the original DBN into structural sub-DBNs, enabling the implementation of three successive filters. The estimation results $\mathbb{E}[\bullet]$ of the individual filter stages are marked by different superscripts, e. g. a quantity q estimated in the first stage is marked by $\hat{q}$, in the second by $\tilde{q}$ and in the third by $\bar{q}$.

and application of the product rule, the final posterior

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) = \sum_{i=1}^{r} \int_{\boldsymbol{g}_k} \underbrace{p(\boldsymbol{x}_k, \boldsymbol{g}_k|\boldsymbol{M}_{i,k}, \boldsymbol{z}_{1:k}) p(\boldsymbol{M}_{i,k}|\boldsymbol{z}_{1:k})}_{\substack{\text{“multiple model estimation”} \\ + \\ \text{“parameter estimation”}}} d\boldsymbol{g}_k$$
$$= \sum_{i=1}^{r} \int_{\boldsymbol{g}_k} p(\boldsymbol{x}_k|\boldsymbol{M}_{i,k}, \boldsymbol{g}_k, \boldsymbol{z}_{1:k}) p(\boldsymbol{M}_{i,k}|\boldsymbol{z}_{1:k}) \times p(\boldsymbol{g}_k|\boldsymbol{M}_{i,k}, \boldsymbol{z}_{1:k})\, d\boldsymbol{g}_k \tag{1}$$

can be extracted. For a Bayesian filter implementation the posterior (1) has to be factorized in a recursive form, considering the structure of the DBN. In this work an alternative approach, approximating the posterior by three successive filters, is proposed.

C. Posterior Approximation

By investigating the DBN's or the posterior's (1) structure it becomes obvious that we have to deal with a multiple-model problem, extended by an underlying parameter estimation process [14]. That is what motivates us to propose an approximate solution for (1) by a structure of three successive filters. Therefore, the original DBN is split up into three DBNs as shown on the right in Fig. 3, which can be calculated successively by separate filters. The first DBN (shaded blue) represents the multiple-model structure induced by the different track geometries, where the mode switching behavior is now only taken into account for the train states $\boldsymbol{x}$. In this stage the underlying mode (track geometry) and the train's motional states are to be estimated, given the sensor data $\boldsymbol{z}_x$, simultaneously. Then (shaded red) the track-geometry parameters are estimated for the most likely mode $\hat{\boldsymbol{M}} = \arg\max_M \left(p(\boldsymbol{M}|\boldsymbol{z}_x)\right)$, assuming a first-order hidden Markov model, at which the estimated train states $\hat{\boldsymbol{x}}$ serve as measurement data $\boldsymbol{z}_g = \text{subset}(\hat{\boldsymbol{x}})$. Finally, in the third stage (shaded green), again assuming a first-order hidden Markov model, the position $\bar{\boldsymbol{p}}$ of the train is calculated. Therefore, the previously estimated parameters $\tilde{\boldsymbol{g}}$ are used in the transition model from $\boldsymbol{p}_{k-1}$ to $\boldsymbol{p}_k$ and the train states serve as measurement data $\boldsymbol{z}_p = \text{subset}(\hat{\boldsymbol{x}})$ again.

Slicing up the originally presented DBN as described before, and only propagating the most probable track geometry $\hat{\boldsymbol{M}}$, makes it possible to use well-known filter techniques in each step. However, this approach is not exactly relying on Bayesian inference. For example, this can be seen from the fact that the knowledge about the parameters is not used to improve the current mode or train state estimation process in turn within each time slice.

IV. Localization Filter Implementation

In this section the implementation of the filter approach presented in Sec. III-C will be explained. In doing so, we start from the following system specifications: For the moment, planar single tracks without any branching tracks will be assumed. The track is unknown, hence, there is no track map given in advance. Furthermore, a GNSS receiver and an IMU are assumed to be the sensors available, providing the position $\boldsymbol{p}_{\text{GNSS}} = (\xi_{\text{GNSS}},\ \eta_{\text{GNSS}})^T$ in the navigation plane as well as the along-track acceleration a_{IMU} and the yaw rate w_{IMU} of the train relative to the navigation coordinate system. The definitions of the various coordinate systems together with the train's track are shown in Fig. 4.

A. Track-Geometry Extraction

In the first stage the track geometry as well as the train's motional states have to be estimated (c. f. Fig. 3, blue

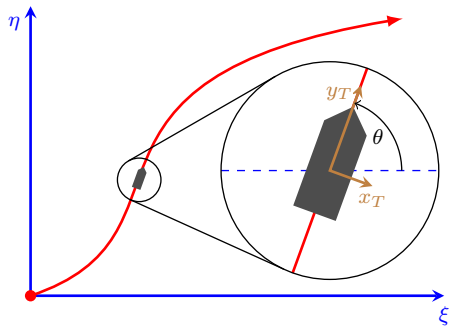

Fig. 4. Definition of the navigation coordinate system (blue) and the train coordinate system (brown). θ is the train's orientation angle, measured between the ξ-axis and the velocity vector in along-track direction. The track is shown in red. The dot marks the track's starting point ($l = 0$) and the arrow the direction of positive counting.

shaded area). As mentioned before, this relates to a multiple-model problem whose optimal solution is intractable. This is because it requires to match a filter to each possible mode history at each time step, leading to r^k filters at time step k, for example. There are different suboptimal approaches from which we chose the interacting multiple model (IMM) filter. It is characterized by its reasonable computational prerequisites, but still yielding adequate results compared to other approaches [14].

As typical tracks consist of straight and circular arc elements most of the time, we will focus on identifying these two. Therefore, at least two models are needed to distinguish between the linear and the circular motion, respectively. A third fall-back model is supplemented additionally to make the distinction even more accurate by covering clutter situations where a precise assignment of the two models mentioned before is not justified, e. g. on transitional arcs. For each model $\boldsymbol{M}_i$ a separate filter has to be set up, solving for the train's motional states $\boldsymbol{x}$ given the measurements $\boldsymbol{z}_x$. We will use an extended Kalman filter (EKF) for each model with $\boldsymbol{x} = (\,\xi,\ \eta,\ d,\ v,\ a,\ \theta,\ w\,)^T$ and $\boldsymbol{z}_x = (\,\xi_{\text{GNSS}},\ \eta_{\text{GNSS}},\ a_{\text{IMU}},\ w_{\text{IMU}}\,)^T$, in which d represents the traveled distance from the start, v the train's speed and θ the train's orientation angle, measured between the ξ-axis and the velocity vector in along-track direction. The various discrete time, dynamic stochastic system equations $\boldsymbol{x}_{k+1} = \boldsymbol{f}_{x,i}(\boldsymbol{x}_k) + \boldsymbol{\Gamma}\boldsymbol{v}_{x,i,k}$ with sampling time T, zero-mean, white Gaussian noise vectors $\boldsymbol{v}_{x,i,k} \sim \mathcal{N}(\boldsymbol{0}, diag(\sigma^2_{i,\text{ACC}},\, \sigma^2_{i,\text{TR}}))$ and their corresponding vector gain

$$\Gamma = \begin{pmatrix} \frac{1}{2}T^2 & \frac{1}{2}T^2 & \frac{1}{2}T^2 & T & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & T & 1 \end{pmatrix}^T \tag{2}$$

are given below. The standard deviations $\sigma_{i,\text{ACC}}$ and $\sigma_{i,\text{TR}}$ are design parameters, representing the acceleration increment and the turn rate increment over one sampling period that should be taken into account in the i-th model, respectively [14].

1) Straight-track model: While the train is moving on a straight track segment its orientation θ stays constant and its motion can be modeled by assuming constant acceleration (CA) [15], which results in

$$\boldsymbol{f}_{x,1} = \begin{pmatrix} \xi_k + v_k T cos(\theta_k) + \frac{1}{2}a_k T^2 cos(\theta_k) \\ \eta_k + v_k T sin(\theta_k) + \frac{1}{2}a_k T^2 sin(\theta_k) \\ d_k + v_k T + \frac{1}{2}a_k T^2 \\ v_k + a_k T \\ a_k \\ \theta_k \\ 0 \end{pmatrix}. \tag{3}$$

2) Circular-arc track model: Extending the former model with an additional constant turn rate assumption, altogether known as constant turn rate and acceleration (CTRA) assumption [15], the movement on a circular arc can be described by[2]

$$\boldsymbol{f}_{x,2} = \begin{pmatrix} \xi_k + \Delta\xi_k \\ \eta_k + \Delta\eta_k \\ d_k + v_k T + \frac{1}{2}a_k T^2 \\ v_k + a_k T \\ a_k \\ \theta_k + w_k T \\ w_k \end{pmatrix}, \tag{4}$$

with

$$\begin{aligned} *\Delta\xi_k = \tfrac{1}{w_k^2}\,[&(v_k\omega_k + a_k\omega_k T)\, sin(\theta_k + \omega_k T) && (5)\\ &+ a_k cos(\theta_k + w_k T) && (6)\\ &- v_k\omega_k sin(\theta_k) - a_k cos(\theta_k)]\,, && (7)\\ \Delta\eta_k = \tfrac{1}{w_k^2}\,[&(-v_k\omega_k - a_k\omega_k T)\, cos(\theta_k + \omega_k T) && (8)\\ &+ a_k sin(\theta_k + w_k T) && (9)\\ &+ v_k\omega_k cos(\theta_k) - a_k sin(\theta_k)]\,. \end{aligned}$$

3) Fall-back model: The fall-back model has the same structure as the circular-arc model

$$\boldsymbol{f}_{x,3}(\boldsymbol{x}_k) = \boldsymbol{f}_{x,2}(\boldsymbol{x}_k) \tag{10}$$

because the latter is already capable of describing all occurring types of motion. The desired fall-back behavior is achieved by applying a much larger process noise $\boldsymbol{v}_{x,3}$ compared to the other two models, through the choice of $\sigma_{3,\text{TR}} \gg \sigma_{\{1,2\},\text{TR}}$.

The common measurement model is given by

$$\boldsymbol{z}_{x,k} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \boldsymbol{x}_k + \boldsymbol{w}_{x,k}, \tag{11}$$

with the zero-mean, white Gaussian noise vector $\boldsymbol{w}_{x,k} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_{x,k})$ and its corresponding measurement covariance matrix $\boldsymbol{R}_{x,k}$.

Once an EKF is set up for each model, only the time-invariant model transition probability matrix $\boldsymbol{\Pi}$, defining the probabilities p_{ij} of a transition from model i to model j, has to be chosen in order to be able to implement the IMM filter. The various p_{ij} are assumed to be known in advance and represent design parameters of the IMM filter. For the concrete implementation of the EKF's and the IMM's algorithms, respectively, we would like to refer to the literature, e. g. [14].

B. Parameter Extraction

The next step is to identify the track-geometry parameters (c. f. Fig. 3, red shaded area) for the most likely model $\hat{\boldsymbol{M}}_k = \arg\max_{M_k}\left(p(\boldsymbol{M}_k|\boldsymbol{z}_{x,1:k})\right)$ calculated before. The estimation task can be formulated in the form $\boldsymbol{z}_g = \boldsymbol{h}_g(\boldsymbol{g}) + \boldsymbol{w}_g$ with observations $\boldsymbol{z}_g$, nonlinear measurement function $\boldsymbol{h}_g$, desired static parameters $\boldsymbol{g}$, and $\boldsymbol{w}_g \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_g)$. This problem can

[2]In the case of very small turn rates w, we switch from the CTRA model to the CA model to avoid numerical inconsistencies.

be interpreted as a nonlinear least squares (LS) problem or a filtering task with a static system function and without any process noise. In the sense of the latter, we again implement different EKFs for each track geometry [16]. Thereby, some of the previously calculated train's motional states serve as observation data $\boldsymbol{z}_{g,k} = \text{subset}(\hat{\boldsymbol{x}}_k)$. The parameters to be estimated for each track geometry are its starting point $(\,\xi_0,\ \eta_0\,)^T$, its tangent angle φ_0 at the starting point and its radius[3] r, summarized in $\boldsymbol{g} = (\,\xi_0,\ \eta_0,\ \varphi_0,\ r\,)^T$. The measurement models $\boldsymbol{z}_{g,\{1,2\},k}$ for the two track geometries we are considering are defined as:

1) Straight line

$$\begin{pmatrix} \hat{\xi}_k \\ \hat{\eta}_k \end{pmatrix} = \begin{pmatrix} \hat{l}_k\, cos(\varphi_{0,k}) + \xi_{0,k} \\ \hat{l}_k\, sin(\varphi_{0,k}) + \eta_{0,k} \end{pmatrix} \tag{12}$$

2) Circular arc

$$\begin{pmatrix} \hat{\xi}_k \\ \hat{\eta}_k \\ \hat{v}_k \end{pmatrix} = \begin{pmatrix} r_k\, sin(\frac{\hat{l}_k}{r_k} + \varphi_{0,k}) + \xi_{0,k} \\ r_k\, \left[1 - cos(\frac{\hat{l}_k}{r_k} + \varphi_{0,k})\right] + \eta_{0,k} \\ \hat{w}_k r_k \end{pmatrix} \tag{13}$$

Both models not only include quantities from $\boldsymbol{g}_k$ exclusively but also from $\hat{\boldsymbol{x}}_k$, in which $\hat{l}_k$, representing the current path length, is an operand calculated through $\hat{l}_k = \hat{d}_k - \hat{d}_{\text{start}}$. The quantity $\hat{d}_{\text{start}}$ is always saved when a new track geometry is detected. For all quantities originating from $\hat{\boldsymbol{x}}_k$, the corresponding entries in the covariance matrices $\boldsymbol{P}_{\tilde{g},k}$ and $\boldsymbol{R}_{g,k}$ are induced by their equivalents in the covariance matrix $\boldsymbol{P}_{\hat{x}_k}$, in each filtering step. In the case of $\hat{l}_k$ the variance is induced by $var(\hat{l}_k) = var(\hat{d}_k) + var(\hat{d}_{\text{start}})$. For the initialization of the filters, the current estimates of $\hat{\xi}$, $\hat{\eta}$ and $\hat{\theta}$[4], together with their corresponding blocks in the covariance matrix $\boldsymbol{P}_{\hat{x}}$ can be used directly, except for r. The radius is initialized through one LS calculation step for $\hat{v} = \hat{\omega} r$, to get an initial estimate as well as a variance [14].

C. Localization

The last step is to calculate the final position estimate $\bar{\boldsymbol{p}}$ together with its covariance matrix $\boldsymbol{P}_{\bar{p}}$ (c. f. Fig. 3, green shaded area). In the case that the current track geometry and its parameters have been identified in the previous steps, the position can be calculated by $\bar{\boldsymbol{p}} = \boldsymbol{Tr}(\hat{l}, \tilde{\boldsymbol{g}})$. Thereby, $\boldsymbol{Tr}$ represents the current track in a parametric form with respect to $\hat{l}$. Since a track geometry $\boldsymbol{Tr}$ can not always be identified (c. f. Sec. IV-A) and also the covariance matrix $\boldsymbol{P}_{\bar{p}}$ is required, an estimation problem, delivering an estimate for the position and its covariance matrix, has to be solved again. Therefore, we implement different EKFs, dealing with each possible situation: 1) straight track identified 2) circular-arc track identified and 3) unknown track geometry.

The state vector for the different filters is defined as $\boldsymbol{p} = \left(\,\xi,\ \eta,\ \xi_0,\ \eta_0,\ \varphi_0,\ r,\ \hat{l}\,\right)^T$, while the estimates $\hat{\xi}$ and $\hat{\eta}$ from the first filter stage form the measurement vector $\boldsymbol{z}_p = \left(\,\hat{\xi},\ \hat{\eta}\,\right)^T$. Except for ξ and η, the necessary blocks in the covariance matrices $\boldsymbol{P}_p$ and $\boldsymbol{R}_p$ are taken directly from their corresponding blocks in $\boldsymbol{P}_{\hat{x}}$ and $\boldsymbol{P}_{\tilde{g}}$ at each time step. For the two former mentioned, the covariance matrix is only initialized through $\boldsymbol{P}_{\hat{x}}$ and then computed as usual in a KF. Furthermore, no additional process noise $\boldsymbol{v}_p$ is included because the purpose of this localization stage is the fusion of all available information and no additional uncertainties should be introduced.

The system models $\boldsymbol{f}_{p,i}(\boldsymbol{p}_k)$ for each situation are given below.

1) Straight-track localization:

$$\boldsymbol{f}_{p,1} = \begin{pmatrix} \hat{l}_k cos(\tilde{\varphi}_{0,k}) + \tilde{\xi}_{0,k} \\ \hat{l}_k sin(\tilde{\varphi}_{0,k}) + \tilde{\eta}_{0,k} \\ \tilde{\xi}_{0,k} \\ \tilde{\eta}_{0,k} \\ \tilde{\varphi}_{0,k} \\ 0 \\ \hat{l}_k \end{pmatrix} \tag{14}$$

2) Circular-arc track localization:

$$\boldsymbol{f}_{p,2} = \begin{pmatrix} \tilde{r}_k\, sin(\frac{\hat{l}_k}{\tilde{r}_k} + \tilde{\varphi}_{0,k}) + \tilde{\xi}_{0,k} \\ \tilde{r}_k\, \left[1 - cos(\frac{\hat{l}_k}{\tilde{r}_k} + \tilde{\varphi}_{0,k})\right] + \tilde{\eta}_{0,k} \\ \tilde{\xi}_{0,k} \\ \tilde{\eta}_{0,k} \\ \tilde{\varphi}_{0,k} \\ \tilde{r}_k \\ \hat{l}_k \end{pmatrix} \tag{15}$$

3) Unknown-Track Localization: If the track geometry is unknown, the first estimates of $\hat{\xi}_k$ and $\hat{\eta}_k$ together with their corresponding covariance block in $\boldsymbol{P}_{\hat{x}}$ are adopted directly.

All states are initialized through their corresponding estimates in $\hat{x}$ or $\tilde{g}$, respectively. Finally, the position estimate $\bar{\boldsymbol{p}} = (\,\bar{\xi},\ \bar{\eta}\,)^T$ and its covariance matrix $\boldsymbol{P}_{\bar{p}}$ can be gathered from the first two states of $\boldsymbol{p}$ and its corresponding block in $\boldsymbol{P}_p$.

V. Analysis

The algorithm presented above will be analyzed in this section, based on simulations. To begin with, the investigated set-up is explained. Then the individual steps of the algorithm are evaluated and its performances discussed, thereafter.

A. Initialization

Throughout all simulations, the track described in Tab. I and shown in Fig. 1 serves as test track. It represents a typical main line with minimal radii of $300\,\text{m}$ and is probably installed in a way like this in many real railway networks. In all simulations the train starts at time $t_0 = 0\,\text{s}$ on the track with path length $l_0 = 0\,\text{m}$ and speed $v = 70\,\text{km/h}$, which is not violating any speed limits on the given track. As the train moves along the track during a simulation run all reference data (ground truth) can be calculated directly and the artificial sensor data

[3] We use $r = 0$ for straight track geometries.

[4] The tangent angle φ_0 is initialized through $\hat{\theta}$.

TABLE II

SIMULATION NOISE PARAMETERS

	T	$\sigma_{i,\text{noise}}$
Sensors	0.05 s	$\sigma_{\text{GNSS}} = 10\,\text{m}$ $\sigma_{\text{ACC}} = 0.001\,\text{g}$ $\sigma_{\text{GYRO}} = 0.05\,\text{deg/s}$
$\text{EKF}_{1\ldots3}$	0.50 s	$\sigma_{1\ldots3,\text{ACC}} = 0.750\,\text{m/s}^2$ $\sigma_{1\ldots2,\text{TR}} = 0.014\,\text{deg/s}$ $\sigma_{3,\text{TR}} = 0.140\,\text{deg/s}$

is generated by adding zero-mean, white Gaussian noise with standard deviations as listed in Tab. II.

To start the localization algorithm described in Sec. IV the three EKFs used in the IMM filter have to be initialized (c. f. Sec. IV-A). For all three, the state vector $\boldsymbol{x}_0$ is chosen in a way so that all not measured quantities are initialized with their ground truth value at t_0, respectively. The initial covariance matrix $\boldsymbol{P}_{x_0}$ for each EKF_i is chosen as $\boldsymbol{P}_{x_0,i} = \boldsymbol{\Gamma}\, diag(\sigma^2_{i,\text{ACC}}, \sigma^2_{i,\text{TR}})\, \boldsymbol{\Gamma}^T$, in which the corresponding noise parameters $\sigma_{i,\text{noise}}$ are listed in Tab. II. The IMU's noise parameters are taken from [17] and also roughly agree with the values presented in [18], while the GNSS's noise is chosen freely and considered to be reasonable in most situations. Altogether, these values represent a sensor system with a rather average performance, affordable in a low to mid cost range set-up. Eventually, the IMM filter is started with the mode probabilities $\boldsymbol{\mu}_0 = (\,0.90,\ 0.05,\ 0.05\,)^T$, favoring the straight-track model at the beginning, and the transition probability matrix

$$\boldsymbol{\Pi} = \begin{pmatrix} 0.990 & 0.000 & 0.010 \\ 0.000 & 0.990 & 0.010 \\ 0.005 & 0.005 & 0.990 \end{pmatrix}.$$

By the choice of $\boldsymbol{\Pi}$ it can be seen how the knowledge of the real world can be incorporated into the IMM filter. As the high values on the diagonal indicate that it is more likely to stick to the current track geometry than switching to another. Furthermore, the zero entries indicate that it is impossible to directly switch from a straight track-geometry to a circular one, and the other way round.

B. Evaluation

Below, the individual steps of the track-geometry constrained (TGC[5]) localization filter are evaluated.

1) Track-Geometry Extraction: The track geometry detection is evaluated by means of the mode probabilities μ_i, which are issued by the IMM filter, as presented in Fig. 5. As the mode probability for the current track geometry quickly approaches a value close to 1 and all others close to 0, it can be seen that a clear distinction between the different track geometries can be made and also the right track geometry is detected for each track element 1 – 9 (c. f. Tab. I). The delays until the right track geometries are detected $\Delta L_{\text{start}} = L_{\text{start,TGC}} - L_{\text{start,ref}}$ in meters are shown in Tab. III.

[5]The abbreviation TGC will be used throughout the evaluation process to mark the approach presented here.

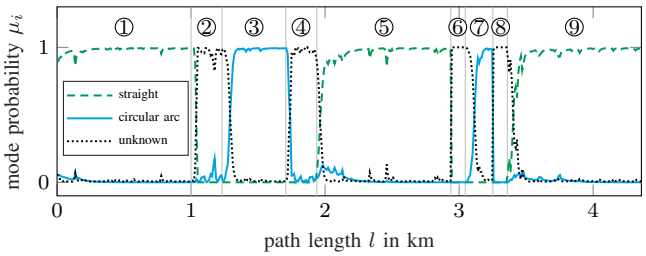

Fig. 5. Mode probabilities μ_i for the various track geometries plotted against the path length l. The grid (gray lines) indicates the true changing points of the reference-track's elements 1–9 (c. f. Tab. I).

TABLE III

GEOMETRY DETECTION DELAYS $\Delta L_{\text{start}} = L_{\text{start,TGC}} - L_{\text{start,ref}}$

Track element	①	②	③	④	⑤	⑥	⑦	⑧	⑨
ΔL_{start} in m	0	40	62	33	26	8	75	5	52

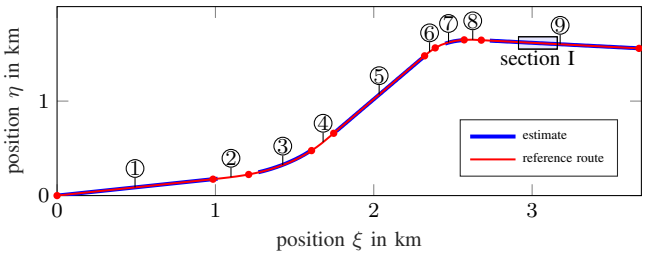

Fig. 6. Estimated track geometries plotted against the reference track. The marked section (blue shaded rectangle) is shown in more detail in Fig. 11.

In this case the maximum delay is 75 m at the detection of the 7th track element.

2) Parameter Extraction: In Fig. 6 the identified track geometries are plotted together with the reference track. As only straight and circular arc track geometries are identified there are no estimates for the transitioning elements 2, 4, 6 and 8. Considered qualitatively, all others are estimated quite well, although the detection delays mentioned before are not corrected in the estimate of the starting points of the track elements, which can especially be seen for the elements 3, 7 and 9. A quantitative assessment is included in the next evaluation step.

3) Localization: Finally, the overall localization performance is evaluated. Therefore, the proposed TGC localization filter is compared to a standard fusion approach with an EKF[6] and the estimates of the IMM filter in the first filter stage (c. f. Sec. IV-A). The absolute positioning error $\|\Delta\boldsymbol{p}\| = \|\boldsymbol{p}_{\text{ref}} - \boldsymbol{p}_{\text{filter}}\|$ is depicted in Fig. 7. All three methods perform in the same order of magnitude, which is also confirmed by calculating the mean $2.21 \pm 0.07\,\text{m}$ and the standard deviation $1.21 \pm 0.02\,\text{m}$ of the absolute positioning error for all three methods, as there are no significant variations.

At least as important as the absolute position accuracy, is the assessment of its uncertainty, because only by that a safe and distinct assignment to a specific track can be performed and thus a track-selective location information provided.

[6]This EKF is initialized like the one matched to the fall-back model (c. f. Sec. V-A).

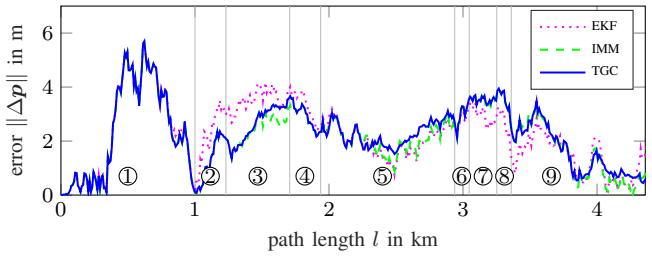

Fig. 7. Positioning error $\|\Delta \boldsymbol{p}\| = \|\boldsymbol{p}_{\text{ref}} - \boldsymbol{p}_{\text{filter}}\|$ plotted against the path length l. The grid (gray lines) indicates the changing points of the reference-track's elements 1–9 (c. f. Tab. I).

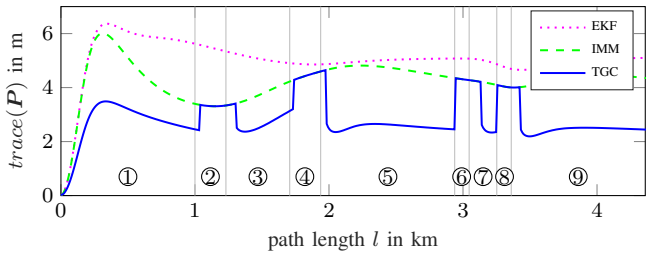

Fig. 8. Positioning uncertainty, expressed by $trace(\boldsymbol{P})$, plotted against the path length l. The grid (gray lines) indicates the changing points of the reference-track's elements 1–9 (c. f. Tab. I).

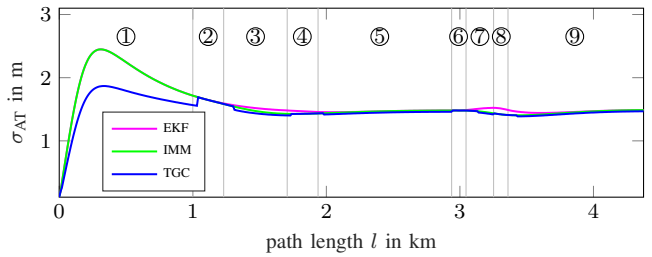

Fig. 9. Along-track uncertainty σ_{AT} plotted against the path length l. The grid (gray lines) indicates the changing points of the reference-track's elements 1–9 (c. f. Tab. I).

The uncertainty of the position estimates is investigated by means of the trace of the correspondingly estimated covariance matrices, as shown in Fig. 8. The trace of the position covariance matrix represents the overall expected mean squared error of the position estimate and is therefore a common measure for the positioning uncertainty [19]. Initially, the uncertainty in all three methods increases quickly and comes to a level where it is only affected by slight changes, which is expectable as the process and measurement noise are static [14]. However, there is a noticeable difference in how big the individual uncertainties can become. In case a track geometry has been identified (track elements: 1, 3, 5, 7, 9) the TGC approach outperforms the other two by a factor of approximately two, while otherwise the performance degrades to the IMM's one (track elements: 2, 4, 6, 8). This makes sense because in the latter case the result of the IMM filter stage is adopted in the localization stage (c. f. Sec. IV-C). Furthermore, the IMM filter performs a little better than the standard EKF because of its faster adaptability to different track geometries.

The uncertainty is investigated further to show more clearly

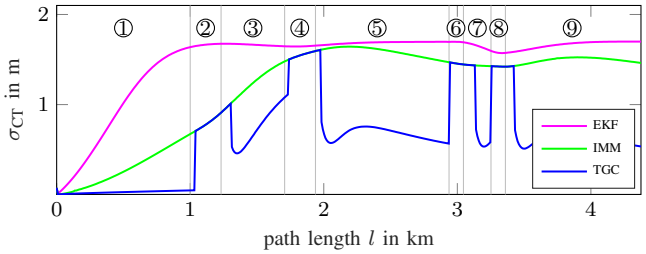

Fig. 10. Cross-track uncertainty σ_{CT} plotted against the path length l. The grid (gray lines) indicates the changing points of the reference-track's elements 1–9 (c. f. Tab. I).

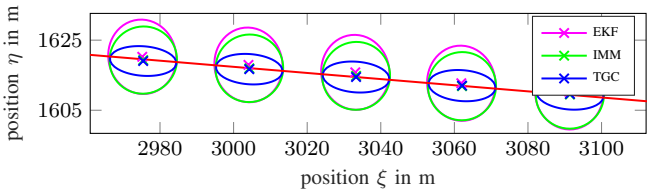

Fig. 11. Positioning results with corresponding error ellipses (with a confidence region of 99.9999999 % in the style of a safety integrity level 4), for the area marked in Fig. 6 as "section I".

in which way the uncertainty in the TGC approach is reduced. Therefore, the standard deviation in along-track σ_{AT} and cross-track σ_{CT} direction is calculated from the covariance matrices, respectively, as shown in Fig. 9 and Fig. 10 [20]. Compared to the other two solutions the TGC approach shows a significantly smaller cross-track deviation when the track-geometry has been identified. Therefore, a better track accuracy can be ensured in these cases. This result also can be seen by comparing the size and shape of the positioning error ellipses in Fig. 11, which shows the section at the end of the track, marked as "section I" in Fig. 6.

C. Discussion

As shown in the previous evaluation, the presented TGC approach leads to a significant improvement in the cross-track positioning accuracy when the current track geometry is involved, confirming our initial notion. Some details and limitations will be discussed here.

1) Track-geometry detection performance: The detection of the current track geometries heavily depends on the performance of the utilized sensors and the train's speed, as the differentiation basically relies on the extraction of the curvature profile of the track. This problem is also addressed in another context in [7] and [17]. To ensure a consistent behavior under all conditions and also a more direct and precise track-geometry identification, further adoptions need to be implemented. Possible actions are e. g. a more sophisticated tuning of the IMM filter, the consideration of additional track characteristics (elevation and slope), the integration of additional track-geometry models and the use of additional sensors.

2) Track-geometry parameter estimation: At the moment, the parameters for each geometry are calculated separately. This makes it difficult to compensate for the delayed detection of each track geometry and the therewith associated estimation

of a shifted starting point for the individual track elements (c. f. Sec. V-B.2). Therefore, an approach considering all past track-geometries and estimating the parameters for all known track geometries together is imaginable. Consequently, this leads to a filter exactly relying on Bayesian inference, which is planned for the future.

3) Initial assumptions: During the evaluation within simulations some assumptions were made that need to be checked before applying the filter to practical applications. Especially, the assumptions concerning the process and measurement noises need further attention, as many sensors are affected by a drift or other effects which does not justify the zero-mean assumption. Even the assumption of Gaussian noise is not justified in all cases. Besides the noise assumptions, the multiple linearizations throughout the EKFs can be improved to gain a more correct propagation of the uncertainties. To deal with these problems preprocessing of the sensor data and more sophisticated filter approaches like an unscented Kalman filter (UKF) or a particle filter (PF) can be applied.

4) Track map generation: It is interesting to note that the estimated track-geometry parameters $\tilde{\boldsymbol{g}}$ also can be used directly as a compact geometric track map (c. f. Fig. 6). Actually, track maps like this are required by existing train localization approaches [7]–[9], [11], [12]. The compactness results from the minimal information that needs to be saved for each track element (c. f. Sec. IV-B), which stands in contrast to existing mapping solutions where the track map is often represented by a kind of lookup-table, containing many data points [21], [22]. The compact form can be very beneficial when the track map needs to be transferred wirelessly, which is thinkable in future train control systems.

VI. Conclusions

Throughout this paper, we presented a recursive multistage filtering approach for future train-borne localization systems. In simulations we demonstrated that the filter significantly increases the cross-track positioning accuracy compared to common filter techniques which utilize an EKF. This is achieved by exploiting track-geometry constraints known in advance. By that the presented approach potentially contributes to an improved track selectivity in future train-borne localization systems, which is one of the most crucial factors for the introduction of these systems. Furthermore, compact geometric track-maps can be extracted, which can be utilized in existing train localization approaches.

In the future, we want to concentrate on further developing the filter by utilizing exact Bayesian inference, expanding it by additional track-geometry models as well as considering branching situations. It is also planned to test the approach with actual sensor data.

References

[1] G. Theeg and S. Vlasenko, Eds., *Railway Signalling & Interlocking: International Compendium*, 1st ed. Eurailpress, 2009.

[2] J. Otegui, A. Bahillo, I. Lopetegi, and L. E. Díez, "A survey of train positioning solutions," *IEEE Sensors J.*, vol. 17, no. 20, pp. 6788–6797, Aug. 2017.

[3] *EN50126 - Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS)*, Eur. Commitee for Electrotechnical Standardization Std.

[4] J. Haldor and F. Lademann, *Planung von Bahnanlagen: Grundlagen-Planung-Berechnung*, 2nd ed. Carl Hanser Verlag GmbH & Co. KG, 2017.

[5] K. Lüddecke and C. Rahmig, "Evaluating multiple GNSS data in a multi-hypothesis based map-matching algorithm for train positioning," in *Proc. IEEE Intelligent Vehicles Symp. (IV)*, Jun. 2011, pp. 1037–1042.

[6] M. Lauer and D. Stein, "A train localization algorithm for train protection systems of the future," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 970–979, Apr. 2015.

[7] A. Broquetas, A. Comerón, A. Gelonch, J. M. Fuertes, J. A. Castro, D. Felip, M. A. López, and J. A. Pulido, "Track detection in railway sidings based on MEMS gyroscope sensors," *MDPI AG Sensors J.*, vol. 12, pp. 16 228–16 249, Nov. 2012.

[8] O. Heirich, "Bayesian train localization with particle filter, loosely coupled GNSS, IMU, and a track map," *Hindawi Ltd. J. Sens.*, vol. 2016, Mar. 2016.

[9] J. Liu, B.-g. Cai, and J. Wang, "Track-constrained GNSS/odometer-based train localization using a particle filter," in *Proc. IEEE Intelligent Vehicles Symp. (IV)*, Jun. 2016, pp. 877–882.

[10] A. Neri, V. Palma, F. Rispoli, and A. M. Vegni, "Track constrained PVT estimation based on the double-difference technique for railway applications," in *Proc. IEEE 21st Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2013, pp. 1–5.

[11] O. G. Crespillo, O. Heirich, and A. Lehner, "Bayesian GNSS/IMU tight integration for precise railway navigation on track map," in *Proc. IEEE Position, Location and Navigation Symp. - PLANS*. IEEE/ION, May 2014, pp. 999–1007.

[12] P. Zeller, B. Siebler, A. Lehner, and S. Sand, "Relative train localization for cooperative maneuvers using GNSS pseudoranges and geometric track information," in *Proc. IEEE Int. Conf. on Location and GNSS (ICL-GNSS)*, 2015, pp. 1–7.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., 2006.

[14] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, 2001.

[15] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," in *Proc. IEEE Int. Conf. on Information Fusion*, Cologne, Germany, Jun. 2008.

[16] M. S. Nixon, "Circle extraction via least squares and the Kalman filter," in *Proc. Int. Conf. on Computer Analysis of Images and Patterns (CAIP)*. Budapest, Hungary: Springer, Sep. 1993, pp. 199–207.

[17] B. Belabbas, A. Grosch, O. Heirich, A. Lehner, and T. Strang, "Curvature classification for trains using along-track and cross-track accelerometer and a heading rate gyroscope," in *Proc. of the Eur. Navigation Conf. (ENC)*, Vienna, Austria, Apr. 2013. [Online]. Available: http://elib.dlr.de/82900/

[18] M. Malvezzi, G. Vettori, B. Allotta, L. Pugi, A. Ridolfi, and A. Rindi, "A localization algorithm for railway vehicles based on sensor fusion between tachometers and inertial measurement units," *Proc. of the Institution of Mech. Engineers, Part F: J. of Rail and Rapid Transit*, vol. 228, no. 4, pp. 431–448, 2014.

[19] C. Yang, L. Kaplan, and E. Blasch, "Performance measures of covariance and information matrices in resource management for target state estimation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2594–2613, Jul. 2012.

[20] J. Gong, "Clarifying the standard deviational ellipse," *John Wiley & Sons Geographical Analysis J.*, vol. 34, no. 2, pp. 155–167, 2002.

[21] C. Hasberg, S. Hensel, and C. Stiller, "Simultaneous localization and mapping for path-constrained motion," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 541–552, Jun. 2012.

[22] O. Heirich, P. Robertson, and T. Strang, "RailSLAM - localization of rail vehicles and mapping of geometric railway tracks," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013, pp. 5212–5219.