# Snapshot Estimator for Magnetic Field-based Train Localization with Uncalibrated Magnetometers

Benjamin Siebler*, Andreas Lehner*, Stephan Sand*, and Uwe D. Hanebeck†
*German Aerospace Center (DLR), Institute of Communications and Navigation
†Karlsruhe Institute of Technology (KIT), Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
benjamin.siebler@dlr.de

***Abstract*—Magnetic field-based train localization uses distortions of the Earth magnetic field caused by magnetic material in the vicinity of the tracks. From the distortions, a position estimate can be obtained by comparing magnetometer measurements to a map of the magnetic field. The comparison to the map requires the calibration of the train-mounted magnetometer. However, calibration is difficult because common calibration techniques require a rotation of the magnetometer and the platform it is mounted on. Thus, in this paper we propose a maximum likelihood (ML) snapshot estimator that simultaneously estimates the calibration parameters and the train position. The snapshot estimator reduces the ML estimation problem to a one-dimensional optimization problem that can be solved efficiently. To show the feasibility of the approach, an evaluation is carried out based on measurements recorded with a train driving through Berlin.**

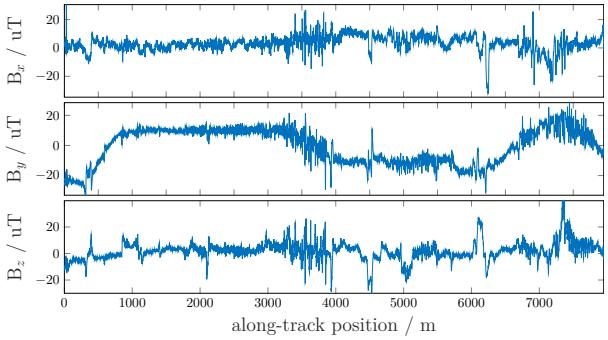


Figure 1. Three components of the magnetic vector field along a ≈ 8 km railway track.

## I. Introduction

The automation of railway traffic requires accurate absolute positions of all trains in the track network. These positions must be provided also in environments where global navigation satellite system (GNSS) signals are blocked or degraded, such as tunnels and urban canyons. In this paper, we therefore propose a position estimator that can provide this information also in GNSS-denied environments.

The proposed position estimator is based on local distortions of the Earth magnetic field. The distortions are a result of the interaction of the Earth magnetic field with magnetic material in the vicinity of railway lines. Most of this material is steel found in the rails, the signal poles, and in reinforced concrete in close by buildings. Since the material is static, also the caused distortions are static, allowing us to build a digital map of the magnetic field, where the map is a function that returns for each position the corresponding magnetic field. An example for a magnetic map of a ≈ 8 km long railway track is shown in Fig. 1.

The idea of magnetic localization is not new and its feasibility has been shown in indoor environments [1], [2], on roads [3], in the air [4], and for railways [5]. For magnetic localization, magnetometer calibration is crucial because otherwise the measured magnetic field cannot be matched accurately to the map and each train would need its own custom map. For trains, magnetometer calibration is particularly difficult since common calibration methods, e.g., [6], [7], require the magnetometer to be rotated in a homogeneous magnetic field while mounted on the platform on which it will be used. Thus, in [8] we showed that the magnetometer calibration can be carried out simultaneously with the position estimation.

The proposed simultaneous localization and calibration (SLAC) algorithm from [8] is based on a Rao-Blackwellized particle filter and achieved meter-level accuracy during an evaluation with real train data. However, the recursive nature of the filter requires an initial guess of the train position with a uncertainty of a few tens of meters. In this paper, we now propose a novel SLAC snapshot estimator that can cope with much higher uncertainties and thus can be used to either provide an initial estimate for the particle filter or can be used as a standalone localization system.

## II. Methodology

### A. Magnetometer Sensor Model

The magnetometer sensor model for calibration is given by

$$\mathbf{z}_k = \mathbf{C}\,\mathbf{m}(s_k) + \mathbf{b} + \mathbf{n}_k \ , \tag{1}$$

where $\mathbf{z}_k \in \mathbb{R}^3$ is the measured magnetic vector at time step $k$ and $\mathbf{m}(s_k) \in \mathbb{R}^3$ is the magnetic map of the railway track. The magnetic map relates the along-track train position $s_k \in [0, L]$, with track length $L$, to the magnetic field vector. The remaining parts in (1) are the calibration matrix $\mathbf{C} \in \mathbb{R}^{3\times3}$, the bias vector $\mathbf{b} \in \mathbb{R}^3$ and the sensor noise $\mathbf{n}_k \in \mathbb{R}^3$. The calibration matrix accounts for the interaction of soft iron material close to the sensor with the magnetic vector field which can scale and rotate the magnetic field. The bias vector accounts for hard iron effects caused by magnetized material close to the sensor. For the sensor, noise zero mean

white Gaussian noise is considered with a diagonal covariance matrix $\mathbf{n}_k \sim \mathcal{N}(0, \sigma_n^2 \mathbf{I}_{3\times 3})$.

For the derivation of the snapshot estimator the sensor model is written in terms of a single parameter vector $\boldsymbol{\theta} \in \mathbb{R}^{12}$

$$\mathbf{z}_k = \mathbf{H}(s_k)\boldsymbol{\theta} + \mathbf{n}_k \ , \tag{2}$$

with the matrix valued function

$$\mathbf{H}(s_k) = \begin{bmatrix} \begin{bmatrix}\mathbf{m}^\top(s_k) & 1\end{bmatrix} & \mathbf{0}_{1\times 4} & \mathbf{0}_{1\times 4} \\ \mathbf{0}_{1\times 4} & \begin{bmatrix}\mathbf{m}^\top(s_k) & 1\end{bmatrix} & \mathbf{0}_{1\times 4} \\ \mathbf{0}_{1\times 4} & \mathbf{0}_{1\times 4} & \begin{bmatrix}\mathbf{m}^\top(s_k) & 1\end{bmatrix} \end{bmatrix}^\top . \tag{3}$$

The parameter vector contains the nine elements of the calibration matrix and the three elements of the bias vector

$$\boldsymbol{\theta} = \begin{bmatrix}[\mathbf{C}]_{1:} & [\mathbf{b}]_1 & [\mathbf{C}]_{2:} & [\mathbf{b}]_2 & [\mathbf{C}]_{3:} & [\mathbf{b}]_3\end{bmatrix}^\top \ , \tag{4}$$

where $[\mathbf{C}]_{i:}, i \in \{1,2,3\}$ are the rows of the calibration matrix and the $[\mathbf{b}]_i, i \in \{1,2,3\}$ are the elements of the bias vector.

From (2) it can be seen that there is a linear relationship between the measurement vector $\mathbf{z}_k$ and the calibration parameters $\boldsymbol{\theta}$. In the following a linear estimator for the calibration parameter under the assumption of known train position is derived that then will be extended to SLAC.

### B. Magnetometer Calibration with Known Positions

With known train positions and sensor model (2), calibration can be achieved with a maximum likelihood (ML) estimator. For the ML estimator, we stack $N$ magnetometer measurements into a single long vector

$$\mathbf{z}_{1:N} = \begin{bmatrix}\mathbf{z}_1^\top & \cdots & \mathbf{z}_N^\top\end{bmatrix}^\top \ . \tag{5}$$

Similarly, the corresponding right-hand side of the sensor model is stacked into the matrix

$$\mathbf{H}(s_{1:N}) = \begin{bmatrix}\mathbf{H}(s_1)^T & \cdots & \mathbf{H}(s_N)^\top\end{bmatrix}^\top \tag{6}$$

and the noise vector is given by

$$\mathbf{n}_{1:N} = \begin{bmatrix}\mathbf{n}_1^\top & \cdots & \mathbf{n}_N^\top\end{bmatrix}^\top \ . \tag{7}$$

With (5)–(7), the linear system of equations for the $N$ measurements is given by

$$\mathbf{z}_{1:N} = \mathbf{H}(s_{1:N})\boldsymbol{\theta} + \mathbf{n}_{1:N} \ . \tag{8}$$

Since the noise vector is Gaussian and white, the ML estimate of $\boldsymbol{\theta}$ is equal to the least squares (LS) estimate and thus is found by left multiplying the pseudo inverse of $\mathbf{H}_{1:N}$ with the stacked measurement vector

$$\hat{\boldsymbol{\theta}}_{\text{LS}}(s_{1:N}) = \left(\mathbf{H}(s_{1:N})^\top \mathbf{H}(s_{1:N})\right)^{-1} \mathbf{H}(s_{1:N})^\top \mathbf{z}_{1:N} \ , \tag{9}$$

where the estimated parameters $\hat{\boldsymbol{\theta}}_{\text{LS}}(s_{1:N})$ are a function of the $N$ train positions.

### C. Simultaneous Localization and Calibration (SLAC)

If we drop the assumption of known train positions, the estimation problem becomes considerably more complex due to the nonlinear relation of the magnetic field and the train position $s_k$. From a formal point of view, the ML estimate for unknown positions can be found by maximizing the joint likelihood $\mathrm{p}(\mathbf{z}_{1:N}|s_{1:N}, \boldsymbol{\theta})$ w.r.t. the positions and parameters

$$\hat{s}_{1:N}, \hat{\boldsymbol{\theta}} = \underset{s_{1:N}, \boldsymbol{\theta}}{\arg\max}\ \mathrm{p}(\mathbf{z}_{1:N}|s_{1:N}, \boldsymbol{\theta}) \ . \tag{10}$$

Finding the global maximum of the joint likelihood is difficult in practice since the nonlinear nature of the magnetic field results in multiple local maxima and the number $N$ of optimized positions can be large. The concrete number depends mainly on two parameters, the length of track segment on which the measurements $\mathbf{z}_{1:N}$ are collected and the spatial density of the measurements. Both factors cannot be adapted freely because the measurements have to cover a track segment that is not too short because this would lead to ambiguities in the position estimation. Furthermore, the measurements have to be dense enough to capture local distortions. Practical values for these two parameters will be given in Section III. In addition to the above mentioned challenges, the optimization should be computable online with a reasonable update rate.

To keep the complexity of the optimization low and to find the global maxima, we propose two simple steps. In the first step, the number of positions that are optimized is reduced to one. The second step then exploits the linear relationship of the calibration parameters and the magnetometer measurements when the train position is known.

*1) Virtual Magnetometer Array:* The reduction in the number of optimized positions is achieved with odometer measurements. The odometer provides measurements $d_{N-k}$ that describe the distance the train has traveled between the most current position $s_N$ and an earlier position $s_{N-k}$. With the measured distances we can form a virtual array of magnetometer measurements and assign each magnetometer measurement its absolute position by relating it to $s_N$

$$s_{N-k} = s_N - d_{N-k} \ . \tag{11}$$

Note, for brevity we assume here that the orientation of the train on the track is known and the sign of odometer measurements has been corrected accordingly. With (11), all positions can be computed from $s_N$ and therefore only $s_N$ has to be optimized

$$\hat{s}_N, \hat{\boldsymbol{\theta}} = \underset{s_N, \boldsymbol{\theta}}{\arg\max}\, p(\mathbf{z}_{1:N}|s_{1:N}, \boldsymbol{\theta}) \ . \tag{12}$$

*2) Conditional Linearity:* The joint optimization of the likelihood can be replaced with a composition of the optimization over the position and the optimization over the parameters

$$\hat{s}_N, \hat{\boldsymbol{\theta}} = \underset{s_N}{\arg\max} \left[ \underset{\boldsymbol{\theta}}{\arg\max}\, p(\mathbf{z}_{1:N}|s_{1:N}, \boldsymbol{\theta}) \right] \ . \tag{13}$$

The inner $\arg\max$ operator seeks to find $\boldsymbol{\theta}$ that maximizes the likelihood of the measurements for a given position. Here,

the optimization can make use of the conditional linearity of the parameters on the position and the optimization can be replaced with the closed-form LS solution $\hat{\boldsymbol{\theta}}_{\mathrm{LS}}(s_{1:N})$ given by (9). With (9), the optimization in (13) reduces to a one-dimensional optimization over the along-track train position $s_N$

$$\hat{s}_N, \hat{\boldsymbol{\theta}} = \arg\max_{s_N} p(\mathbf{z}_{1:N}|s_{1:N}, \hat{\boldsymbol{\theta}}_{\mathrm{LS}}(s_N)) \ . \tag{14}$$

With (14), we can now find the global maxima of the likelihood by evaluating the likelihood at "all" positions along the track. For each position, this requires the calculation of $\hat{\boldsymbol{\theta}}_{\mathrm{LS}}(s_{1:N})$ followed by the evaluation of the likelihood defined by the sensor model (8) and the noise probability density function

$$p(\mathbf{z}_{1:N}|s_{1:N}, \hat{\boldsymbol{\theta}}_{\mathrm{LS}}(s_N)) = \mathcal{N}\left(\mathbf{z}_{1:N}; \boldsymbol{\mu}(s_{1:N}), \boldsymbol{\Sigma}(s_{1:N})\right) \ , \tag{15}$$

where the mean and covariance are given by

$$\boldsymbol{\mu}(s_{1:N}) = \mathbf{H}(s_{1:N})\hat{\boldsymbol{\theta}}_{\mathrm{LS}}(s_{1:N}) \text{ and } \boldsymbol{\Sigma}(s_{1:N}) = \sigma_n^2 \mathbf{I}_{3N\times 3N} \ . \tag{16}$$

Because $s_k$ is restricted to an interval, a practical approach for global optimizations is to evaluate (15) for discrete along-track position placed in the whole interval. This ensures that always the whole track is considered and that the optimization does not run into local minima close to the initial position.

### D. Implementation

The optimization requires the evaluation of the likelihood (15) at multiple positions. This section provides some insights how this can be implemented efficiently. The key for a fast calculation lies in the prepossessing of all parts that are independent from the observations, the discretization of the magnetic map and the interpolation of the virtual magnetometer array data.

*1) Map Discretization:* The magnetic map is discretized on a equidistant grid with spacing $\Delta_{\mathbf{m}}$. The map can therefore be represented as an array that contains for each grid point the corresponding magnetic vector. To retrieve the magnetic vector from the map at a given position $s$, we can simply calculate the index of the closest grid point using spacing $\Delta_{\mathbf{m}}$.

*2) Interpolation of Virtual Array:* The magnetometer measurement times series is transformed into the spatial domain using the odometer measurements. The result is a virtual array with variable spacing since the distance between two measurements depends on the train speed. Then an interpolation on a fixed grid is performed to simplify the comparison of the virtual array measurement with the map. In this paper, an equidistant grid with spacing $\Delta_{\mathbf{z}}$ is selected. The transformed and interpolated time series can be seen as a magnetic "template" that represents the magnetic field along a short track segment. The interpolation is always done such that the template has a predefined length $L_T$. In practice, this means that we wait until the train has driven at least $L_T$ meters and then the interpolation is performed.

*3) Precalculations:* When the likelihood is evaluated for a position, first the calibration parameters are estimated with (9) and second, the parameters are used to obtain the mean of the likelihood with (16). This requires the stacked measurement matrix $\mathbf{H}(s_{1:N})$ from (6) and its pseudo-inverse. Fortunately, both of these matrices can be calculated offline. The offline calculation is possible because the magnetic vectors from the map required for the matrices depend only on the absolute position, and the template length and spacing. Because the latter two are fixed, only the position dependency remains. Thus, the matrices can be calculated once for each position and then stored as an additional layer in the map.

*4) Independence of Calibration Parameters:* So far all calibration parameters are stacked into a single long vector $\boldsymbol{\theta}$ to keep the notation simple, but for the actual implementation the vector will be split into three shorter vectors. The splitting is possible because the twelve parameters in $\boldsymbol{\theta}$ can be divided into three independent sets with four parameters each. The independence is due to the diagonal structure of $\mathbf{H}(s_k)$ in (3) and the independent sensor noise. For the three shorter vectors the measurements model is given by

$$[\mathbf{z}_k]_i = \begin{bmatrix} \mathrm{m}^\top(s_k) & 1 \end{bmatrix} \boldsymbol{\theta}_i + [\mathbf{n}_k]_i \ , \tag{17}$$

where the vectors $\boldsymbol{\theta}_i, \in \{1, 2, 3\}$ contain one row $[\mathbf{C}]_{i:}$ of the calibration matrix and one element $[\mathbf{b}]_i$ of the bias vector

$$\boldsymbol{\theta}_i = \begin{bmatrix} [\mathbf{C}]_{i:} & [\mathbf{b}]_i \end{bmatrix}^\top \ . \tag{18}$$

The use of three smaller vectors replaces all vector-matrix multiplications with the twelve-dimensional parameter vector with three multiplications with a four-dimensional vector. This requires less multiplications and additions, which makes it computationally more efficient.

### E. Summary of Proposed SLAC Snapshot Estimator

When the magnetic map has been generated, and all matrices $\mathbf{H}(s_{1:N})$ and pseudo-inverses have been calculated and stored, the estimation of the train position can be summarized by the following steps:

- Wait until train has driven $L_T$ meters and store the corresponding magnetometer and odometer measurements.
- Create template from the magnetometer and odometer measurements and interpolate on equidistant 1D grid.
- Evaluate the likelihood and store the result for all discrete positions in the map that are inside a given search window. The evaluation at each position requires the calculation of (9) and (15). For speeding up these computations, (17) can be used.
- Find the maximum value of the likelihood. The map position associated to this value is then the desired position estimate.

## III. Evaluation

In this section, the feasibility of the proposed algorithm is shown and its accuracy is evaluated based on a data set recorded with a diesel-electric train of the Deutsche Bahn.


Figure 2. (top) Advanced TrainLab of the Deutsche Bahn. (bottom) Track between Berlin Grunewald and Tempelhof used in the evaluation. Image data: Google Earth

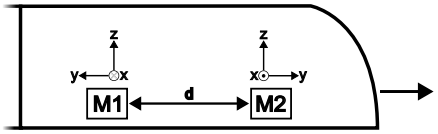


Figure 3. Magnetometer placement and orientation of sensor axes. The magnetometers are mounted inside the wagon on the floor in the center line of the track with a shift in the along-track position of $d = 0.36\,\mathrm{m}$.

### A. Measurement Campaign

The data used for the evaluation was recorded with the advanced TrainLab of the Deutsche Bahn on a $\approx 8\,\mathrm{km}$ long track in Berlin. The train was equipped with a wheel encoder, a Septentrio GNSS receiver for getting the ground truth position, and multiple low-cost Kionix KMX62 vector magnetometers. For the evaluation, the magnetic map was created based on the GNSS ground truth and one of the magnetometers. During mapping, the GNSS position was matched to a map of the railway track and the magnetometer data was linearly interpolated between the ground truth positions. Since the magnetometers measure with 200 Hz and the train speed was limited to roughly $60\,\mathrm{km\,h^{-1}}$, linear interpolation is sufficient here to obtain the densely sampled map shown in Fig. 1. The wheel encoder was mounted on a not-driven wheel and accurate odometer readings at a rate of 1 Hz were provided. The advanced TrainLab and a satellite image of the test track is shown in Fig. 2.

For evaluating the proposed SLAC estimator, the measurements of a another run on the same track, performed after the map was created, is used. Furthermore, the localization algorithm is provided measurements from a second magnetometer, different to the one used to create the map. In Fig. 3 the placement of the magnetometers used for mapping (M1) and localization (M2) is shown including the orientation of their sensor axes. The two sensor are mounted in different orientations and with a small offset of $d = 0.36\,\mathrm{m}$ in the along-track direction.

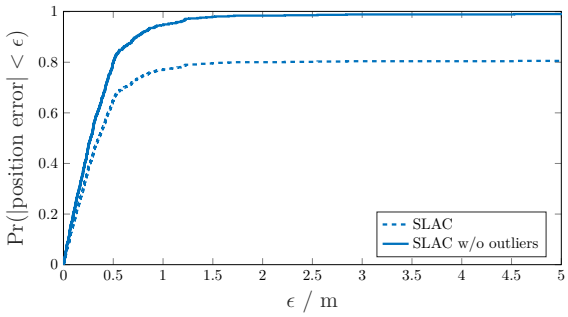


Figure 4. Position error cdf for first scenario.

### B. Evaluation Setup

During the evaluation, a template length of 50 m is used. The sampling grids of the map and the template are set to a spacing of $\Delta_{\mathbf{m}} = 0.1\,\mathrm{m}$ and $\Delta_{\mathbf{z}} = 0.3\,\mathrm{m}$, respectively. These values result in a template with 167 magnetic vectors. As search window for optimization, the whole test track is used and the likelihood is evaluated for each grid point in the map. With the map spacing of 0.1 m and with some margins at the beginning and the end, this results in 79500 evaluations of the likelihood for one position estimate. To test the estimator also for different positions, a new template and estimate is calculated when the train has moved 10 m until the end of the track is reached. With the parameters given above, our Matlab implementation needs less than 0.5 s for one position estimate while running on a laptop CPU.

In the evaluation, three scenarios are considered. In the first scenario, the estimator uses the data from M2 as it was recorded. Thus, the data is rotated relative to sensor M1 as shown in Fig. 3. In the second scenario, M2 is rotated into the frame of M1. For the third scenario, we first rotate M2 in the frame of M1 and then apply artificial calibrations parameters to it. The applied parameters have been selected such that there is a considerable scaling and cross-talk between different sensor axes. The calibration matrix and the bias vector applied on the rotated sensor data is given by

$$\tilde{\mathbf{C}} = \begin{bmatrix} 1.2 & 0.6 & -0.2 \\ 0.1 & 0.7 & -1.1 \\ -0.3 & 0.2 & 0.4 \end{bmatrix} \text{ and } \tilde{\mathbf{b}} = \begin{bmatrix} 8\,\mu\mathrm{T} \\ -15\,\mu\mathrm{T} \\ 23\,\mu\mathrm{T} \end{bmatrix} . \quad (19)$$

As a benchmark for SLAC, a second estimator was implemented based on the correlation between the map and the template. In this estimator, the correlation coefficient is calculated for all positions in the search range and for each sensor axis. To get a position estimate, the three coefficients are averaged at each position and then the position with the maximal value is selected.

### C. Results

In Fig. 4, the cumulative distribution function (cdf) of the position error is shown for the first scenario. The dashed line

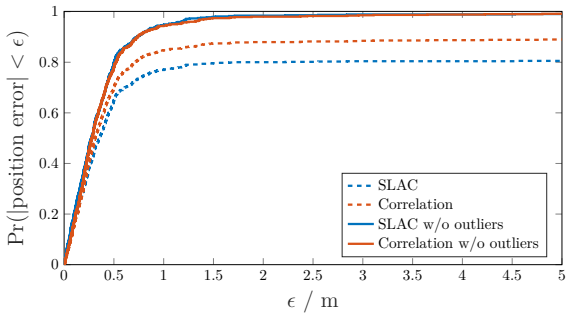


Figure 5. Position error cdf for second scenario.

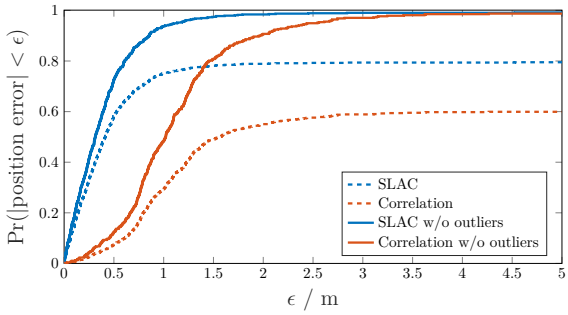


Figure 6. Position error cdf for third scenario.

is the cdf taken over all template positions along the test track and the solid line shows the cdf when the 19 % outliers are removed. Outliers are defined here as errors above 15 m. In the first scenario, the correlation has almost 99 % outliers and therefore is not shown here. The bad performance of the correlation was expected since it cannot cope with the relative orientation of M2 w.r.t. M1. Here SLAC has an advantage, since a dense calibration matrix is estimated that can handle also rotations. Overall the SLAC algorithm achieved a root mean squared error (RMSE) of 0.89 m when the outliers are not taken into account.

For the second scenario, the cdf is shown in Fig. 5. When M2 is rotated into the map frame, also the correlation provides accurate position estimates. This is the case because the rotated measurements of M2 are mainly biased and scaled compared to M1 and the correlation coefficient is invariant against scaling and translations. When the cdf is calculated over all estimates, including outliers, the correlation outperforms SLAC. This is mainly due to the lower amount of outliers which was in the range of 10 % for the correlation and again 19 % for SLAC. If outliers are not considered, shown by the solid lines in Fig. 5, the difference between the algorithms is neglectable. The RMSE without outliers of the correlation reaches 0.99 m while the SLAC RMSE is the same as in the first scenario.

The cdf for the third scenario is shown in Fig. 6. Here again SLAC has an advantage over the correlation due to the cross-talk between sensor axes introduced by the parameters from (19), which cannot be compensated by the correlation coefficient. Neglecting again the outliers, SLAC can here maintain a RMSE of 0.89 m, as in the other scenarios, while the correlation RMSE increases to 1.58 m. Furthermore, the amount of outliers for the correlation increases to 39 % while SLAC shows only a slight increase in outliers to 20 %.

In the shown cdfs, the outliers are removed on the basis of the GNSS ground truth. This is not feasible in GNSS-denied areas. Therefore, we would like to briefly discuss outlier detection using only the SLAC estimates and the odometer measurements. As a simple test statistic for outlier detection, the standard deviation over the last five estimates is calculated. For this, the odometer data is used to correct the shift between the estimates. If the estimates are accurate, they should now be close to each other and the sample standard deviation should be low. If they are still far apart, the standard deviation is high. With the comparison of the standard deviation to a threshold, we were already able to detect all outliers. The downside of this approach is that it has a false alarm rate of $\approx 17\,\%$, which reduces the rate at which valid position estimates can be provided. For train localization this is not a big issue since the odometer can predict the train position for multiple seconds while keeping the error on a meter-level accuracy.

## IV. Conclusion

This paper proposed a new snapshot estimator for magnetic field-based train localization with uncalibrated magnetometers. The estimator performs simultaneous localization and calibrations (SLAC), which enables the use of a single magnetic map for different train types. By exploiting the conditional linearity of the calibration parameters, the estimation problem could be reduced to a one-dimensional optimization problem that can be solved efficiently.

The estimator was tested with measurement data recorded on a 8 km test track in Berlin. The results show that an RMSE below one meter is achievable and a position estimate can be provided with a rate of 2 Hz.

## References

[1] M. Frassl, M. Angermann, M. Lichtenstern, P. Robertson, B. J. Julian, and M. Doniec, "Magnetic maps of indoor environments for precise localization of legged and non-legged locomotion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov. 2013, pp. 913–920.

[2] M. Kok and A. Solin, "Scalable Magnetic Field SLAM in 3D Using Gaussian Process Maps," in *2018 21st International Conference on Information Fusion (FUSION)*, Jul. 2018, pp. 1353–1360.

[3] J. A. Shockley and J. F. Raquet, "Navigation of Ground Vehicles Using Magnetic Field Variations," *Navigation*, vol. 61, no. 4, pp. 237–252, 2014.

[4] A. Canciani and J. Raquet, "Airborne Magnetic Anomaly Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 1, pp. 67–80, Feb. 2017.

[5] B. Siebler, O. Heirich, S. Sand, and U. D. Hanebeck, "Joint Train Localization and Track Identification based on Earth Magnetic Field Distortions," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Apr. 2020, pp. 941–948.

[6] V. Renaudin, M. H. Afzal, and G. Lachapelle, "Complete Triaxis Magnetometer Calibration in the Magnetic Domain," *Journal of Sensors*, 2010.

[7] M. Kok and T. B. Schön, "Magnetometer Calibration Using Inertial Sensors," *IEEE Sensors Journal*, vol. 16, no. 14, pp. 5679–5689, Jul. 2016.

[8] B. Siebler, A. Lehner, S. Sand, and U. D. Hanebeck, "Simultaneous Localization and Calibration (SLAC) Methods for a Train-Mounted Magnetometer," *NAVIGATION: Journal of the Institute of Navigation*, vol. 70, no. 1, 2023.