Hasberg, Carsten ; Hensel, Stefan ; Stiller, Christoph:
Simultaneous Localization and Mapping for Path-Constrained Motion.
IEEE Transactions on Intelligent Transportation Systems 13 (2012), June,
No. 2, pp. 541–552

Simultaneous Localization and Mapping for Path-Constrained Motion

Carsten Hasberg, *Student Member, IEEE,* Stefan Hensel, *Student Member, IEEE* and Christoph Stiller, *Senior Member, IEEE*

***Abstract*—Accurate localization is a fundamental component of driver assistance systems and autonomous vehicles. For path-constrained motion a map offers significant information and assists localization with valuable information about the evolution of the kinematic vehicle states. We propose natural parameterized cubic spline curves to approximate the true motion constraints, in particular the centerline of individual road lanes or rail tracks. Vehicle kinematics are modeled in one dimensional curve coordinates. Since map information is subject to uncertainties a probabilistic treatment is a prerequisite to obtain consistent localization results. The proposed probabilistic curvemap (PCM) and the close map-to-vehicle relation enable a straight forward derivation of measurement update equations without additional map matching steps and offer themselves to classical filter techniques. Incoming sensor measurements are used for a simultaneous vehicle localization and a local PCM update around the current vehicle position. Thus, every revisit of a location reduces uncertainty in the local PCM. Moreover, when no prior information is provided in the PCM, extrapolation is carried out to handle these situations with incomplete maps. The proposed filter is validated through simulations and real world railway experiments.**

***Index Terms*—Cubic Splines, Simultaneous Localization and Mapping (SLAM), Probabilistic Curvemap (PCM).**

I. Introduction

LOCALIZATION is a key capability of driver assistance systems and autonomous vehicles. Incorporating prior information about dynamic behavior and measurement model both have a strong impact on the performance of vehicle localization algorithms [1], [2]. Assumptions on road- or track-constrained motion may augment significant information to the localization process for ground vehicles. Restricting vehicle position on one dimensional curves is natural for numerous applications, e.g. for passenger cars that are constrained on road lanes or track-bounded railway vehicles. Moreover, a correct and up-to-date map of the road network must be permanently available to realize a road map assisted localization and to obtain localization results [3], [4], [5], etc. Finally, gross violations of the path constraint indicate exceptions from normal operation, thus their detection offers important information to initiate appropriate exception handling.

Several concepts have been proposed to assist localization of moving vehicles using roadmap information [6]. A key difference of the concepts is the level of map integration into the localization filter: The map matching methods presented in [7] process the available sensor observations and constrain the resulting estimate onto the map in a post filter processing step. Hence the map is considered as ground truth. In contrast, a second group of algorithms obtains pseudo-measurements of the map, which are considered as data from an additional static sensor [8]. The approaches subsumed in a third group directly integrate the available roadmap within the filtering step. In particular the approaches presented in [4] and [9] make use of a 1D representation of the vehicle kinematics in arc-length coordinates. This integration strategy is adapted in the present proposal. In combination with arc-length parameterized curvemaps it enables analytical transformations between measurement- and state-space and therefore guarantees consistent observation equations.

Realistic motion constraints depend on construction rules and vehicle dynamics: Roads or tracks are commonly composed of a sequence of geometric primitives to enable comfortable driving without abrupt variations in lateral acceleration. Often polygonal models [3], [10], [11] or clothoids [12] are applied to approximate the centerline of each road lane or rail track. In this contribution smooth third order spline curves are proposed. We show that this modeling strategy overcomes systematic interpolation errors and has similar characteristics as B-Splines used in [13] to model road course. It yields highly accurate approximation results, both for strongly curved and straight sections of variable length. Last but not least the model allows a consistent integration into the estimation process.

Any digital map is corrupted by topological and geometrical uncertainty due to generic surveying and mapping processes that contain measurement and approximation steps [5]. Topological inconsistencies arise from the fact that map features, e.g. junctions, have been missed or simplified during map creation. Geometrical uncertainty originates from the geometric displacement of map features that are caused by error-prone mapping steps. This contribution focuses on a probabilistic treatment of geometric map uncertainty. For that reason only single segments within the network of motion constraints are considered and a correct vehicle-to-segment (e.g. lane) association is assumed to be known. A detection of lane changing maneuvers, as proposed in [14], could extend the proposed strategy to multi lane scenarios with unknown association.

A probabilistic model is required for a systematic treatment of the geometric map errors. In [4] the roadmap supports the proposed vehicle localization with uncertainty information for each single road segment. This contribution extends this concept: The presented probabilistic curvemap (PCM) offers continuous uncertainty information along each map

C. Hasberg, S. Hensel and C. Stiller are with the Institute of Measurement and Control Systems, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany, e-mail: {carsten.hasberg, stefan.hensel, christoph.stiller}@kit.edu.

Manuscript received January 10, 2010; revised January 11, 2010.

element. The resulting problem to locate the vehicle and update the PCM with incoming sensor measurements exhibits many similarities to a fundamental problem in robotics, the simultaneous localization and mapping problem (SLAM). In SLAM the robot acquires a map of its environment while simultaneously localizing itself relative to the map [15]. A survey that summarizes work on that problem is presented in [16]. A consistent SLAM solution employs the extended Kalman filter (EKF) [17]. It has successfully been adapted to vehicle localization in [18] and is applied within the proposed framework.

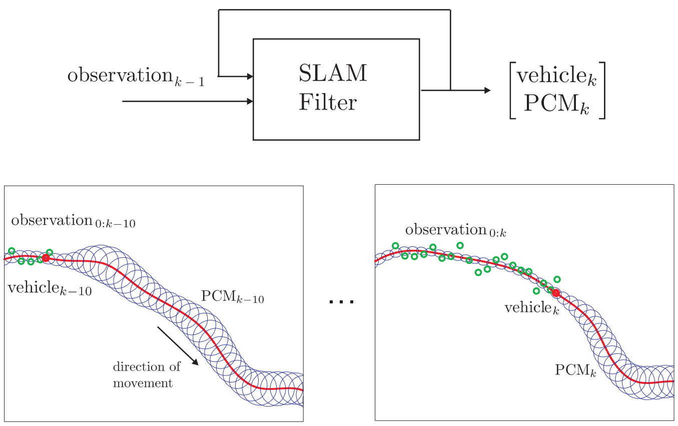


Fig. 1. Concept of recursive localization assisted by probabilistic curvemap (PCM): Observations, e.g. from GPS or odometer are fed into the localization filter together with current vehicle kinematic and PCM states. Within the filter the map- and the vehicle-states are adapted simultaneously. As illustrated in the lower series the map segments and their uncertainties are adapted in the vicinity of the traveling vehicle.

In order to follow the paradigm of EKF-SLAM, an appropriate model for curve-constrained motion has to be developed. The main contribution of this proposal is the deduction of a probabilistic model that combines the vehicle motion and the curve constraint in one state. Based on that model mapping is conducted, while the vehicle is traveling and localization is performed. The result is an estimate that simultaneously represents the map and the vehicle state, as visualized in Fig. 1. A Kalman filter parameter update of spline curves and their covariances based on GPS observations has been presented in prior work [19]. We extend this approach by incorporating additional observations and a kinematic vehicle model. Hence, the map does not only contribute to improve localization but map and localization are processed to mutually improve one another. During the map update procedure the geometric map parameters are adapted locally around the current vehicle position. The map accuracy increases and the overall precision of the map assisted localization is continuously improved, each time a measurement is assigned to a certain map element.

The remainder of this paper is structured as follows: Sec. II introduces the fundamental concept of PCM interpolation with cubic spline curves. Sec. III focuses on vehicle kinematics and presents the selected motion model in curve coordinates. In Sec. IV the recursive estimation of vehicle kinematics and PCM is described step by step. The compact curve representation enable a wide range of opportunities to integrate sensor measurements. In particular, the integration of speed, position and heading observations is presented. Additionally, a method is proposed to handle map growing situations by extrapolating the available map and adding new states to the map parameter vector. Several simulations of a scenario with a vehicle that is already assigned to a certain map element are presented in Sec. V to verify the proposed approach. Finally, the methods are validated within a real world train positioning system. The results are presented in Sec. VI. Section VII summarizes the proposed method and draws conclusions. Throughout this contribution the derivative with respect to time is denoted by $\dot{x}$ while the derivative with respect to curve parameter is denoted by x'.

II. Probabilistic Curvemap (PCM)

Polynomial concepts played a crucial role for interpolation but are nowadays of more theoretical value: Faster and more accurate methods have been developed [20]. Those methods are piecewise polynomial, but still rely on the classical polynomial concepts. A prominent class for piecewise polynomial interpolation are cubic spline curves, that are used throughout this proposal to approximate the true progression of road lanes or track segments.

A general distinction is drawn between local and global cubic splines. In contrast to local cubic splines or hermite splines, the slopes at the supporting points of global cubic splines are not specified, but are instead computed from imposed continuity conditions [21]. The latter is chosen to approximate motion constraints imposed by road lanes or rail tracks precisely. Two properties of global cubic splines support that choice: Commonly roads or tracks are structured to enable a comfortable driving with low lateral jerk. Therefore, the variation of the road curvature is smooth. As a consequence accurate interpolation can be obtained by a two times continuously differentiable model. Global cubic splines implicitly fulfill this requirement. When the interpolation of a given set of supporting points along the motion constraints is carried out over-fitting effects have to be avoided. Amongst all twice continuously differentiable functions that interpolate a set of supporting points and satisfy the same end conditions, the global cubic spline $\mathbf{s}(l)$ yields the smallest norm of the approximative strain energy

$$\mathbf{E} = \int [\mathbf{s}''(l)]^2 dl, \tag{1}$$

which corresponds to a minimization of curvature energy [20]. The result is a curve progression with minimal oscillations in between the supporting points.

A. Interpolating Paths with Global Cubic Splines

The interpolation is carried out stepwise, based on a given set of supporting points $\mathbf{p}_i = (p_{x,i}, p_{y,i})^\mathrm{T}$ for $i = 0, \ldots, n$. The first step is to determine a curve parameterization for each supporting point $\mathbf{p}_i$. In chord-length parameterization a parameter u_i is calculated in a recursive manner, following the equation

$$u_{i+1} \quad = \quad u_i + \|\mathbf{p}_{i+1} - \mathbf{p}_i\|, \tag{2}$$

with $u_0 = 0$ and $i = 0, \dots, n-1$. Each geometric segment is interpolated with a planar curve

$$\mathbf{s}_i(u) = \begin{bmatrix} a_{x,i} + b_{x,i}\Delta u_i + c_{x,i}\Delta u_i^2 + d_{x,i}\Delta u_i^3 \\ a_{y,i} + b_{y,i}\Delta u_i + c_{y,i}\Delta u_i^2 + d_{y,i}\Delta u_i^3 \end{bmatrix}, \tag{3}$$

where u denotes chord-length and $\Delta u_i = u - u_i$ with $u_i \leq u < u_{i+1}$. The global curve $\mathbf{s}(u)$ is composed piecewise by the curve segments $\mathbf{s}_i(u)$. It interpolates the supporting points and satisfies a set of smoothness conditions. The geometric parameters are calculated for the x- and y-component of $\mathbf{s}(u)$ separately. The next paragraph presents the calculations for the x-component. The values for the y-component are extracted in an analogous manner.

The unknown parameters are evaluated based on the supporting points $p_{x,i}$ and the unknown second derivatives at the supporting points, the so called moments $m_{x,i} = m_x(u_i)$:

$$\begin{aligned} a_{x,i} &= p_{x,i} && (4)\\ b_{x,i} &= \frac{p_{x,i+1} - p_{x,i}}{h_i} - \frac{h_i(2m_{x,i} + m_{x,i+1})}{6} && (5)\\ c_{x,i} &= \frac{m_{x,i}}{2} && (6)\\ d_{x,i} &= \frac{m_{x,i+1} - m_{x,i}}{6h_i} && (7) \end{aligned}$$

with $h_i = u_{i+1} - u_i$ and $i = 0, \dots, n-1$ [22]. The yet unknown moments are calculated by imposing the condition of first-derivative continuity at the inner supporting points. Postulating linearity of the second derivative and integrating twice combined with the continuity condition of the first derivatives yields

$$\begin{aligned} &h_i m_{x,i} + 2(h_i + h_{i+1})m_{x,i+1} + h_{i+1}m_{x,i+2} = \\ &\qquad \frac{6}{h_{i+1}}(p_{x,i+2} - p_{x,i+1}) - \frac{6}{h_i}(p_{x,i+1} - p_{x,i}) \end{aligned} \tag{8}$$

for each interval $[u_{i-1}, u_i]$ [22] [23]. Since the moments at the first and the last supporting point disappear, due to the additionally chosen natural end conditions [21], the resulting set of $n-1$ equations is uniquely solvable and yields the inner moments $m_{x,i}$. Once the moments $m_{x,i}$ are known for all supporting points $p_{x,i}$, the parameters of the piecewise polynomials are calculated according to (4) to (7).

The resulting curve $\mathbf{s}(u)$ is given in chord-length rather than arc-length parameterization and the parameterization error $e = l - u$ increases along the arc-length

$$l = f(u) = \int_0^u \|\mathbf{s}'(\tau)\| d\tau \tag{9}$$

of the spline curve.

To enable a stable integration of one-dimensional kinematic states into the spline curve a re-parameterization to arc-length parameterization is conducted. Principally, the arc-length parameterization of $\mathbf{s}(u)$ can be calculated in two steps. The first step is to evaluate the inverse of (9) to obtain chord-length as function of arc-length: $u = f^{-1}(l)$. Substitution of the inverse into $\mathbf{s}(u)$ yields an arc-length parameterized curve $\mathbf{s}(l) = \mathbf{s}(f^{-1}(l))$. In general, this equation does not offer an analytical solution but the arc-length parameterization has to be approximated using numerical methods.

Throughout this proposal the approximating technique presented in [24] is adapted: Initially the arc-length values for the supporting points $\mathbf{p}_i$ are calculated, according to

$$l_{i+1} \quad = \quad l_i + \int_{u_i}^{u_{i+1}} \|\mathbf{s}'(\tau)\| d\tau, \tag{10}$$

in a recursive manner. Through a final processing of supporting points $\mathbf{p}_i$ and corresponding arc-length values l_i an approximate arc-length parameterized curve $\mathbf{s}(l)$ is calculated.[1]

B. Linear Gauss-Markov Model of the Spline Map

Based on the recursive formulation presented in the last section a linear Gauss-Markov model is formulated. A similar model has been proposed in [26] for machine tool calibration. Again, we restrict ourselves to the first component s_x. Let the polynomial coefficients be arranged in column vectors

$$\begin{aligned} \mathbf{a}_x &= [a_{x,0} \ \dots \ a_{x,n-1}]^{\mathrm{T}} && (11)\\ &\vdots \qquad \vdots \\ \mathbf{d}_x &= [d_{x,0} \ \dots \ d_{x,n-1}]^{\mathrm{T}}. && (12) \end{aligned}$$

Additionally the x-components of the supporting point vectors $\mathbf{p}_i$ are grouped according to:

$$\mathbf{q}_x = [p_{x,0} \ \dots \ p_{x,n}]^{\mathrm{T}}. \tag{13}$$

As shown in the Appendix A the recursive equations (4) to (7) and (8) can be rewritten in linear matrix formulation as

$$\begin{matrix} \mathbf{a}_x = \mathbf{A}\mathbf{q}_x & \mathbf{c}_x = \mathbf{C}\mathbf{q}_x \\ \mathbf{b}_x = \mathbf{B}\mathbf{q}_x & \mathbf{d}_x = \mathbf{D}\mathbf{q}_x \end{matrix}, \tag{14}$$

yielding the coefficients of the piecewise defined polynomials.

For a known curve parameter l the corresponding curve segment i is defined through $l_i \leq l < l_{i+1}$. The masking vector $\mathbf{k}_i$ is designed to select the corresponding component function from vector $\mathbf{s}_x(l)$ according to

$$s_x(l) = \mathbf{k}_i^{\mathrm{T}} \cdot \mathbf{s}_x(l) = \begin{bmatrix} \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix}^{\mathrm{T}} \cdot \begin{bmatrix} \vdots \\ s_{x,i-1}(l) \\ s_{x,i}(l) \\ s_{x,i+1}(l) \\ \vdots \end{bmatrix}. \tag{15}$$

Substitution of the spline component vector function $\mathbf{s}_x(l)$ with the analytical expression in (14) and transformation of the equation

$$\begin{aligned} s_x(l) &= \mathbf{k}_i^{\mathrm{T}} \mathbf{s}_x(l) \\ &= \mathbf{k}_i^{\mathrm{T}}[\mathbf{a}_x + \mathbf{b}_x\Delta l_i + \mathbf{c}_x\Delta l_i^2 + \mathbf{d}_x\Delta l_i^3] \\ &= \mathbf{k}_i^{\mathrm{T}}[\mathbf{A}\mathbf{q}_x + \mathbf{B}\mathbf{q}_x\Delta l_i + \mathbf{C}\mathbf{q}_x\Delta l_i^2 + \mathbf{D}\mathbf{q}_x\Delta l_i^3] \\ &= \mathbf{k}_i^{\mathrm{T}}[\mathbf{A} + \mathbf{B}\Delta l_i + \mathbf{C}\Delta l_i^2 + \mathbf{D}\Delta l_i^3]\mathbf{q}_x \\ &:= \mathbf{g}^{\mathrm{T}}(\mathbf{l}, l)\mathbf{q}_x \end{aligned} \tag{16}$$

[1] The shape of $\mathbf{s}(l)$ slightly differs from $\mathbf{s}(u)$ because of the nonlinear characteristics of the transformation that has been applied to $u_0, \dots, u_n$ to obtain $l_0, \dots, l_n$ [25].

yields a linear relationship[2] between the x-component supporting point vector $\mathbf{q}_x$ and the function value s_x for the parameter value l with $\mathbf{l} = [l_0 \dots l_n]^{\mathrm{T}}$.

For the second component one yields an analogue linear relationship between the component function $s_y(l)$ and the y-component supporting point vector

$$s_y(l) \quad = \quad \mathbf{g}^{\mathrm{T}}(\mathbf{l}, l)\mathbf{q}_y. \tag{17}$$

In conclusion, the whole process of spline curve parameter calculation of $\mathbf{s}(l) = [s_x(l) \;\; s_y(l)]^{\mathrm{T}}$ for a given set of supporting points is expressed by a linear vector equation:

$$\mathbf{s}(l) \quad = \quad \begin{bmatrix} \mathbf{g}^{\mathrm{T}}(\mathbf{l}, l) & \mathbf{0} \\ \mathbf{0} & \mathbf{g}^{\mathrm{T}}(\mathbf{l}, l) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{q}_x \\ \mathbf{q}_y \end{bmatrix} = \mathbf{G}(\mathbf{l}, l) \cdot \mathbf{x}_m, \tag{18}$$

where all spline supporting points are subsumed in $\mathbf{q}_x$ and $\mathbf{q}_y$ [19].

The parameter vector $\mathbf{x}_m$ is assumed to be a Gaussian distributed random variable[3], with

$$\mathbf{x}_m \;\sim\; \mathcal{N}(\hat{\mathbf{x}}_m, \mathbf{P}_{mm}). \tag{19}$$

Due to the linear transformation in (18), the vector $\mathbf{s}(l)$ is also Gaussian distributed, with

$$\begin{aligned} \mathbf{s}(l) \;&\sim\; \mathcal{N}(\hat{\mathbf{s}}(l), \mathbf{P}_{\mathbf{ss}}(l)) \\ &\sim\; \mathcal{N}(\mathbf{G}(\mathbf{l}, l)\, \hat{\mathbf{x}}_m, \mathbf{G}(\mathbf{l}, l)\, \mathbf{P}_{mm}\, \mathbf{G}^{\mathrm{T}}(\mathbf{l}, l)). \end{aligned} \tag{20}$$

The dependency of the parameter vector $\mathbf{l}$ in $\mathbf{G}(\mathbf{l}, l)$ is skipped in the sequel to keep the notation uncluttered.

C. First Derivative of the Spline Map

The derivation of the tangent vector $\mathbf{t}(l) = \mathbf{s}'(l)$ as a function of the arc-length is straight forward and will be applied to design a state-to-measurement relationship for course observations.

A linear relationship between the geometric map parameter vector $\mathbf{x}_m$ and the tangent vector $\mathbf{t}(l) = [t_x(l) \;\; t_y(l)]^{\mathrm{T}}$ is derived from (18)

$$\mathbf{t}(l) \quad = \quad \begin{bmatrix} (\mathbf{g}^{\mathrm{T}}(l))' & \mathbf{0} \\ \mathbf{0} & (\mathbf{g}^{\mathrm{T}}(l))' \end{bmatrix} \mathbf{x}_m = \mathbf{G}'(l)\mathbf{x}_m \tag{21}$$

with

$$(\mathbf{g}^{\mathrm{T}}(l))' \quad = \quad \mathbf{k}_i^{\mathrm{T}}[\mathbf{B} + 2\mathbf{C}\Delta l_i + 3\mathbf{D}\Delta l_i^2], \tag{22}$$

whereas the tangent vector is always normalized $\|\mathbf{t}(l)\| = 1$ for arc-length parameterized curves [20]. Following the reasoning above, $\mathbf{t}(l)$ is again Gaussian distributed

$$\begin{aligned} \mathbf{t}(l) \;&\sim\; \mathcal{N}(\hat{\mathbf{t}}(l), \mathbf{P}_{\mathbf{tt}}(l)) \\ &\sim\; \mathcal{N}(\mathbf{G}'(l)\, \hat{\mathbf{x}}_m, \mathbf{G}'(l)\, \mathbf{P}_{mm}\, (\mathbf{G}'(l))^{\mathrm{T}}) \end{aligned} \tag{23}$$

and can be calculated for given arc-length position.

[2]Strictly, the parameter separation in (16) is not perfect because of the dependency of the supporting point parameter values l_i on the supporting point positions according to (10). Due to its insignificant influence this interrelationship is neglected in this approach.

[3]Beside the central limit theorem this assumption is mainly motivated through the resulting analytical properties. In general other distributions can also be used.

D. Evaluation of Approximation Errors

The calculated curve $\mathbf{s}(l)$ is an approximation of the true trace in two senses: The shape of $\mathbf{s}(l)$ is an approximation of the shape of the true trace and the curve $\mathbf{s}(l)$ is approximately arc-length parameterized. Both error sources are interdependent and are highly related to the arc-length distance between adjacent supporting points $\Delta l = l_{i+1} - l_i$. A reasonable choice for Δl may be derived form a study of the overall approximation accuracy. Therefore a set of S-shaped test traces with varying curvature have been approximated with global cubic splines.

For a given trace the discrete Fréchet distance d_{fr} to the approximated spline curve is calculated initially.[4] In order to evaluate the performance of the arc-length approximation the maximum arc-length error

$$e_{\max} = \max_{l}\{l - \int_0^l \|\mathbf{s}'(\tau)\| d\tau\} \tag{24}$$

is calculated. If the minimum curve radius r is known Fig. 2 allows to pre-estimate the approximation errors for a given distance Δl. Thus, an appropriate choice of Δl can directly be conducted for a given curve radius and admissible uncertainty.

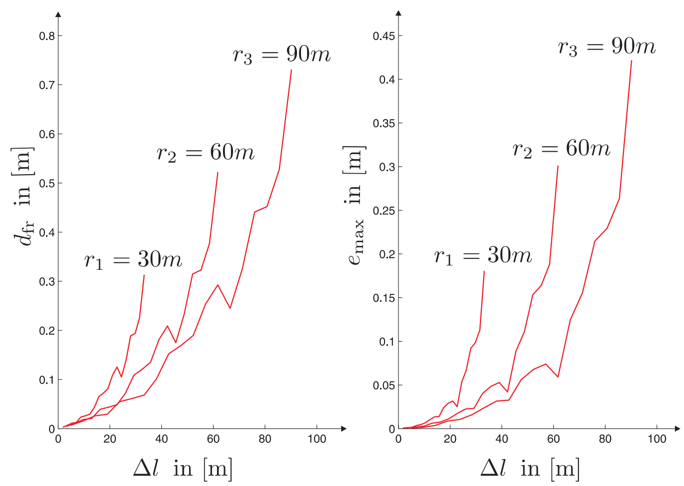


Fig. 2. Fréchet distance d_{fr} and arc-length error $e_{\max}$ between trace and cubic spline curve as function of distance between adjacent supporting points $\Delta l = l_{i+1} - l_i$ for three different curve radiuses r_1, r_2 and r_3.

III. Curve Constrained Motion Model

The design of the kinematic vehicle model conducted in an appropriate coordinate frame serves to predict the vehicle states from one time step to the next. This choice has significant impact on the performance of vehicle localization [2] and has to be adapted vehicle specific.

A. Motion Model in Local Curve Coordinate Frame

Object tracking is conventionally formulated in global Cartesian coordinates. If a PCM is given, an elegant formulation is provided in local curve coordinates. While classical

[4]The Fréchet distance measures the similarity between two curves. It takes into account the location and ordering of the points along the curves [27].

models in Cartesian coordinates allows the vehicle to reach every position in the plane measurement space, a description in local curve coordinates results in a vehicle that is constrained to the given PCM. Assuming the PCM to be known, the local curve model summarizes the past history of the system sufficiently and predicts future positions more precisely. Local curve coordinates constrain the mean of the dynamic states to remain on the path, while time- and measurement updates are calculated. Therefore a post-filter map matching step to project the estimate onto the map can be avoided [28].

In this proposal the kinematic states of a moving vehicle are modeled in local curve coordinates, combining arc-length position and arc-length position derivatives

$$\mathbf{x}_{d,k} = [l_k\ \dot{l}_k\ \ldots\ l_k^{(n)}]^{\mathrm{T}}. \tag{25}$$

B. Discrete Wiener Process Acceleration Model

Numerous models to predict position and its uncertainty of a moving vehicle are reported in literature. Throughout this approach the Discrete Wiener Process Acceleration (DWPA) model [1] is chosen and arc-length position l_k, arc-length velocity $\dot{l}_k$ and arc-length acceleration $\ddot{l}_k$ are grouped in the vehicle state vector, as $\mathbf{x}_{d,k} = [l_k\ \ \dot{l}_k\ \ \ddot{l}_k]^{\mathrm{T}}$, i.e. motion is inherently constrained on the PCM. The transition equation of the third order state equation is given by

$$\mathbf{x}_{d,k+1} = \mathbf{F}_d \mathbf{x}_{d,k} + \mathbf{\Gamma}_d \mu_k \tag{26}$$

with

$$\mathbf{F}_d = \begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \mathbf{\Gamma}_d = \begin{bmatrix} \frac{1}{2}T^2 \\ T \\ 1 \end{bmatrix}. \tag{27}$$

In this model the white noise μ_k is the acceleration increment during the k-th sampling period. It is assumed to be a zero-mean white sequence $\mu_k \sim \mathcal{N}(0, \sigma_\mu^2)$. The only design parameter σ_μ should be selected of the order of the magnitude of the maximum acceleration increment Δa over a sampling period T. A practical range is $0.5\Delta a \leq \sigma_\mu \leq \Delta a$ [1].

IV. Simultaneous Localization and Mapping

Throughout the presented approach the kinematic states of a moving vehicle $\mathbf{x}_{d,k}$ are modeled in local curve coordinates and the geometrical spline map information is subsumed by $\mathbf{x}_{m,k}$ as outlined in Sec. II and Sec. III. Altogether the resulting state vector contains both components and is assumed to be a Gaussian distributed random variable, with mean and covariance

$$\hat{\mathbf{x}}_k = \begin{bmatrix} \hat{\mathbf{x}}_{d,k} \\ \hat{\mathbf{x}}_{m,k} \end{bmatrix} \quad \text{and} \quad \mathbf{P}_k = \begin{bmatrix} \mathbf{P}_{dd,k} & \mathbf{P}_{dm,k} \\ \mathbf{P}_{dm,k}^{\mathrm{T}} & \mathbf{P}_{mm,k} \end{bmatrix}. \tag{28}$$

After the state $\mathbf{x}_0$ is initialized, an EKF is applied to update the entire state $\mathbf{x}_k$ in a time update and a measurement update step. When the vehicle travels beyond the last supporting point map extrapolation is performed, before a re-parameterization and a re-sampling step completes one processing loop. It is worth noting that the proposed algorithm handles the initial mapping of unknown areas and a map update in areas of existing but uncertain maps in a unified framework.

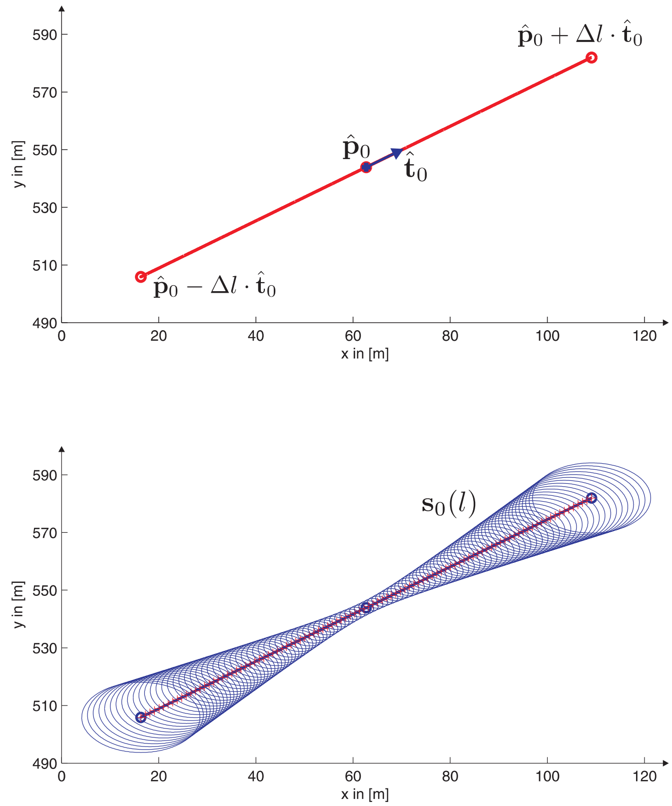


Fig. 3. Tangential PCM initialization based on the first available sensor reading $\mathbf{z}_0 = [\mathbf{p}_0^{\mathrm{T}}\ \mathbf{t}_0^{\mathrm{T}}\ v_0]^{\mathrm{T}}$ and a fixed distance Δl. The PCM is constructed as $\mathbf{s}_0(l) = \mathbf{G}(l)\mathbf{x}_{m,0}$.

A. State Vector Initialization

The initial state vector $\mathbf{x}_0$ is composed of a kinematic and a static component. It is based upon the first available observation $\mathbf{z}_0$. Each measurement $\mathbf{z}_k$ consists of three main components. The position $\mathbf{p}_k$ and the direction $\mathbf{t}_k$ are both given in global Cartesian coordinates. In contrast, speed v_k is measured in direction of movement in local curve coordinates. Each measurement $\mathbf{z}_k = [\mathbf{p}_k^{\mathrm{T}}\ \mathbf{t}_k^{\mathrm{T}}\ v_k]^{\mathrm{T}}$ is assumed a Gaussian random variable $\mathbf{z}_k \sim \mathcal{N}(\hat{\mathbf{z}}_k, \mathbf{R}_k)$ with known covariance. Measurement and system noise are assumed to be uncorrelated.

Two main cases can be distinguished for state vector initialization: If an explicit assignment to a certain map element characterized by $\mathbf{x}_{m,0}$ is given, orthogonal projection of the first available measurement $\mathbf{z}_0$ onto that element yields $\mathbf{x}_{d,0}$. If no map element is provided it has to be initialized, too. Based on the position measurement $\mathbf{p}_0$ three supporting points are calculated in direction of the first tangent observation $\mathbf{t}_0$ and form the initial map state

$$\mathbf{x}_{m,0} = \begin{bmatrix} \mathbf{I} & -\Delta l\,\mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \Delta l\,\mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{z}_0 + \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{w}_{tan}, \tag{29}$$

with Δl denoting the arc-length between adjacent supporting points, as shown in Fig 3. The white noise term $\mathbf{w}_{tan} \sim \mathcal{N}(\mathbf{0}, \sigma_{tan}^2\mathbf{I})$ is added to explain model errors that occur in curved sections, where the assumption of tangential extrapolation does not hold.

B. EKF Update

Throughout the following subsection the state update process during one time interval T is described step by step. The superscript $(.)^+$ is used to identify variables that have been updated within the current subsection while it is omitted at the end of the subsection.

The core of the proposed recursive localization and mapping problem is solved with the EKF, an approximation of the Bayes filter [29]. Assuming a third order kinematic model in local curve coordinates and a static map, the system equation can be written as

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{\Gamma}\mu_{k-1} = \begin{bmatrix} \mathbf{F}_d & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} \mathbf{\Gamma}_d \\ \mathbf{0} \end{bmatrix} \mu_{k-1}. \quad (30)$$

This defines the prediction step of mean and covariance of the state vector computed in a KF [30].

A transformation of local arc-length coordinates to global measurement coordinates is performed to update the current state estimate $\mathbf{x}_k$ with incoming measurements $\mathbf{z}_k = [\mathbf{p}_k^{\mathrm{T}}, \mathbf{t}_k^{\mathrm{T}}, v_k]^{\mathrm{T}}$. The mathematical relation between both coordinate frames is given by the natural parameterized spline curve as presented in Sec. II. The nonlinear observation equation $\hat{\mathbf{z}}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{w}_k$ can be written as

$$\hat{\mathbf{z}}_k = \begin{bmatrix} \mathbf{s}_k(l_k) \\ \mathbf{t}_k(l_k) \\ \dot{l}_k \end{bmatrix} + \mathbf{w}_k = \begin{bmatrix} \mathbf{G}(l_k)\mathbf{x}_{m,k} \\ \mathbf{G}'(l_k)\mathbf{x}_{m,k} \\ \dot{l}_k \end{bmatrix} + \mathbf{w}_k, \quad (31)$$

which requires the use of an EKF. The error sources for the measurement uncertainty are assumed to be uncorrelated and white, zero-mean distributed Gaussian random sequences $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$.

Given the linear relation between the measurement and the map parameters, linearization is only necessary for the dynamic state components. Based on its Jacobian the EKF measurement update equations adjust the predicted estimate with new incoming measurement information. While the measurement update is calculated the arc-length position estimate $\hat{l}_k^+$ are fixed on the PCM. In parallel the geometric parameters are adapted to position and course measurement. The update yields a smooth adaption of the curve progression, according to the minimum-curvature-property [20] of cubic spline curves. The process is visualized in Fig. 4.

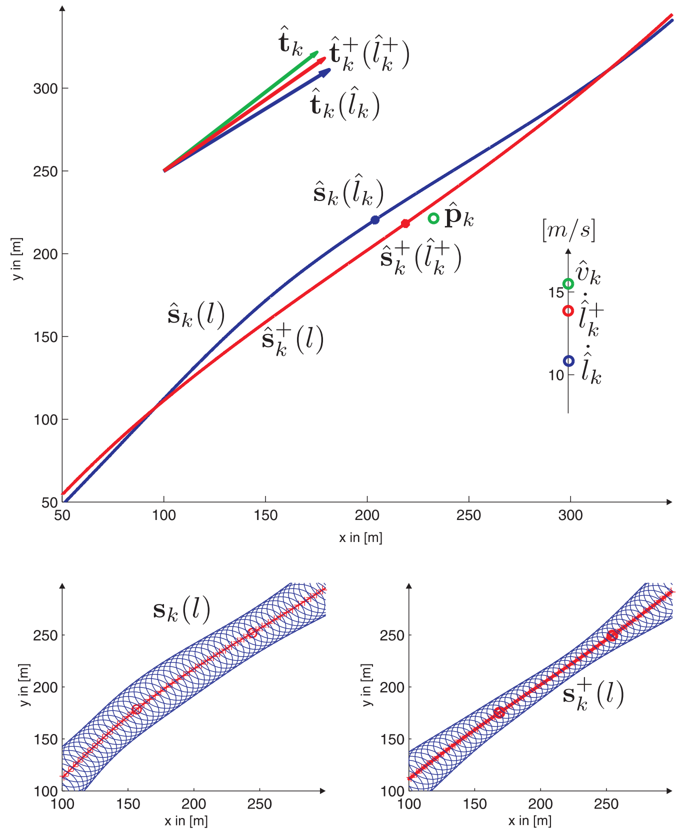


Fig. 4. Measurement innovation of state vector $\mathbf{x}_k$ with measurement $\mathbf{z}_k$: The mean position of the vehicle $\hat{l}_k$ is updated while the curve progression $\hat{\mathbf{s}}_k(l)$ is adapted in parallel. The result is a smooth convergency to the observation, marked with $(.)^+$. For visualization purposes the mean tangent observation $\hat{\mathbf{t}}_k$ and the corresponding predicted course measurements $\hat{\mathbf{s}}_k'(\hat{l}_k) = \hat{\mathbf{t}}_k(\hat{l}_k)$ before and after the update step are linear shifted to a random origin at $\mathbf{p} = [100 \quad 250]^{\mathrm{T}}$. The velocity adjustment is plotted separately, too. The lower plots illustrate the local decrease of geometric uncertainty of the PCM.

C. PCM Extrapolation

Analogue to time-prediction of the dynamic components of the state vector a geometric prediction is performed when the vehicle leaves the defined state space, limited by the total arc-length of the available PCM. Tangential geometric extrapolation is employed, if the current arc-length position estimate reaches the end of the spline map, $\hat{l}_k > l_n$. At first, a new supporting point is augmented to the state vector. The distance Δl is set to a fixed value, with $\Delta l = l_{n+1} - l_n$. The tangentially extrapolated point is then given as

$$\mathbf{p}_{n+1} \;=\; \mathbf{s}(l_n) + \Delta l\, \mathbf{t}(l_n) + \mathbf{w}_{tan}, \quad (32)$$

where the white noise $\mathbf{w}_{tan} \sim \mathcal{N}(\mathbf{0}, \sigma_{tan}^2 \mathbf{I})$ is added with large variance to explain the high uncertainty model errors that occur in unknown curved road sections. Using (18) and (21) yields

$$\begin{aligned} \mathbf{p}_{n+1} &= \mathbf{G}(l_n)\,\mathbf{x}_{m,k} + \Delta l\, \mathbf{G}'(l_n)\mathbf{x}_{m,k} + \mathbf{w}_{tan} \\ &= [\mathbf{G}(l_n) + \Delta l\, \mathbf{G}'(l_n)]\, \mathbf{x}_{m,k} + \mathbf{w}_{tan} \\ &:= \mathbf{U}(l_n)\, \mathbf{x}_{m,k} + \mathbf{w}_{tan}. \end{aligned} \quad (33)$$

The state vector $\mathbf{x}_k$ is augmented by the new supporting point $\mathbf{p}_{n+1}$, according to $\mathbf{x}_k^+ = [\mathbf{x}_k^{\mathrm{T}} \;\; \mathbf{p}_{n+1}^{\mathrm{T}}]^{\mathrm{T}}$. Thus extrapolation can finally be written as

$$\mathbf{x}_k^+ = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{U}(l_n) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_{d,k} \\ \mathbf{x}_{m,k} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{w}_{tan} \end{bmatrix}. \quad (34)$$

Mean and covariance of the extrapolated state vector can now be calculated employing calculations on Gaussian random variables [31].

D. PCM Re-Parameterization

During the measurement update of the system state vector the supporting point positions, subsumed in $\mathbf{x}_{m,k}$ are adjusted, while the arc-lengths l_i are assumed to be constants. Although actual arc-lengths will change only moderately during map adjustment, these need to be updated to avoid drift accumulations.

Initially the exact arc-length is calculated for each vertex based on the current mean run of the spline curve $\mathbf{s}_k(l)$

$$l_{i+1}^{+} \quad = \quad l_i^{+} + \int_{l_i}^{l_{i+1}} \|\hat{\mathbf{s}}_k'(\tau)\| d\tau. \tag{35}$$

Calculation of the curve $\mathbf{s}_k^{+}(l)$ based on the updated set of supporting points $\mathbf{x}_{m,k}$ and corresponding curve parameters $l_0^{+}, \ldots, l_n^{+}$ yields an arc-length parameterized curve.[5]

E. PCM Re-Sampling

In order to obtain a constant curve approximation quality the arc-length distances between adjacent supporting points are reset to the initially chosen value Δl. This is achieved through sampling the curve $\mathbf{s}_k(l)$ at the arc-length values $l_{i+1}^{+} = l_i^{+} + \Delta l$ with $i = 0, \ldots, n$. While doing so the first supporting point parameter value has to remain unchanged $l_0 = l_0^{+}$. The new state vector is given by

$$\mathbf{x}_k^{+} = \begin{bmatrix} \mathbf{x}_{d,k} \\ \mathbf{s}_k(l_0^{+}) \\ \vdots \\ \mathbf{s}_k(l_n^{+}) \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}(l_0^{+}) \\ \vdots & \vdots \\ \mathbf{0} & \mathbf{G}(l_n^{+}) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_{d,k} \\ \mathbf{x}_{m,k} \end{bmatrix}. \tag{36}$$

Again the matrix formulation of the re-sampling step in (36) is a linear transformation of a Gaussian random variable and mean and covariance of the new state vector can be calculated directly. Fig. 5 visualizes the re-sampling process. For a reasonable re-sampling $\mathbf{s}_k^{+}(l) \approx \mathbf{s}_k(l)$ and hence $\mathbf{l}^{+} = \mathbf{l}$ hold.[6]

V. SIMULATIONS

The presented SLAM algorithm was assessed in numerical simulations to verify filter performance under specified conditions. In particular it is examined to what extent different types of corrupted initial maps, e.g. noisy or biased, affect the results.

A. Simulation Setup

At first the true run of the motion path is generated within realistic curvature values. Third order kinematics with a zero mean process noise acceleration sequence of varying variances σ_d^2 are chosen to approximate the maneuvering behavior of different vehicles. Based on this ground truth typical sensor error characteristics are taken into account to generate the measurement input for the presented filter algorithm.

According to typical railway or country road scenarios, maps of different quality levels are generated based on the true

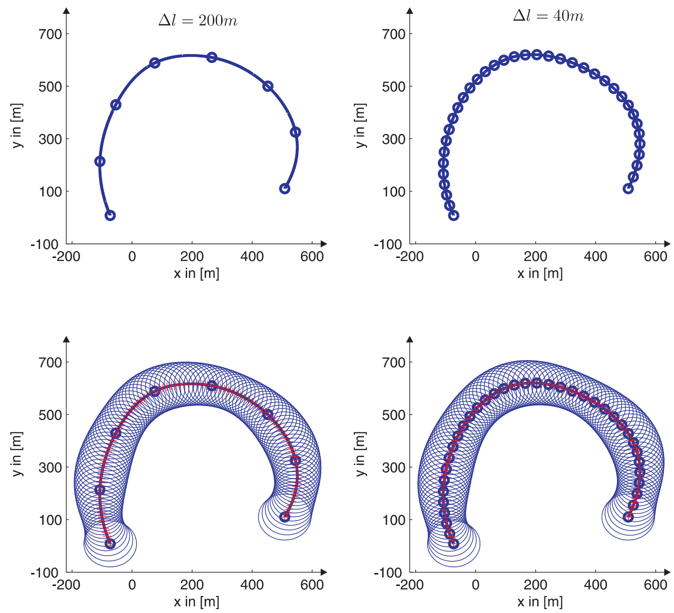


Fig. 5. In the left column the mean progression and the probabilistic curve are given for a supporting point distance of $\Delta l = 200m$. The re-sampling result for $\Delta l = 40m$ is depicted in the right column.

progression, while the minimum radius is set to $r_{\min} = 40$m. Afterwards, each supporting point position $\mathbf{p}_i$ is corrupted with normal distributed white noise and an additional fixed offset $\Delta\mathbf{p}$ according to

$$\hat{\mathbf{p}}_i = \mathbf{p}_i + \Delta\mathbf{p} + \mathcal{N}(\mathbf{0}, \sigma_p^2 \cdot \mathbf{I}) \tag{37}$$

to compute the elements of the initial map state vector $\mathbf{x}_{m,0}$. It is worth noting that our method also allows to set $\mathbf{x}_{m,0}$ empty, when no initial map knowledge is available.

Following the analysis of Section II-D the arc-length distance between adjacent supporting points is set to $\Delta l = 20$m. For the considered traces, this choice guarantees a maximal approximation error of 0.3m as depicted in Fig. 2.

Now measurements are generated based on the the simulated vehicle motion along the true curve progression. Therefore the true values $\mathbf{z}_k = [\mathbf{p}_k^{\mathrm{T}} \;\; \mathbf{t}_k^{\mathrm{T}} \;\; v_k]^{\mathrm{T}}$ are corrupted with additive zero mean white noise of known covariance $\mathbf{R} = \mathrm{diag}(1.0\mathrm{m} \;\; 1.0\mathrm{m} \;\; 0.1\mathrm{m} \;\; 0.1\mathrm{m} \;\; 0.05\mathrm{m/s})^2$.

B. Performance Evaluation

During the simulation 10 runs are executed successively. The final map parameters of run j are passed over to run $j+1$. To evaluate the absolute quality of the filter estimate the position error

$$e_{k,j} = \|\mathbf{p}_k - \hat{\mathbf{s}}_k(\hat{l}_k)\|_j \tag{38}$$

is computed at each time step k during every run j.

Additionally the Fréchet distance $d_{\mathrm{fr},j}$ between the final mean map estimate $\hat{\mathbf{s}}(l)$ and the true progression is calculated before and after each total run j, as presented in Sec. II-D.

To evaluate filter consistency the normalized innovation squared (NIS)

$$\epsilon_{k,j} = (\hat{\mathbf{z}}_k - \mathbf{h}(\hat{\mathbf{x}}_k))^{\mathrm{T}}(\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k)^{-1}(\hat{\mathbf{z}}_k - \mathbf{h}(\hat{\mathbf{x}}_k))|_j \tag{39}$$

[5] Strictly speaking the parameters l_i^{+} are also random variables. Due to the smooth, weak non-linear characteristics of (35) passing through the mean of the current curve estimate is sufficient in the presented framework.

[6] The errors that are caused during the re-sampling step are negligible compared other error sources and are therefore not regarded as single error sources.

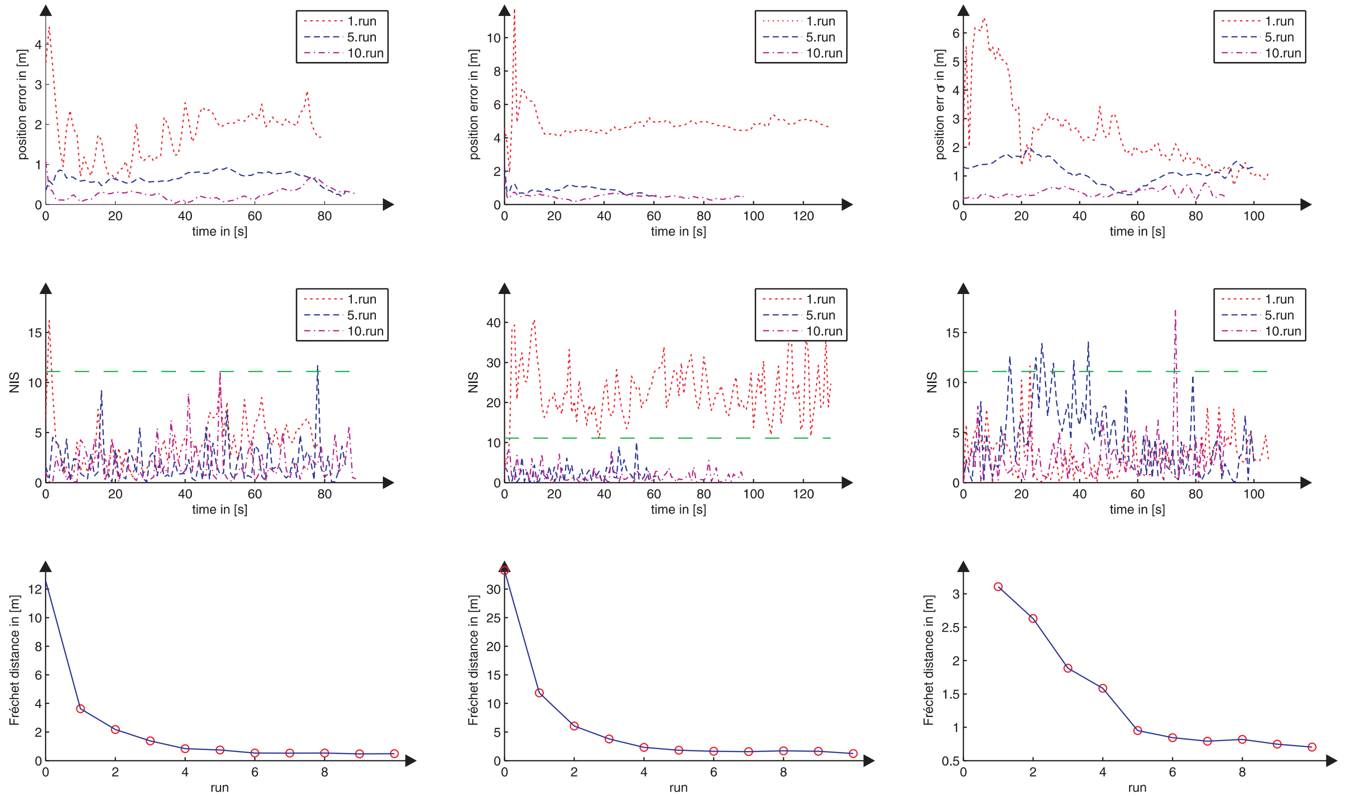


Fig. 6. Column-wise simulation results for three different filter initializations: While σ_d was always set to 0.4m/s^2 and σ_p was fixed to 7.5m during all runs, the map offset Δd additionally corrupted the true map in the second column with a value of $\Delta d = [0.0\ \ 15.0]^{\mathrm{T}}\text{m}$. In the third column no initial map was given at all. The first and the second row give respectively position errors and the normalized innovation squared (NIS) to evaluate the absolute behavior and filter consistency compared to the 95% probability region for three selected runs. The lowest row presents the Fréchet distance $d_{\mathrm{fr},j}$ after each run.

is computed [1]. Under the hypothesis that the filter is consistent for $n_{\mathbf{z}} = 5$ degrees of freedom, where $n_{\mathbf{z}}$ is the dimension of the measurement vector $\mathbf{z}$, the normalized innovation squared ϵ should be a chi-squared random variable. During a single test run a maximal sum of 6 out of 100 values are allowed to be outside the 95% probability region, while the 5%-tail point is approximately $\chi^2_5(0.95) = 11.1$ in that case.

Fig. 6 depicts the simulation results for three different scenarios columnwise. During the presented scenarios we set $\sigma_d = 0.4\text{m/s}^2$ for fixed sampling time $T = 1\text{s}$. The first row gives the resulting vehicle position error $e_{k,j}$ for three runs out of a total number of 10 and in the second row the NIS values again for the same runs. In the third row the Fréchet distance is plotted before and after each run.

In the first scenario, presented in the first column, the true map was corrupted according to (37) with $\Delta p = [0.0\ \ 0.0]^{\mathrm{T}}\text{m}$ and $\sigma_p = 7.5\text{m}$. Caused by the sequential use of the map an improvement of the geometric map information yields an increase of the vehicle localization accuracy while less than 3 NIS values fall outside the 95% region, which is acceptable for a consistent estimator. During the second scenario a constant noise with a standard deviation of $\sigma_m = 7.5\text{m}$ and additionally a map bias with $\Delta p = [0.0\ \ 15.0]^{\mathrm{T}}\text{m}$ corrupts the initial map estimate according to (37). Caused by the biased initialization nearly all NIS values during the first run fall outside the 95% region. The sequential use of the PCM compensates that initial map bias $\Delta\mathbf{p}$ and a consistent behavior renders possible in later runs. The Fréchet distance decreases and in parallel the map estimate converges towards the true curve progression. Simultaneously, the resulting vehicle localization accuracy improves continuously and filter consistency is obtained in addition due to the increasing map quality. In the last scenario an exploration of a totally unknown area is simulated. The filter is still consistent and the improvement of the map causes a better performance during a sequential use of new map segment as shown in the third column of Fig. 6.

Tab. I and Tab. II present the results of several additional scenarios. In particular the time average position error $\overline{e}_j$ and the time average normalized innovation squared $\overline{\epsilon}_j$ are given for the initial and the final run of each series of runs over a map segment. Additionally, the improvement in the map estimate is evaluated by the initial and the final Fréchet distance $d_{\mathrm{fr},j}$.

The filter performance is always reasonable during the evaluated scenarios. Even for changing initial map errors and vehicle dynamic behaviors as presented in Tab. I the final map estimate always converges towards the expected approximation error. In the majority of cases a decrease of all recorded position errors can be observed, while exceptions are caused by randomly generated measurement series. Caused by a refinement of the available PCM the corresponding map uncertainty decreases, what results in a slight rise of the calculated NIS values, in some cases. For biased map initializations in Tab. II the filter is typically mismatched during the

TABLE I
SCENARIOS WITH NOISY INITIAL MAPS

		$\sigma_p = 5.0\text{m}$ $\Delta\mathbf{p} = [0.0\ \ 0.0]^{\mathrm{T}}\text{m}$			$\sigma_p = 7.5\text{m}$ $\Delta\mathbf{p} = [0.0\ \ 0.0]^{\mathrm{T}}\text{m}$			$\sigma_p = 10.0\text{m}$ $\Delta\mathbf{p} = [0.0\ \ 0.0]^{\mathrm{T}}\text{m}$		
		$\bar{e}_j$	$\bar{\epsilon}_j$	$d_{\mathrm{fr},j}$	$\bar{e}_j$	$\bar{\epsilon}_j$	$d_{\mathrm{fr},j}$	$\bar{e}_j$	$\bar{\epsilon}_j$	$d_{\mathrm{fr},j}$
$\sigma_d = 0.2\text{m/s}^2$	initial run $j=1$	1.2600	1.2962	17.6207	1.2238	1.4105	11.0147	2.1793	5.0963	31.8155
	final run $j=10$	2.4958	2.5781	0.6718	2.4046	3.4067	0.2998	1.1566	2.2959	0.6018
$\sigma_d = 0.4\text{m/s}^2$	initial run $j=1$	1.1477	1.3480	12.9426	1.8087	3.9980	12.5567	2.3977	1.2219	26.0882
	final run $j=10$	0.2926	2.1919	0.2226	0.2641	1.8637	0.4894	0.3065	1.7507	0.3954
$\sigma_d = 0.6\text{m/s}^2$	initial run $j=1$	2.0554	1.4505	9.9086	1.8840	1.5053	11.5093	3.1673	3.4365	12.6254
	final run $j=10$	1.6610	2.8705	0.4641	0.3833	1.9838	0.3956	0.4900	2.0543	0.4218

TABLE II
SCENARIOS WITH NOISY AND BIASED OR WITHOUT INITIAL MAP

		$\sigma_p = 7.5\text{m}$ $\Delta\mathbf{p} = [0.0\ \ 10.0]^{\mathrm{T}}\text{m}$			$\sigma_p = 7.5\text{m}$ $\Delta\mathbf{p} = [0.0\ \ 15.0]^{\mathrm{T}}\text{m}$			- -		
		$\bar{e}_j$	$\bar{\epsilon}_j$	$d_{\mathrm{fr},j}$	$\bar{e}_j$	$\bar{\epsilon}_j$	$d_{\mathrm{fr},j}$	$\bar{e}_j$	$\bar{\epsilon}_j$	$d_{\mathrm{fr},j}$
$\sigma_d = 0.2\text{m/s}^2$	initial run $j=1$	4.8810	34.8544	26.8952	5.5775	20.6083	29.9128	1.4720	3.0160	-
	final run $j=10$	0.8518	2.6901	1.3118	0.5353	2.7855	1.8158	0.2301	2.1650	0.3136
$\sigma_d = 0.4\text{m/s}^2$	initial run $j=1$	1.7791	1.7421	21.5615	0.7214	1.2308	26.9683	2.4969	2.5932	-
	final run $j=10$	1.1116	2.3694	0.5402	0.6638	1.8489	0.8086	0.4105	2.4384	0.7038
$\sigma_d = 0.6\text{m/s}^2$	initial run $j=1$	0.9455	1.5500	19.5185	4.8275	22.6060	33.2470	4.3616	35.9339	-
	final run $j=10$	0.6500	2.1626	0.3588	0.4702	1.7745	1.2414	2.8359	13.8504	6.6901

first runs but still ends up in a consistent behavior while the map quality improves more slowly compared to the unbiased scenarios. Exploration scenarios with a dynamic behavior up to $\sigma_d = 0.4\text{m/s}^2$ show a reasonable filter performance. For larger values of σ_d model limitations result in an inconsistent filter performance. All over the obtained performance indicates that the Gaussian assumptions of the relevant quantities seem to be reasonable for a system according to the specifications given.

In summary, the simulations show that the proposed filter handles typical kinds of map initializations errors successfully. In case of noisy or biased map scenarios the performance is convincing, due to the steadily increasing localization accuracy. Even in absence of an initial map available filter performance is still satisfying up to a certain degree of dynamic vehicle behavior.

VI. EXPERIMENTS

A real world railway experiment is processed to validate filter performance in real world and to quantify the influence of modelling errors. The motion of rail vehicles is tightly constrained on their tracks. Hence, the proposed method is ideally suited for railway scenarios. Compared to an automotive environment, railway systems allow to validate the proposed PCM assisted localization strategy for two main reasons: Caused by the mechanical track guiding the accuracy of track following while repeatedly passing a certain track segment is very high and driver- or situation-dependent lateral offsets are nonexistent. Additionally, a high precision reference map is available for the chosen test area [32]. Based on that ground truth the deviation between the estimate and the true track is computed.

A. Experimental Setup

An integrated navigation system [1] is mounted in the gravity center of the test vehicle. It consists of two main components: The first component is a dead-reckoning inertial navigation system (INS). Altogether it contains six sensors, namely three accelerometers and three gyroscopes, that measure acceleration and rotational rate of the vehicle. The second component is a GPS receiver to cope with the unbounded growing position drift of the INS. For details about the underlying principles and algorithms see [33] or [34].

The navigation solution and their covariances are converted to Universal Transverse Mercator (UTM) coordinate system, a two dimensional Cartesian coordinate system [33] [35]. Based on the transformed output of the integrated navigation system the presented estimation is computed.[7]

B. Performance Evaluation

During the experiment three map segments of varying length in between 2 and 4 kilometers are chosen and 12 runs are

[7]In general an expansion of the SLAM states with the states of the integrated navigation system might enhance the performance of the proposed method. In order to emphasize the possibility to replace that particular sensor system we kept both systems separated.

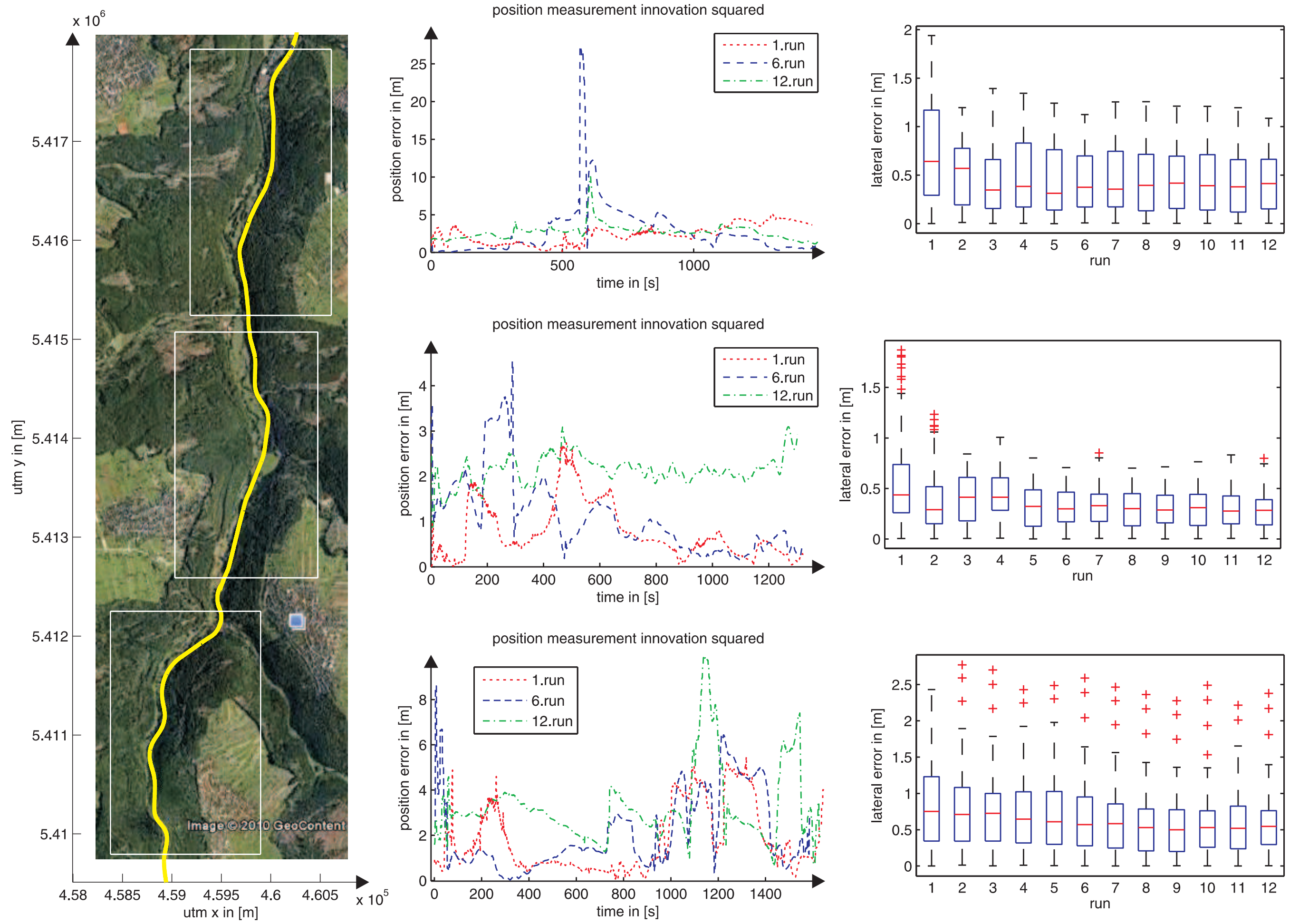


Fig. 7. Row-wise simulation results for three different test areas, plotted in the left column. In the second column the squared position innovation for three selected run are visualized exemplarily and in the third column the lateral geometric map errors are subsumed in a box plot.

executed successively. Again the final map estimate of run j is passed over to run $j+1$.

The results for three different map segments are visualized row-wise in Fig. 7. In the first column a map gives the exact test areas in UTM coordinates. In the second column the resulting squared measurement residuals for three selected runs are presented and in the third column a box plot visualizes the signed map error for sampled positions along the final estimate after each run.

Because all three test areas are situated in a densely wooded mountainous region, challenging GPS conditions result from multi-path and shadowing effects. Due to imperfect error models within each available integrated navigation system this causes an underestimation of the covariances of the estimated navigation solution. As a consequence the resulting geometric map errors slightly exceed the theoretical values computed in Sec. II-D and the simulative results presented in Sec. V. Caused by local (e.g. 6. run in area 1) or global (e.g. 12. run in area 2) mismatched outputs of the integrated navigation solution the squared position innovation is biased in the corresponding time intervals. Presumably, the true position error was smaller because of the precise map information, that was already available during the mentioned runs.

Overall, the geometric map quality obtained is sufficient to enable precise localization even in situations with moderate accuracy of the available navigation solution. Moreover, the map estimate converged to its final precision already after 3 to 4 runs and additional runs do not improve the results any further.

VII. Conclusion

The main contribution of this proposal is a novel simultaneous localization and mapping strategy for moving vehicles in case of tightly curve-constrained, i.e. 1D, motion. In that context curvemaps offer a wide range of invaluable information; hence they are a key component of the presented map assisted localization method.

The proposed probabilistic curvemap (PCM) enables an explicit representation of geometric map information and its local uncertainty. Combined with a 1D model of the vehicle kinematics in curve coordinates a simultaneous localization and mapping without additional map matching renders possible: As soon as a new observation is available, a simultaneous update of the kinematic vehicle state and the geometric map parameters is carried out in one filter step. Due to negligible nonlinearities in the observation equations an extended

Kalman filter is appropriate and yields consistent performance. During subsequent time and measurement updates and in particular during sequential use of the corrected PCM, the adapted map improves the localization accuracy step by step. Moreover PCM re-sampling is implemented to guarantee constant interpolating properties. A tangential extrapolation model generates new map segments in situations with incomplete maps. The update strategy and the chosen global spline curves yield a dense covariance matrix of the combined vehicle and map state.

The presented PCM-assisted localization framework is useful for a broad variety of scenarios in intelligent transportation systems and autonomous vehicles. Different sets of available sensor information with varying characteristics can be processed. Experiments in simulations as well as in real world railway vehicles demonstrate the method allows for iterative mapping of railway tracks with a precision of some 10cm with a vehicle position error of typically less than one meter. The focus of further work will be on other measurement principles, additional motion models and observations. Moreover, the framework will be extended from single segments to road- or track networks, multi lane road scenarios and the arising vehicle-to-lane association problem.

Appendix A
Matrix Calculations

The linear matrix equations in (14) are derived from the conditions in (4) to (7) and the system of equations in (8). Initially the moments vector $\mathbf{m}_x = [m_{x,0} \ \dots \ m_{x,n}]^{\mathrm{T}}$ is defined. Arranging the $4n$ equations in (4) to (7) in matrix notations yields the following terms

$$\begin{aligned} \mathbf{a}_x &= \mathbf{A}_1\mathbf{q}_x & \mathbf{b}_x &= \mathbf{B}_1\mathbf{q}_x + \mathbf{B}_2\mathbf{m}_x \\ \mathbf{c}_x &= \mathbf{C}_1\mathbf{m}_x & \mathbf{d}_x &= \mathbf{D}_1\mathbf{m}_x \end{aligned} \tag{40}$$

while the matrices are given by

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{B}_1 = \begin{bmatrix} \frac{-1}{h_1} & \frac{1}{h_1} & 0 & \cdots & 0 \\ 0 & \frac{-1}{h_2} & \frac{1}{h_2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{-1}{h_{n-1}} & \frac{1}{h_{n-1}} \end{bmatrix}$$

$$\mathbf{B}_2 = \begin{bmatrix} \frac{-2h_1}{6} & \frac{-h_1}{6} & 0 & \cdots & 0 \\ 0 & \frac{-2h_2}{6} & \frac{-h_2}{6} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{-2h_{n-1}}{6} & \frac{-h_{n-1}}{6} \end{bmatrix}$$

$$\mathbf{C}_1 = \begin{bmatrix} \frac{1}{2} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{2} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{1}{2} & 0 \end{bmatrix}$$

$$\mathbf{D}_1 = \begin{bmatrix} \frac{-1}{6h_1} & \frac{1}{6h_1} & 0 & \cdots & 0 \\ 0 & \frac{-1}{6h_2} & \frac{1}{6h_2} & \cdots & 0 \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{-1}{6h_{n-1}} & \frac{1}{6h_{n-1}} \end{bmatrix}.$$

Completing the system of equations in (8) with the determined values for the moments at the spline endings $m_{x,0} = m_{x,n} = 0$ and rearranging the n equations in matrix notation yields

$$\mathbf{M}\mathbf{m}_x = \mathbf{L}\mathbf{q}_x \tag{41}$$

with the two matrices

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ h_1 & 2(h_1+h_2) & h_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ \frac{6}{h_1} & (\frac{-6}{h_1}+\frac{-6}{h_2}) & \frac{6}{h_2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \frac{6}{h_{n-2}} & (\frac{-6}{h_{n-2}}+\frac{-6}{h_{n-1}}) & \frac{6}{h_{n-1}} \\ 0 & \cdots & 0 & 0 & 0 \end{bmatrix}.$$

Because the matrix $\mathbf{M}$ is tridiagonal the system of equations in (41) can be solved efficiently with a computational effort that is proportional to the total number of equations [36]. The solution

$$\mathbf{m}_x = \mathbf{M}^{-1}\mathbf{L}\mathbf{q}_x \tag{42}$$

can be inserted in (4) to (7). Rearranging yields linear matrix relations

$$\begin{aligned} \mathbf{a}_x &= \mathbf{A}_1\mathbf{q}_x = \mathbf{A}\mathbf{q}_x & (43) \\ \mathbf{b}_x &= \mathbf{B}_1\mathbf{q}_x + \mathbf{B}_2\mathbf{m}_x = \mathbf{B}_1\mathbf{q}_x + \mathbf{B}_2(\mathbf{M}^{-1}F)\mathbf{q}_x \\ &= (\mathbf{B}_1 + \mathbf{B}_2\mathbf{M}^{-1}\mathbf{F})\mathbf{q}_x = \mathbf{B}\mathbf{q}_x & (44) \\ \mathbf{c}_x &= \mathbf{C}_1\mathbf{m}_x = \mathbf{C}_1(\mathbf{M}^{-1}\mathbf{F})\mathbf{q}_x = \mathbf{C}\mathbf{q}_x & (45) \\ \mathbf{d}_x &= \mathbf{D}_1\mathbf{m}_x = \mathbf{D}_1(\mathbf{M}^{-1}\mathbf{F})\mathbf{q}_x = \mathbf{D}\mathbf{q}_x & (46) \end{aligned}$$

to compute the unknown spline polynomial coefficients based on the supporting points $\mathbf{q}_x$.

Acknowledgment

The authors would like to thank the German Federal Ministry of Economics and Technology (BMWi), which partially supported this work within the research project DemoOrt. Thank also goes to Karlsruher Verkehrsbetriebe (KVV) for their support to execute the test runs.

REFERENCES

[1] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. John Whiley and Sons, 2001.

[2] S. Julier and H. Durrant-Whyte, "On The Role of Process Models in Autonomous Land Vehicle Navigation Systems," *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 1–14, 2003.

[3] T. Kirubarajan, Y. Bar-Shalom, K. R. Pattipati, I. Kadar, B. Abrams, and E. Eadan, "Tracking Ground Targets with Road Constraints using IMM Estimator," in *Aerospace Conference*, 1998.

[4] M. Ulmke and W. Koch, "Road-Map Assisted Ground Moving Target Tracking," *IEEE Transactions on Aerospace and Electric Systems*, vol. 42, pp. 1264–1274, 2006.

[5] I. Skog and P. Händel, "In-Car Positioning and Navigation Technologies - A Survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, pp. 4–21, 2009.

[6] C. Yang and E. Blasch, "Fusion of Tracks with Road Constraints," *Journal of Advances in Information Fusion*, vol. 3, pp. 14–32, 2008.

[7] M. Quddus, W. Ochieng, and R. Noland, "Current map-matching Algorithms for transport application: State-of-the art and future research directions," *Transportation Research Part C*, vol. 15, no. 5, pp. 312–328, 2007.

[8] P. Shea, T. Zadra, D. Klamer, E. Frangione, and R. Brouillard, "Improved State Estimation through Use of Roads in Ground Tracking ," in *SPIE: Signal and Data Processing of Small Targets*, 2000.

[9] C. Yang, M. Bakich, and E. Blasch, "Nonlinear Constrained Tracking of Targets on Roads," in *International Conference on Information Fusion*, 2005.

[10] B. Pannetier, K. Benameur, V. Nimier, and M. Rombaut, "VS-IMM using Road Map Information for a Ground Target Tracking," in *8. International Conference on Information Fusion*, 2005.

[11] C. Agate and K. Sullivan, "Road-Constrained Target Tracking and Identification using a Particle Filter," in *SPIE: Signal and Data Processing of Small Targets*, 2003.

[12] D. Bétaille and R. Toledo-Moreo, "Creating Enhanced Maps for Lane-Level Vehicle Navigation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, pp. 786–798, 2010.

[13] A. Wedel, H. Badino, C. Rabe, H. Loose, U. Franke, and D. Cremers, "B-Spline Modeling of Road Surfaces With an Application to Free-Space Estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, pp. 572–583, 2009.

[14] R. Toledo-Moreo and M. Zamora-Izquierdo, "IMM-Based Lane-Change Prediction in Highways With Low-Cost GPS/INS," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, pp. 180–185, 2009.

[15] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge: The MIT Press, 2005.

[16] H. Durrant-Whyte and T. Bailey, "Simultaneous Localization and Mapping (SLAM): Part 1," *IEEE Robotics and Automation Magazine*, vol. 2, pp. 99–108, 2006.

[17] M. Gamini Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A Solution to the Simultaneous Localization and Map Building (SLAM)Problem," *IEEE Transactions on Robotics and Automation*, vol. 3, pp. 229–241, 2001.

[18] D. Schleicher, L. Bergasa, M. Ocaña, R. Barea, and M. López, "Real-Time Hierarchical Outdoor SLAM Based on Stereovision and GPS Fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, pp. 440–452, 2009.

[19] C. Hasberg and S. Hensel, "Online Estimation of Road Map Elements using Spline Curves," in *11. International Conference on Information Fusion*, 2008.

[20] G. Farin, *Curves and Surfaces for CAGD*. Morgan Kaufmann, 2002.

[21] G. Knott, *Interpolating Cubic Splines*, vol. 18 of *Progress in Computer Science and Applied Logic*. Boston: Birkhaeuser, 2000.

[22] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh, *The Theorie of Splines and their Application*, vol. 38 of *Mathematics in Science and Engineering*. New York: Academic Press, 1967.

[23] C. de Boor, *A practical guide to splines*. Springer Verlag, 2001.

[24] H. Wang, J. Kearney, and K. Atkinson, *Curve and Surface Design: Saint Marlo 2002*, ch. Arc-Length Parameterized Spline Curves for Real-Time Simulation, pp. 387–396. Nashboro Press, 2002.

[25] M. Floater and T. Surazhsky, *Topics in Multivariate Approximation and Interpolation*, ch. Parameterization for curve interpolation, pp. 101–115. Elsevier B. V., 2005.

[26] D. Brunn and U. Hanebeck, "A Model-based Framework for optimal Measurements in Machine Tool Calibration," in *Proceedings of IEEE International Conference of Robotics and Automation*, 2005.

[27] T. Eiter and H. Mannila, "Computing Discrete Fréchet Distance," tech. rep., Christian Doppler Labor für Expertensysteme, Technische Universität Wien, 1994.

[28] C. Hasberg, S. Hensel, W. M., and K. Bach, "Integrating Spline Curves in Road Constrained Object Tracking," in *11. International Conference on Intelligent Transportation Systems*, 2008.

[29] C. Bishop, *Pattern Recognition and Machine Learning*. Information Science and Statistics, 2006.

[30] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley, 2006.

[31] P. Maybeck, *Stochastic Models, Estimation and Control. Bd. 1*, vol. 141 of *Mathematics in Science and Engineering*. New York: Academic Press, 1979.

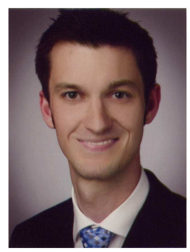
[32] F. Böhringer and A. Geistler, "Location in railway traffic: Generation of a digital map for secure applications," in *Computers in Railways X*, (Southampton), pp. 459–468, WIT Press, 2006.

[33] D. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, vol. 207. Reston: American Institute of Aeronautics and Astronautics, 2 ed., 2004.

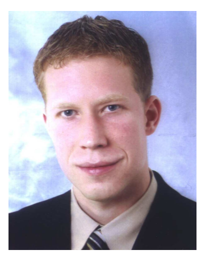
[34] M. Grewal, L. Weill, and A. Andrews, *Global Positioning Systems, Inertial Navigation and Integration*. New York: John Wiley & Sons, 2007.

[35] J. Farrell and M. Barth, *The global positioning system and inertial navigation*. McGraw-Hill, 1999.

[36] G. Marcuk, *Methods of numerical mathematics*. Springer, 1975.



Carsten Hasberg (S10) received the Diploma in electrical engineering at the University of Karlsruhe, Germany, in 2006. He is currently working at the Institute of Measurement and Control Systems, Karlsruhe Institute of Technology (KIT). His research interests include environment modeling, mapping and localization methods for autonomous systems.



Stefan Hensel (S09) received the Diploma in mechanical engineering at the University of Karlsruhe, Germany, in 2006. His research interest lies in statistical signal processing and pattern recognition with applications in rail vehicle localization. He is currently working at the Institute of Measurement and Control Systems, Karlsruhe Institute of Technology (KIT).

Christoph Stiller (S93M95SM99) studied electrical engineering at the Universities in Aachen, Germany, and Trondheim, Norway. He received the Dr.-Ing. degree (Ph.D.) with distinction from Aachen University in 1994. He worked as postdoc at INRS-Telecommunications, Montreal, QC, Canada, and in the advanced development for Robert Bosch GmbH, Hildesheim, Germany. In 2001, he became a chaired Professor and Head of the Institute of Measurement and Control Systems, Karlsruhe Institute of Technology (KIT), Germany. His present interests cover cognition of mobile systems, computer vision, and real-time applications thereof. He is the author or coauthor of more than 100 publications and patents in these fields. Dr. Stiller is Vice President Publications of the IEEE Intelligent Transportation Systems Society. He served as Associate Editor for the IEEE Transactions on Image Processing (1999-2003) and, since 2004, he has served as an Associate Editor for the IEEE Transactions on Intelligent Transportation Systems. Since January 2009, he is Editor-in-Chief of the IEEE Intelligent Transportation Systems Magazine.