Machine-Learning-Based Position Error Estimation for Satellite-Based Localization Systems

Tianxiang Lan [1], *Andreas Dodinoiu* [1], *Arne Geffert* [1], *Uwe Becker* [1]

[1] *Institute for Traffic Safety and Automation Engineering, Technische Universität Braunschweig, Braunschweig, Germany*

ORCID: 0000-0002-3943-1401 (Lan), 0000-0002-8183-6816 (Dodinoiu), 0000-0001-5115-5908 (Geffert)

t.lan@tu-braunschweig.de

Abstract

Satellite-based localization systems are nowadays widely deployed in transportation, especially with the progress of global navigation satellite systems (GNSS). However, GNSS signals are easily degraded by the local environment. This compromises the accuracy of the position solution and makes it challenging to implement satellite-based localization systems in the road and railway domain. With the help of Bayes filters (e.g., Kalman and particle filter), the localization accuracy can be improved. However, these filters are constrained by assumptions, and they require accurate modeling of errors for optimal estimation. Under these circumstances, another modeling method is researched in this paper. As machine learning has become more sophisticated over the years, neural networks are now suitable for learning the relation between the position errors and the abundant information from the GNSS receiver without prior knowledge. Therefore, the dilution of precision, the elevation angle and the carrier-to-noise ratio are appropriate indicators for signal degradation. In this paper, it is shown how neural networks are trained to estimate the position error of satellite-based localization systems. For modeling the temporal correlation in position error measurements, the long short-term memory (LSTM) network is applied. Finally, it can be demonstrated that the neural networks are able to learn the trend in position errors.

Key words: satellite-based localization systems, machine learning, land vehicles, LSTM, error estimation

1 Introduction

With the ongoing advancement of global navigation satellite systems (GNSS), satellite-based localization has become more dependable for transportation applications over the last decades. However, due to terrestrial influences, the application of satellite-based localization encounters major challenges in the road and railway domain. The satellite signals are easily deflected, diffracted and obscured by buildings, trees and tunnels on the ground. Commonly arising effects are for example the multipath effect and non-line-of-sight (NLOS) signals. That degradation of satellite signals leads to a reduction in accuracy and other characteristics concerning the measurement quality of the localization systems.

Many approaches are developed to mitigate the degradation of satellite signals. For example, the Bayes estimators such as the Kalman filter and the particle filter are nowadays widely implemented in the GNSS receivers for that purpose. The Kalman filter is however restricted by the assumptions of linearity and normal distribution.

The particle filter is applicable for nonlinear systems and non-Gaussian distributions, but it comes at the expense of high computational complexity. Furthermore, the Bayes estimators require explicit modeling of the effects to be mitigated.

Multipath and NLOS reception are to a certain extent deterministic effects because they mainly exist due to surrounding objects such as buildings. Therefore, an accurate geometric modeling of the degraded signals is feasible. An example would be the ray-tracing method [1]. By analyzing the path of satellite signals, possible deflection, diffraction and obscuration can be found with a 3-D map of the surrounding area. This method is naturally very computationally expensive and its accuracy is highly dependent on the accuracy of the map as well as on the model of signal transmission.

Other methods focus on the possible indicators for the existence of multipath and NLOS in the GNSS receivers. Groves et al. propose a portfolio to mitigate signal degradation under the consideration of multiple possible indicators as well

as antenna techniques [2]. Considered indicators are for instance the carrier-to-noise ratio (C/N0) and the elevation.

Data fusion is also applied to integrate GNSS with dead reckoning methods such as inertial navigation systems (INS) and odometers. However, the two latter ones lack long-term accuracy due to error accumulation over time and hence cannot guarantee accuracy of the integrated system during GNSS outages.

To improve the measurement quality of GNSS, different signal augmentation techniques can be applied, such as real-time kinematics (RTK), Precise Point Positioning (PPP) or a hybridized version of both to combine the benefits of both approaches (PPP-RTK). RTK improves the measurement quality by not only utilizing the time information of the satellite signal, but also by using the information gained by analyzing the characteristics of the carrier phase. This is achieved by using additional information provided by a near base station (30-50 km) or by a network of base stations and advanced algorithms. Thus, RTK can provide an accuracy of up to 1 cm. In contrast to that, PPP only relies on GNSS satellite clock and orbit corrections provided by a global network via satellite or internet link in order to estimate phase characteristics and can achieve an accuracy in the decimeter level. It is a challenging limitation that the algorithms need a different period of time for the position solution to converge (the so-called position fix), i.e., from a few seconds (RTK) up to 30 minutes (PPP). The position fix can be easily lost in environments with significant amounts of signal degradation, drastically reducing the availability of such systems. [3].

However, to the knowledge of the authors, the existing methods can provide a solution to some – but not to all – issues at hand. Over the last decade, machine learning techniques have become a promising solution especially for the cases when no exact models are required for the problems to be studied [4]. In this paper, the possibility of applying machine learning as an estimator for the position error of satellite-based localization systems is explored and discussed.

2 Relevant Work

Similar to other domains, there is a growing interest in applying machine learning to satellite-based localization systems. In this chapter, relevant research is shortly introduced.

Satellite-based localization systems with GNSS and INS are nowadays commonly found. However, an INS suffers from error propagation over time and the position error from the INS can accumulate to a large level in a short period. Ma-

chine learning is applied to learn how GNSS corrects the INS error during its normal operation. The trained algorithm can then be implemented to mitigate the INS error during GNSS outages and to assure accurate localization over a longer time span using a back propagation neural network [5]. Wang et al. further suggest that a hybrid architecture with a radial basis function neural network and an extreme learning machine can improve the accuracy of prediction under dynamic conditions [6].

Machine learning is also adopted to improve the accuracy of map-matching algorithms for satellite-based localization systems. Hashemi and Karimi apply an online feedforward neural network to predict and reduce GNSS errors in order to improve the correct segmentation rate of the map-matching algorithm. The speed and horizontal dilution of precision (HDOP) are adopted as the inputs for the machine learning algorithm under the assumption that the position error is temporally autocorrelated. However, the results prove their assumption wrong. Instead, the neural network provides the best performance when the input is constant. [7].

Machine learning is also applied to aid the detection of multipath and NLOS. Hsu develops a support vector machine (SVM) to distinguish line-of-sight (LOS), multipath and NLOS signals by analyzing features such as the received signal strength, the change rate of received signal strength and the pseudorange residue [8]. He reports a classification accuracy of about 75% for certain features.

Kuratomi conducts similar research compared to this paper. He develops a position error estimator for GNSS using machine-learning-based methods such as decision trees and support vector machines. Common receiver measurements such as C/N0, elevation as well as the information coming from the navigation filter such as the innovation in the Kalman filter are applied as the inputs for the estimator [9]. Furthermore, a camera is used to detect multipath and the results are fed to the machine learning algorithms as additional inputs. Though the final results are promising, he proposes to use deep neural networks to achieve higher estimation accuracy.

3 Fundamentals

In this chapter, the sources and the propagation of GNSS errors are firstly discussed. After that, the used machine learning algorithm as well as the applied dataset are shortly introduced.

3.1 Error Sources of GNSS

The accuracy of GNSS is under the influence of many sources. The major sources are the clock error, the ionospheric and tropospheric errors as

well as multipath and NLOS. Fig. 1 illustrates that the signal paths from the satellites can be extended in the atmosphere as well as due to reflection on objects surrounding the receiver. These error sources directly cause a deviation in the pseudorange measurement corresponding to every satellite. The receiver position is then estimated by least square estimation or by the Kalman filter based on the pseudorange measurements. The relation between the pseudorange measurement of the ith satellite and the receiver position $[x_r, y_r, z_r]$ to be estimated can be expressed as

$$\rho_i = (x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2 + \delta\rho_i \qquad (1)$$

where $[x_i, y_i, z_i]$ is the satellite coordinate and $\delta\rho_i$ is the sum of pseudorange errors. It can be easily noticed that the pseudorange measurement is the sum of the geometric distance between receiver and satellite and the range errors. Moreover, the relation between pseudorange and position is nonlinear. The propagation of the pseudorange error to various localization solutions such as position, velocity and time (PVT) information is dependent on the signal geometry, which can be generally understood as the distribution of all used satellites in 3-D space. The signal geometry can be quantified by the dilution of precision (DOP) [10]. The aforementioned HDOP is a variant of the DOP and is dedicated to horizontal planes, i.e., to describe the influence of the satellite geometry on the accuracy of the horizontal position solution. A high HDOP value indicates that the pseudorange errors propagate to the horizontal plane to a great extent.
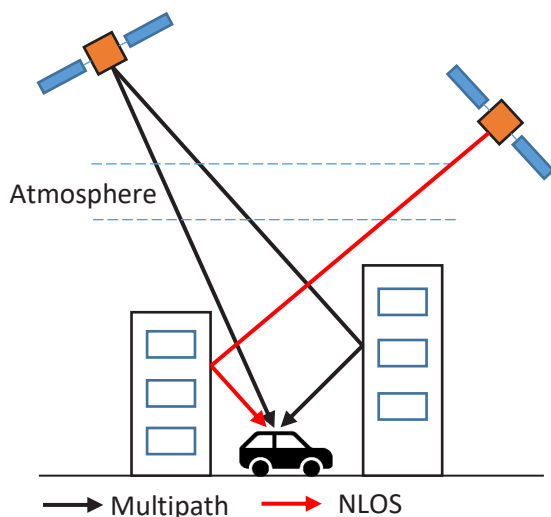

Fig. 1. Illustration of major error sources for GNSS

Several measurements of the receiver can be linked to the error sources. The elevation is the angle between the signal (from satellite to receiver) and the horizontal plane. Generally, signals from satellites of low elevation are more influenced by atmospheric effects since their paths in the atmosphere are longer than the paths of signals from high-elevation satellites. Moreover, low-elevation signals are more likely to be degraded by multipath and NLOS since the signals are more probable to be blocked and reflected [10]. C/N0, the carrier-to-noise ratio or signal-to-noise ratio, is considered to be relevant to the quality of the received signals. In urban areas, reflections happen frequently and C/N0 is observed to be lower. Hence, C/N0 is regarded as indicator for multipath and NLOS [11].

Many models have been developed to analyze the impact of error sources on pseudoranges. For ionospheric and tropospheric effects, there are sophisticated models that are especially applied to receivers of high quality. These models are complex and require current data for a given region from base stations.

For multipath and NLOS, the aforementioned ray-tracing method has already been implemented for simulation purposes in commercial software. However, the required 3-D map is not available everywhere due to high data volumes as well as due to the high complexity in generating the map. If the map is not up-to-date, it would also lead to an accuracy degradation of the ray-tracing method. Moreover, running ray-tracing in real time on vehicles would lead to high expenses on the computation hardware. Furthermore, multipath and NLOS do not simply lead to the geometric elongation of signal paths, but the resulting errors are also dependent on the signal processing inside the receiver [12].

A model to correctly describe the error propagation from diverse error sources via pseudorange measurement to the position errors is extremely difficult to establish. Even if it can be realized, it would be excessively complex with the classic modeling techniques based on linear or nonlinear equations and various assumptions on the underlying statistical distributions.

3.2 Machine Learning

The advantage of machine learning is that it does not require explicit modeling and that it can be trained to discover the model by learning the linkage between data. Due to the difficulties that classic approaches have in finding a model for the position errors of satellite-based localization systems, machine learning could be suitable to find this model based on the collected data in the field.

Generally, there are three main kinds of learning tasks with machine learning: supervised learning, unsupervised learning and reinforcement

learning. Supervised learning is targeted to learn the mapping function between the inputs and the outputs based on the input-output pairs in the training set and it is commonly applied to solve, among others, regression problems. Apart from that, unsupervised learning is often adopted to study the distribution of data and it is widely used for clustering and generating data. Reinforcement learning, however, has the feature that it can update the model during deployment, whereas for supervised and unsupervised learning, the model is determined during the training phase and remains unchanged during deployment. Since the target of this paper is to find the relation between several indicators and the position errors based on the collected data – without further deployment of the model in the field –, supervised learning is suitable.

Many models can be applied to perform a machine learning task, such as the feedforward neural network by Karimi [7]. This type of neural network is normally applied to link the relation between inputs and outputs of the same time epoch, i.e., it only considers the static behavior. However, the position errors of satellite-based localization are timely correlated, i.e., the position error at time epoch t_i can also be under the influence of error sources of previous time epochs [13]. The recurrent neural network (RNN) is specially designed for processing sequential data [14] and is therefore suitable to model time series. As shown in Fig. 2, an RNN layer with multiple cells can process a sequence. The cells not only learn the inputs, but also pass information further to the next cell.
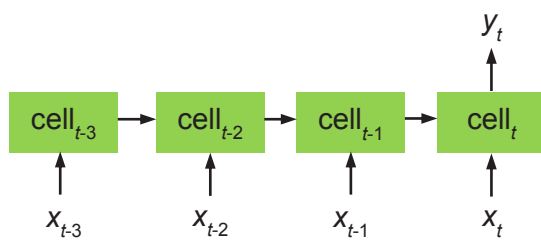

Fig. 2. Illustration of a typical many-to-one mode of RNN

The term RNN covers a class of recurrent neural networks. The most widely applied network in that class is the LSTM [15]. It is suitable for capturing both the long-term and the short-term pattern in the sequence. An LSTM network consists of input i_t, forget f_t and output gates o_t. LSTM networks use the cell state vector c_t to receive information from previous steps in the processed sequence and to pass the information further to the following steps in that sequence. Mathematically, an LSTM cell can be described using following equations

$$f_t = \sigma(W_f\, x_t + U_f\, h_{t\text{-}1} + b_f) \qquad (2)$$

$$i_t = \sigma(W_i\, x_t + U_i\, h_{t\text{-}1} + b_i) \qquad (3)$$

$$o_t = \sigma(W_o\, x_t + U_o\, h_{t\text{-}1} + b_o) \qquad (4)$$

$$c_t = f_t \odot c_{t\text{-}1} + \tanh(W_c\, x_t + U_c\, h_{t\text{-}1} + b_c) \qquad (5)$$

$$h_t = o_t \odot \sigma(c_t) \qquad (6)$$

where σ represents the sigmoid activation function and $\odot$ denotes the element-wise multiplication. The major improvement of LSTM against original RNN is that the vanishing and the exploding of gradients is avoided using the forget gate [15]. LSTM networks can be applied for machine translation [16], also for the prediction of time series [17].

3.3 Dataset

The data applied in this paper were collected in Braunschweig, Germany, on several days in August 2014. For the data collection, an automotive-grade u-blox receiver and an OxTS RT 3002 integrated system were used as the common receiver on the vehicle and as reference respectively. The test vehicle followed the designed trajectory for 20 times in total. Data collected during 5 of those 20 runs are used as dataset since they were all collected under the simplest receiver settings without additional correction services. Those settings truly represent the quality of an automotive-grade GNSS receiver. The trajectory consists of environments including urban areas, multiple bridges, forest, suburban area and highway (Fig. 3). Both the u-blox receiver and the reference system recorded data at 5 Hz frequency.

With the aim to identify suitable inputs for the machine-learning-based model, the available quantities provided by the receivers are analyzed hereinafter. Apart from the PVT information, the u-blox receiver can provide different metrics of dilution of precision, C/N0, elevation and azimuth of the used satellites. Utilizing all available C/N0 and elevation angles of the used satellites as inputs for the neural network would render the dataset too complex. Hence, those measurements are reduced to their mean values. For C/N0, another parameter is considered, i.e., the percentage of satellites with low C/N0. This is done under the assumption that if most used satellite signals are degraded by errors and indicated by lower C/N0, the estimation algorithm in the receiver can neither identify nor exclude the erroneous measurements. Thus, large deviations in the position measurements can be expected. Here, the threshold for low C/N0 is defined at 20 dB. Moreover, the model requires reference values of position errors to evaluate the
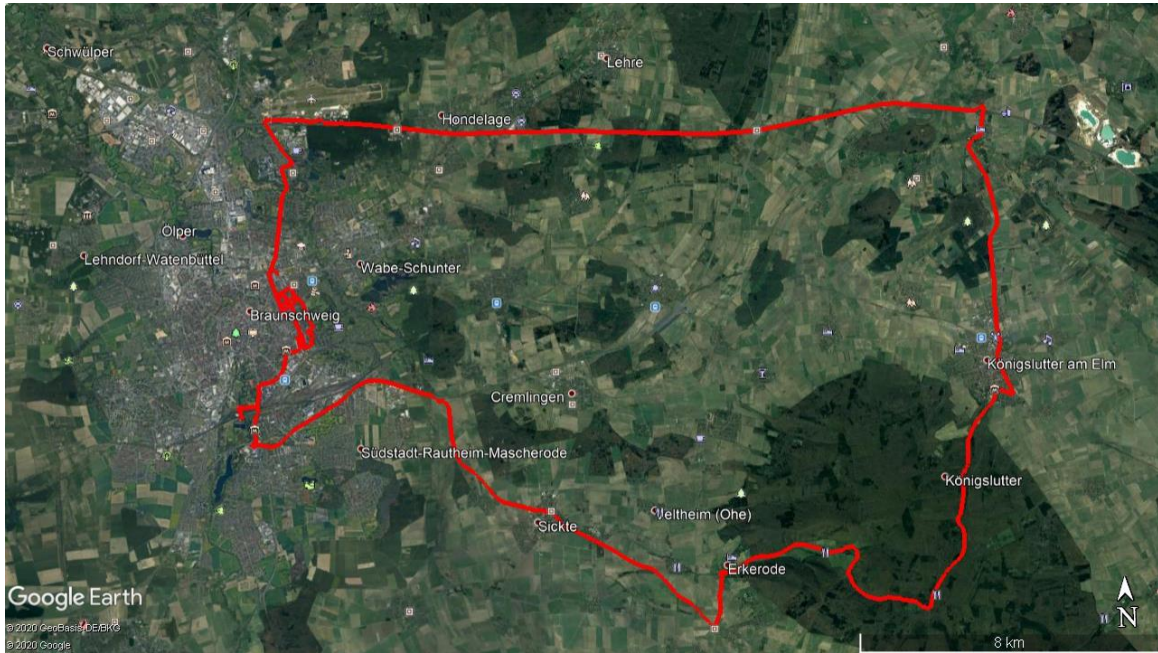
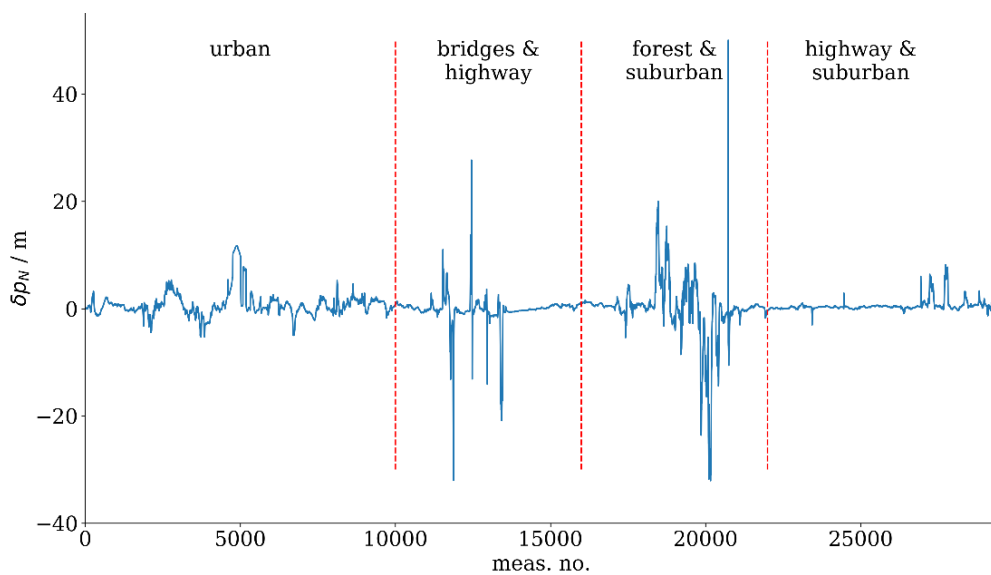*Fig. 3. Trajectory for data collection [18]*

Fig. 4. Position error in north direction collected in one run

estimation. The reference values of the position errors are calculated as the difference between the positions measured by the u-blox receiver and the reference receiver.

It can also be observed in Fig. 4 that the data for diverse environments differ from each other. Using a single model to learn all the features throughout the run would be particularly challenging. Therefore, the dataset is segmented on the basis of the aforementioned environments.

3.4 Model

Based on the understanding of position errors in the fields of satellite-based localization and ma-

chine learning, a neural network with LSTM layers is applied in this paper to model the position errors. The model consists not only of LSTM layers, but also of a fully connected linear layer. The LSTM layers are used to discover the underlying temporal correlation and the fully connected layer links that correlation to the output. Moreover, a dropout layer is inserted between the LSTM and the fully connected layers to improve regularization in order to avoid overfitting. The architecture of the model is depicted in Fig. 5.

Specifically, the following measurements are adopted as the inputs for the neural network:

- number of used satellites
- HDOP

- mean value of C/N0 of used satellites
- mean value of elevation angle of used satellites
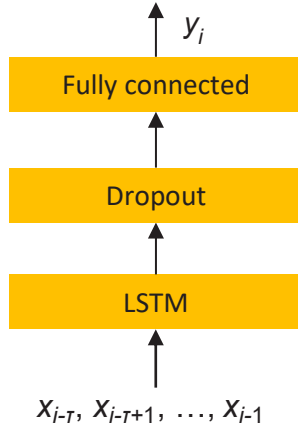- percentage of used satellites with low C/N0 (< 20 dB)
- speed over ground.

Fig. 5. Architecture of the applied neural network

Two models with the same architecture as depicted in Fig. 5 are developed for two kinds of outputs:

- Model 1: the position errors in north and east direction $[\delta p_N, \delta p_E]$ in local navigation coordinates are the outputs:

- Model 2: the horizontal position error δp_H, calculated by following equation, is the output:

$$\delta p_H = \sqrt{(\delta p_N)^2 + (\delta p_E)^2}. \tag{7}$$

Mathematically, the models can be formulated as:

$$y_i = f(x_{i-\tau}, x_{i-\tau+1}, \ldots, x_{i-1}) \tag{8}$$

where

- y_i is the position error(s)
- $x_{i-\tau}$ is the input(s) at $t_{i-\tau}$
- τ is the time lag in the processed sequence.

The target is to minimize the difference between the output estimated by the neural network $\hat{y}_i$ and the reference values of position errors y_i. The mean squared error (MSE) is used as the loss function. Mathematically, the MSE loss is expressed as

$$L = \frac{1}{N} \sum_N^n (\hat{y} - y)^2 \tag{9}$$

where N is the batch size.

4 Experiments and Results Analysis

For the training of the models, the dataset for the environment "urban area" is used. The sizes of training, validation, and test sets are respectively 36000, 4500, 4500. Here the size refers to the number of timestamps with multiple quantities measured per timestamp. The data are first normalized to improve the convergence speed and to avoid instability in the training. Furthermore, the dataset is sliced in a window form for the training of the LSTM network. This process is depicted in Fig. 6. As it can be seen, the window takes every time a subsequence in length of τ, which corresponds to the input array mentioned in eq. 8. After that, the window increments one step further down in the input sequence.
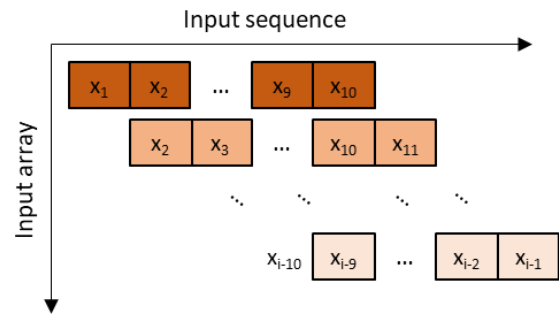

Fig. 6. Window method for dataset preparation with time lag equal to 10 measurement points

The models are implemented with the help of the machine learning library PyTorch [19] in Python. The Adam [20] and the rmsprop [21] optimizer have been tested. For the applied dataset, the rmsprop optimizer provides better results.

For model 1, it can be observed as in Fig. 7 that the estimated value of the model can generally follow the trend of the reference value. However, the model encounters difficulties in capturing the short-term features such as the peaks with extreme values. Moreover, the estimated values show a high frequency fluctuation. After further parameter tuning, it is noticed that a stacked LSTM layer or an LSTM layer with more hidden units cannot improve the drawbacks of the model. For the test set, the MSE loss for model 1 is 14.00 m^2.

Model 2, however, presents different results. It can be noticed in Fig. 8 that model 2 can capture the short-term peak at the beginning of the dataset. It can also trace the trend of the reference value. The MSE loss for the test set is 2.85 m^2. From the 2-D density plot in Fig. 9, it can also be noticed that most estimations lie near to their corresponding reference values. This result shows that model 2 could potentially be used to
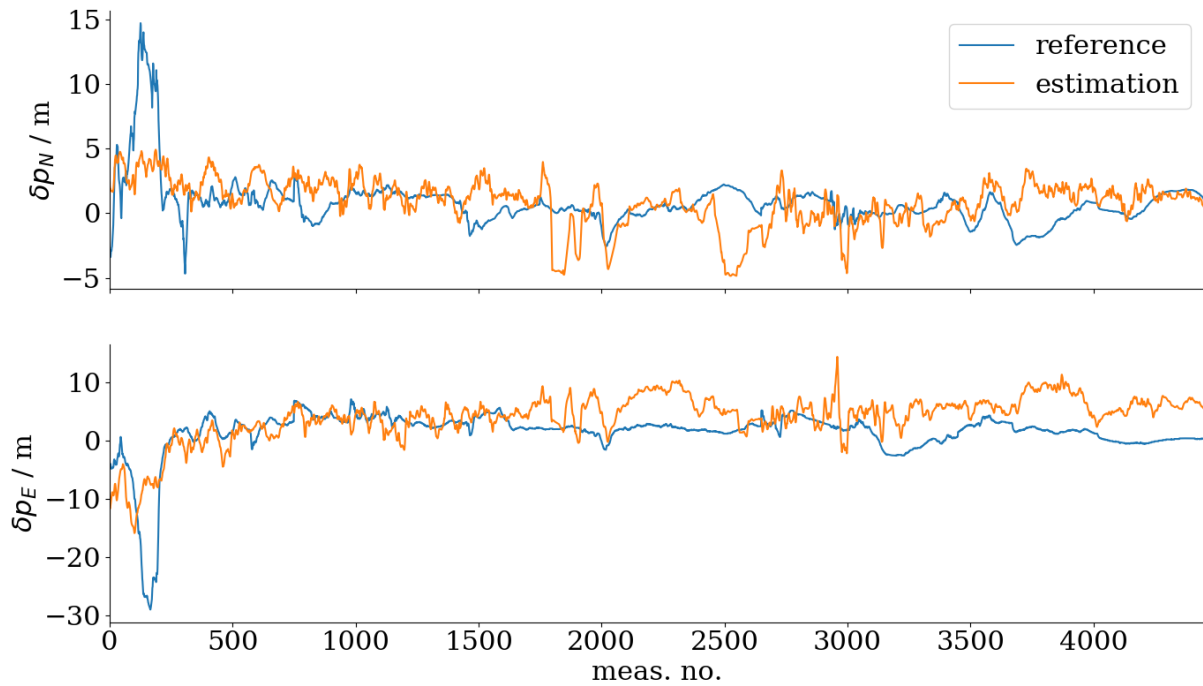
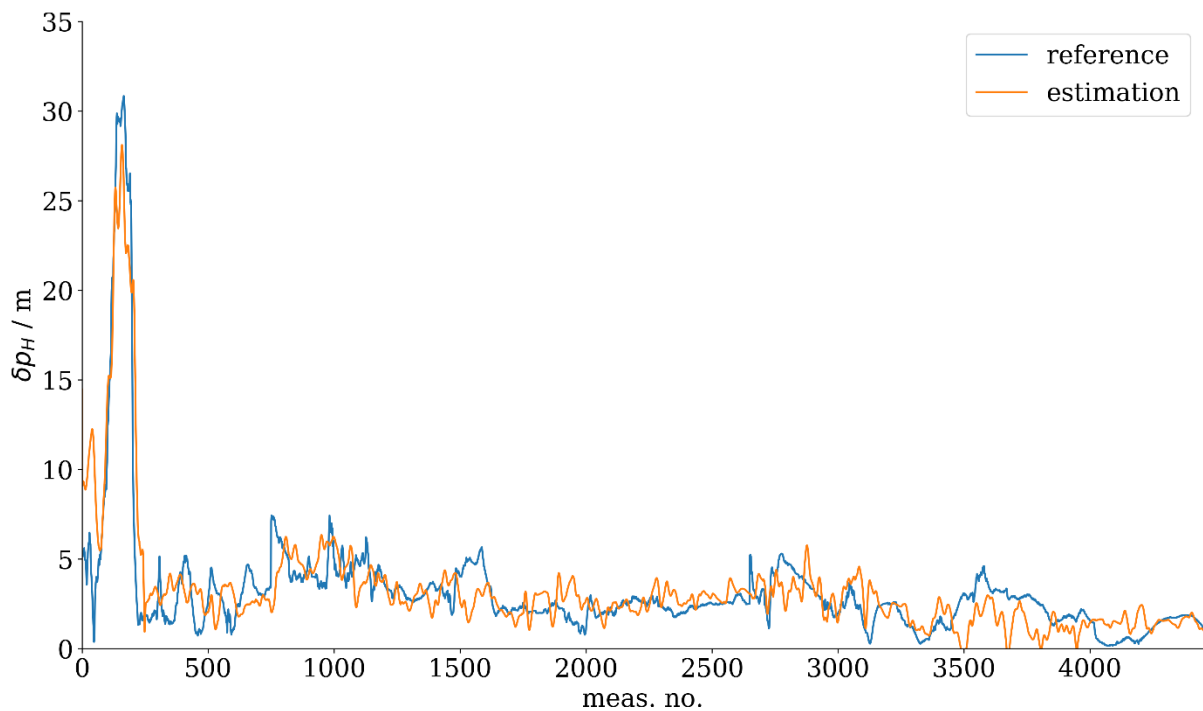*Fig. 7. Comparison of estimated and reference position error for model 1*

Fig. 8. Comparison of estimated and reference position error for model 2

calculate the horizontal protection level which is used for the autonomous integrity monitoring of GNSS-based localization systems.

Possible reasons for the existing problems are summarized as follows:

-   The used inputs are only indirect indicators for the error sources. More inputs are possibly required to capture the transient arising peaks;

-   Position errors contain both the deterministic part, which can be traced back to the inputs, and the stochastic part, which cannot be linked with the inputs. Thus, the model would not be able to learn the stochastic elements in the position errors;

-   Training and test demonstrate slightly different patterns. Despite the effort to improve regularization, the model is still

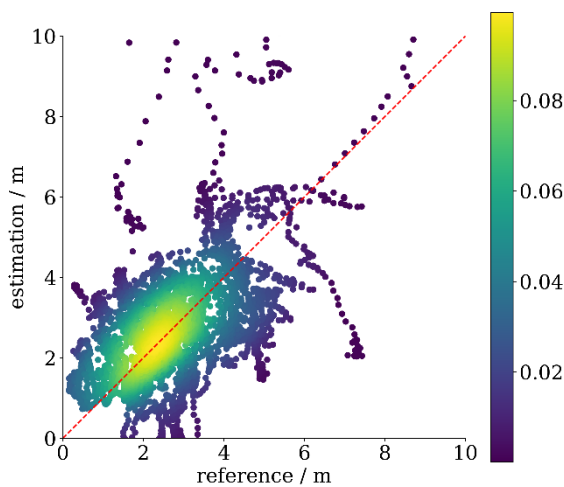overfitted to the training set. This resides in the nature of machine learning that it is data-dependent.

Fig. 9. Comparison of estimated and reference position error for model 2. The color scale represents the probability of occurrence.

Nonetheless, the models show the ability to predict the trend, especially the long-term bias is well learned.

5 Conclusion and Outlook

A machine-learning-based position error estimation for satellite-based localization systems is developed with the aim to explore an alternative to current common methods such as Bayesian estimators. The analysis shows that an explicit modeling from the error sources to the position domain is difficult to establish. Especially in the localization environment of land vehicles, local effects like NLOS complicate the modeling. In contrast, machine learning has the advantage that it can be trained to learn the effects of the measurement environment based on data.

Neural networks with an LSTM layer are applied for position error estimation based on the assumption that position errors are temporally correlated. Several common outputs of automotive-grade GNSS receivers are used as the inputs for the neural network to estimate position errors. Through simulation, it is shown that the trained networks have the ability to learn the long-term trend in the position error datasets. However, they do not satisfactorily accomplish to fit the short-term noise.

Possible improvements could be made by developing an alternative approach to model stochastics, i.e., a decomposition of position errors into deterministic and stochastic parts. Apart from that, modifications to the neural networks, such as a more complex architecture, are worth considering for research in the future as well.

References

[1] M. Obst, S. Bauer, and G. Wanielik, Urban multi-path detection and mitigation with dynamic 3D maps for reliable land vehicle localization, in *IEEE/ION Position, Location and Navigation Symposium (PLANS), 2012: April 23-26, 2012, Myrtle Beach Marriott Resort & Spa, Myrtle Beach, South Carolina, Myrtle Beach, SC, USA,* 685–691 (2012); doi: 10.1109/PLANS.2012.6236944

[2] P. D. Groves, Z. Jiang, M. Rudi, and P. Strode, A Portfolio Approach to NLOS and Multipath Mitigation in Dense Urban Areas, in *ION GNSS 2013* (2013)

[3] European Global Navigation Satellite Systems Agency, PPP-RTK market and technology report (2019). Accessed: May 10, 2020. [Online]. Available: https://www.gsa.europa.eu/sites/default/files/calls_for_proposals/rd.03_-_ppp-rtk_market_and_technology_report.pdf

[4] G. Rebala, A. Ravi, and S. Churiwala, *An introduction to machine learning*. Cham: Springer (2019); doi: 10.1007/978-3-030-15729-6

[5] Q. Zhang and B. Li, A low-cost GPS/INS integration based on UKF and BP neural network, *2014 Fifth International Conference on Intelligent Control and Information Processing (ICICIP): 18 - 20 Aug. 2014, Dalian, Liaoning, China, Dalian, China*, 100–107 (2014); doi: 10.1109/ICICIP.2014.7010322

[6] D. Wang, X. Xu, and Y. Zhu, A Novel Hybrid of a Fading Filter and an Extreme Learning Machine for GPS/INS during GPS Outages, *Sensors (Basel, Switzerland)*, vol. 18, no. 11 (2018); doi: 10.3390/s18113863

[7] M. Hashemi and H. A. Karimi, A Machine Learning Approach to Improve the Accuracy of GPS-Based Map-Matching Algorithms (Invited Paper), in *2016 IEEE 17th International Conference on Information Reuse and Integration: IRI 2016: 28-30 July 2016, Pittsburgh, Pennsylvania, United States: proceedings, Pittsburgh, PA, USA*, 77–86 (2016); doi: 10.1109/IRI.2016.18

[8] L.-T. Hsu, GNSS multipath detection using a machine learning approach, in *IEEE ITSC 2017: 20th International Conference on Intelligent Transportation Systems: Mielparque Yokohama in Yokohama, Kanagawa, Japan, October 16-19, 2017, Yokohama*, 1–6 (2017); doi: 10.1109/ITSC.2017.8317700

[9] A. Kuratomi, GNSS Position Error Estimated by Machine Learning Techniques with Environmental Information Input, KTH Royal Institute of Technology, Sweden, 2019. Accessed: Apr. 26 2020. [Online]. Available: http://kth.diva-portal.org/smash/get/diva2:1362035/FULLTEXT01.pdf

[10] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Boston: Artech House (2013)

[11] N. Viandier, D. F. Nahimana, J. Marais, and E. Duflos, GNSS Performance Enhancement in Urban Environment Based on Pseudo-Range Error Model, in *IEEE/ION Position, Location and Navigation Symposium, 2008: Monterey, CA, 5 - 8 May 2008, Monterey, CA*,377–382 (2008); doi: 10.1109/PLANS.2008.4570093

[12] C. Hegarty, E. D. Kaplan, M. Uijt de Haag, and R. Cosentino, GNSS Errors, in *The GNSS technology and applications series, Understanding GPS/GNSS: Principles and Applications / Elliott D. Kaplan, Christopher J. Hegarty*, editors, E. D. Kaplan and C. Hegarty, Eds., 3rd ed., Boston: Artech House, 619–659 (2017)

[13] M. Olynik, M. G. Petovello, M. E. Cannon, and G. Lachapelle, Temporal Variability of GPS Error Sources and Their Effect on Relative Positioning Accuracy, in *Proceedings of the 2002 National Technical Meeting of The Institute of Navigation*, 877–888 (2002)

[14] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, (2016); [Online]. Available: http://www.deeplearningbook.org/

[15] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural computation*, vol. 9, 1735–1780 (1997)

[16] I. Sutskever, O. Vinyals, and Q. V. Le, Sequence to Sequence Learning with Neural Networks, in *Advances in Neural Information Processing Systems* 27, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds.: Curran Associates, Inc., 3104–3112 (2014); [Online]. Available: http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf

[17] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, Deep Learning with Long Short-Term Memory for Time Series Prediction, *IEEE Commun. Mag.*, vol. 57, no. 6, 114–119 (2019); doi: 10.1109/MCOM.2019.1800155

[18] Google Earth 7.3.2.5776, Braunschweig and surrounding area, Germany: 52°14'44.70" N, 10°39'25.73", elevation 103 m: 2020 GeoBasis-DE/BKG; 2020 Google. Accessed: April 27 2020

[19] A. Paszke et al., Automatic differentiation in PyTorch (2017)

[20] D. P. Kingma and J. Ba, Adam: A Method for Stochastic Optimization (2014); [Online]. Available: http://arxiv.org/pdf/1412.6980v9

[21] G. Hinton, N. Srivastava, and K. Swersky, Neural networks for machine learning lecture 6a overview of mini-batch gradient descent