Tropospheric delays derived from ground meteorological parameters: comparison between machine learning and empirical model approaches

Luca Miotti[*], Endrit Shehaj[*], Alain Geiger[*], Stefano D'Aronco[*], Jan D. Wegner[*], Gregor Moeller[*], Markus Rothacher[*]

[*]Institute of Geodesy and Photogrammetry, ETH Zürich, Zürich, Switzerland
Email: miottil@student.ethz.ch; endrit.shehaj@geod.baug.ethz.ch

Abstract: High spatio-temporal variability of atmospheric water vapor is directly reflected in the tropospheric pathdelays that microwave satellite signals experience. The so-called zenith total delays (ZTDs) need to be estimated in case of Global Navigation Satellite Systems (GNSS). Usually, models describe the ZTD with three meteorological parameters measured on ground: pressure, temperature and partial water vapor pressure. However, these models are determined empirically and it is especially a struggle to accurately determine the delay caused by the water vapor (wet delay) from meteorological data. In this work, we provide an alternative approach of estimating the tropospheric path delay using machine learning (ML) algorithms. During the last two decades machine learning algorithms have become widely used in many fields of science and engineering. Therefore, a large amount of time series of ZTDs and meteorological data and the successful applicability of machine learning to various applications are the main motivation behind this work. Besides, we also investigated another approach to compute ZTDs, based on the well-known Saastamoinen model [Saastamoinen, 1973], after interpolating the meteorological parameters at GNSS sites. The idea behind this work is to genarate GNSS zenith pathdelays without processing any GNSS data, but only using meteorological parameters.
Therefore, GNSS zenith pathdelays from 72 permanent stations in Switzerland and meteorological data from the permanent SwissMetNet network (with over 120 stations) have been used for training and validation for a period of 11 years. The distribution of the sites all over Switzerland allows the network to be trained and validated with stations at different altitudes and with various meteorological conditions.
The ML approach showed an overall accuracy of 1.6 cm in terms of standard deviation, with almost no bias. Moreover, results show that stations at higher altitudes can benefit more from this approach. Compared to the Saastamoinen model, it had an overall improvement of about 20%, with a much better estimation in summer periods, when the amount of water vapor is higher. This work is a contribution to using ML algorithms to compensate for atmospheric errors in GNSS signals, and to compare its capabilities with empirically derived models.

1. Introduction

Signals from Global Navigation Satellite Systems (GNSS) experience delays and distortions when propagating through the atmosphere of the Earth. Due to the dispersive character of the ionosphere layer, its respective delay can be removed almost completely [1] from the GNSS observations using the ionosphere-free linear combination. However, the neutral atmosphere is non-dispersive and therefore the tropospheric delays have to be modelled (or estimated during GNSS processing) as accurately as possible. Several decades ago, researchers developed models, which describe the delay experienced by GNSS signals (such as [2], [3]) based on surface meteorological parameters. In more recent works [4], [5], [6], improved mapping functions and meteorological parameters have been used to model the tropospheric delay. [2] is probably still the most utilized model to correct for tropospheric errors in the GNSS community. These models were derived empirically by finding a connection between meteorological parameters and tropospheric delays using laboratory data or even numerical weather prediction models. It must be pointed

out that the most variable part of the delay is experienced by atmospheric water vapor, with a very high (and thus difficult to model) spatio-temporal distribution.

Meanwhile, machine learning algorithms have become widely popular in almost every field of science or engineering during the last two decades [7]. This was on the one hand caused by the refinement or development of new algorithms and on the other hand by ever more powerful and cheaper computational devices. They allow for extracting complex relations in large amounts of data and once the underlying models have been built, they are usually very time efficient. In general, it can be said that machine learning techniques can be used to increase accuracy with a low the computational time.

Nowadays, several decades of meteorological data, as well as few decades of GNSS-derived pathdelays from research centers are available in data servers ready to be exploited. In this context, the prediction of tropospheric delays by machine learning algorithms comes as a natural area of research. A few papers have already discussed this topic such as [8] and [9]. [8] uses a Long Short Term Memory (LSTM) neural network approach in combination with the SSA+Copula method to predict the future zenith wet delay based on past time series of temperature, water vapour pressure as well the wet delay itself. They achieve a mean absolute error of around 1 cm for a prediction over the next 24 hours. [9] uses a Gaussian Process (GP) regression model based on the zenith total delay to predict corrections for the tropospheric phase delay in InSAR time series. They claim an improvement of the correction of 81 % as well as a reduction of 50 % of the root mean squared error compared to the usual approach.

In this paper, we investigate a machine learning approach to accurately predict the zenith total delay based on meteorological parameters and compare it with an empirical model. We use a fully-connected neural network. The meteorological parameters are used to predict pathdelays for GNSS stations at location that differ from the meteorological sites. Initially, the delay from all available GNSS stations as well as the data from nearby meteorological stations is split up into two different datasets with respect to time: the network learns the relation between tropospheric parameters and zenith total delay from the first dataset and makes its predictions using the meteorological values for the second dataset. We compare this approach with an empirical model, where the meteorological parameters are initially interpolated at GNSS locations using the in-house-developed software COMEDIE (Collocation of Meteorological Data for Interpretation and Estimation of Tropospheric Path Delays) [10], [11] and then the Saastamoinen model is used to compute the zenith path delay. Therefore, both methods are compared with the actual zenith delays which have been estimated by the GNSS processing and which are used as reference for validation.

Especially for applications in mobile devices, machine learning approaches for tropospheric corrections of the GNSS signals could prove very impactful as pre-trained models can make accurate predictions without the need for large computational power.

In Section 2, the relation between tropospheric delays and meteorological parameters is shows, whilst Section 3 explains the used datasets. In Section 4, the methods used in this work are displayed and the results are shown in Section 5. Finally, Section 6 sums up the conclusions of this work.

2. Meteorological parameters and tropospheric delay

Meteorological quantities, such as air pressure, air temperature and water vapor partial pressure, can be used to determine the refractivity N at a specific location as follows [12]:

$$N = k_1 \frac{P_{tot} - P_{wet}}{T} + k_2 \frac{P_{wet}}{T} + k_3 \frac{P_{wet}}{T^2} \tag{1}$$

where P_{tot}, P_{wet} and T are the total pressure, the water vapor partial pressure and the temperature. In this work, we have used the coefficients from [13], where k_1, k_2 and k_3 are 77.6890 $\mathrm{KhPa^{-1}}$, 71.2952 $\mathrm{KhPa^{-1}}$ and 375463 $\mathrm{K^2hPa^{-1}}$, respectively.

The delay that microwave signals experience when travelling in the neutrospheric atmosphere, is the integrated value of the refractivity along the path between the satellite and the station:

$$\Delta\rho_{TROPO} = 10^{-6} \int_{Slant\ Path} N(s)ds \tag{2}$$

However, to correct for tropospheric delays in GNSS signals it would be extremely difficult to get the refractivity along the entire station-satellite path, as usually meteorological parameters are only measured at ground stations. Models based on surface meteorological parameters have been developed such as [2], [3]. The Saastamoinen model being the most typical one, represents the ZTD as:

$$ZTD = 0.002277(1 + 0.0026\cos(2\varphi) + 0.00028h_{ellip})[P_{tot} + (\frac{1255}{T} + 0.05)P_{wet}] \tag{3}$$

with φ geographical latitude, h_{ellip} ellipsoidal height. A GNSS signal experiences a tropospheric delay in the slant direction. However, the most basic tropospheric parameter, that GNSS community uses, is the ZTD. It is estimated as an unknown parameter in the GNSS adjustment, by using individual slant delays from each visible satellite combined into one parameter in the zenith direction.

3. Dataset

In this work, meteorological data from the SwissMetNet network [14] and GNSS data from the AGNES/COGEAR networks (operated by swisstopo and the Mathematical and Physical Geodesy group (MPG) of ETH Zurich) [15] over all Switzerland have been used. These networks are displayed in Fig. 1, where in blue the meteorological stations and in orange the GNSS sites are displayed. Here, we used data from 72 permanent GNSS stations and about 120 permanent meteorological stations. Both, meteorological and ZTD time series are available in an hourly interval. The time period available consists of eleven years and lies between the beginning of 2008 and the end of 2018.

It is important to point out that there were epochs (hours to weeks or months) with data gaps for some specific stations. Fig. 2 (top plot) displays exemplary the ZTDs time series, for the COGEAR GNSS station ERDE at an altitude of 731 m, together with meteorological parameters for the closest meteorological station Sion (SIO) (at an altitude of 484 m). The seasonal variations are visible in these time series, which indicate the relation of ZTDs with the meteorological data.

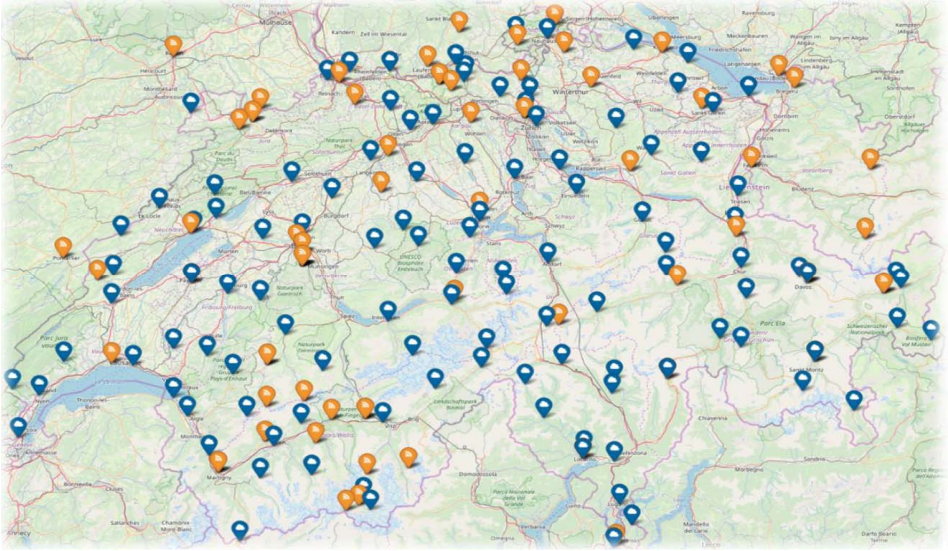

Figure 1 SwisMetNet (blue) meteorological network and AGNES/COGEAR GNSS (orange) network.

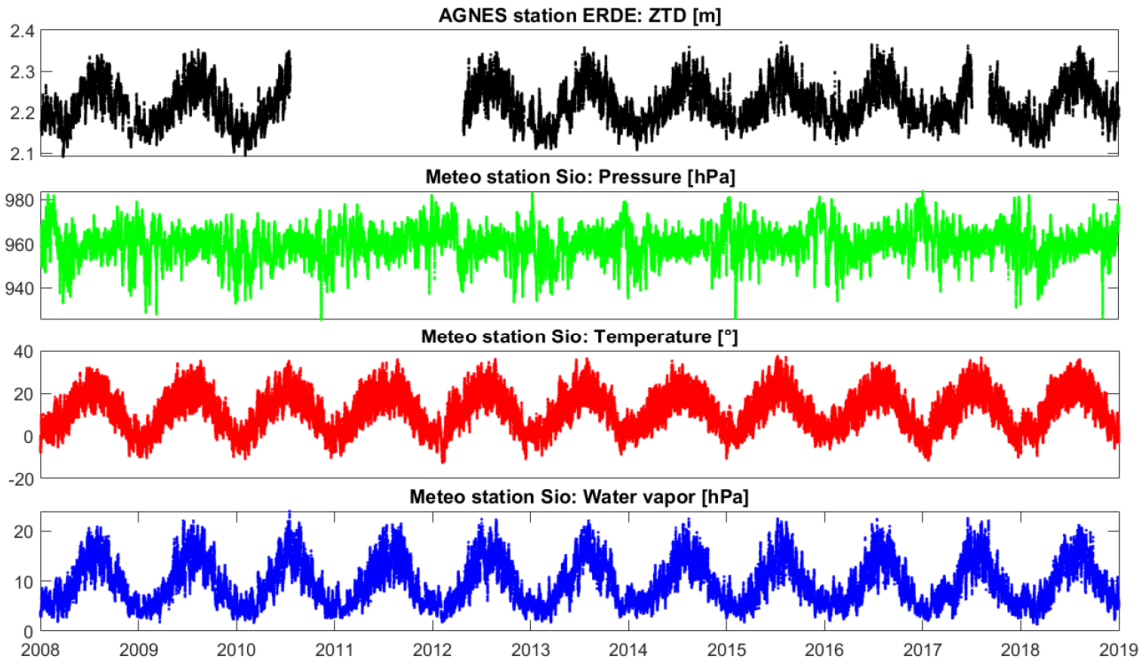

Figure 2 Time series of ZTDs for the station ERDE and metorological parameters for the nearby station SIO.

To generate the samples of the dataset, all GNSS stations were combined with their four nearest meteorological stations. On the one hand, the information used from GNSS stations is ZTDs and their absolute coordinates, while, on the other hand, the absolute coordinates as well as the respective meteorological parameters are the data used from the meteorological stations.

As shown in Fig. 1, the available meteorological stations are distributed quite evenly all over Switzerland. Furthermore, most GNSS stations have multiple meteorological stations in their close proximity. This can also be seen in Tab. 1. That table shows the average distance of each GNSS station to its respective four nearest meteorological stations.

	To 1st station	To 2nd station	To 3rd station	To 4th station
Mean distance [km]	8.2	13.2	17.7	21.5

Table 1 Mean distance of GNSS stations to their four closest meteo stations.

The providers (swisstopo and meteoswiss) removed outliers in the time series beforehand. According to [16], pressure, temperature and water vapor partial pressure have an accuracy of 0.15 hPa, 0.2 K and 0.5 hPa in terms of 1 sigma, respectively. Whilst GNSS ZTDs are nowadays estimated with cm accuracy [17].

Here, we used the years from 2008 to 2016 to train the network. 2017 and 2018 served as validation and test dataset respectively. This results in no overlap between training and test dataset. A final number of more than three million samples was generated, for which GNSS ZTDs and meteo parameters from the four closest stations were available simultaneously, as input to the training process.

4. Methodology

Machine learning approach

In the past decade, artificial neural networks (or multi-layer perceptron (MLP)) have become increasingly popular in the field of machine learning. Especially for large datasets, they are widely applied. The fully connected neural network applied here corresponds to the most original form of MLPs, where the neurons of one layer are connected to all neurons of the previous layer. Input and target values are simply given by the first and last layer of the network, respectively. A simple example of that principle is shown in Fig. 3. Table 2 summarizes the architecture in detail.

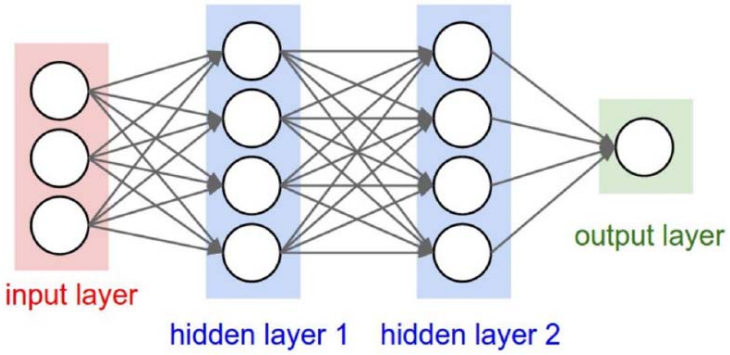

Figure 3 Fully connected neural network (source: http://cs231n.github.io/).

Layer	Input layer	Hidden layer 1	Hidden layer 2	Hidden layer 3	Hidden layer 4	Hidden layer 5	Output layer	Total
Number of neurons	27	512	128	128	128	128	1	-
Parameters	-	14'336	65'664	16'512	16'512	16'512	129	129'665

Table 2 Architecture of the used neural network.

The value of an individual neuron can be calculated in the following way [18]:

$$neuron = f(\sum_{i=1}^{n} input_i * weigth_i + bias) \tag{4}$$

where $input_i$ is the value of one neuron in the previous layer with n the amount of neurons in the previous layer. As it can be seen, each connection of the network gets a certain $weigth_i$. An additional $bias$ for this particular neuron is then added. These weights and biases make up all the parameters of a fully connected neural network. The nonlinearity of the network is introduced by the Rectified Linear Unit (ReLU) activation function f [18].

$$f(x) = \max(0, x) \tag{5}$$

The training of the network is performed by searching for a local minimum in the loss function. In this case, this loss function consists of the mean squared error (mse) between the prediction of the network and the actual label. The unknown parameters of the loss function are exactly described by the weights and biases of the network.

$$mse = \frac{1}{n}\sum_{i=1}^{n}(pred_i - label_i)^2 \tag{6}$$

More information regarding the loss function can be found in [19].

The hyperparameters (learning rate, batch size and epochs) selected for the training were 0.0001, 1000 and 250, respectively. These values were determined and optimized empirically.

Saastmoinen model after collocation of meteorological data

For comparison purposes, Saastmoinen model values were computed for the corresponding stations and timeframe, following three steps:

1. A collocation of the meteorological data within a radius of about 60 km in east and north direction is performed. In collocation the measurement l is expressed as the sum of a functional part f (depending

on the coordinates where the measurement was computed), a signal term s (which is a stochastic term accounting for correlated noise in the residuals of the observations) and a noise term ϵ that remains from the residuals after subtracting the signal [20]:

$$l = f + s + \epsilon \tag{7}$$

The functional models for pressure, temperature and water vapor pressure are [20]:

$$P(x, y, h, t) = (P_0 + a(x - x_0) + b(y - y_0) + c(z - z_0)) * e^{\frac{h-h_0}{H}} \tag{8}$$

$$T(x, y, h, t) = T_0 + a(x - x_0) + b(y - y_0) + c(z - z_0) + H * (h - h_0) \tag{9}$$

$$E(x, y, h, t) = (E_0 + a(x - x_0) + b(y - y_0) + c(z - z_0)) * e^{\frac{h-h_0}{H}} \tag{10}$$

where a, b and c correspond to the coefficients of horizontal and temporal gradients, H is the scale height and P_0, T_0, E_0 are the pressure , temperature and water vapor pressure at the reference point (x_0, y_0, h_0 and t_0). x, y, h, t are the 4D coordinates of the measurement.

The stochastic term s is based on a covariance function, which describes the weighting of the individual sampling points to each other. This covariance function is determined empirically and it contains information about the correlations between the measurements. More information is available in [10], [11] and [20].

2. The collocated meteorological data is then interpolated at the location of the GNSS stations, where collocation parameters estimated in step 1 and the empirically defined covariance matrix are utilized.
3. Finally, the computation of the delay is performed with the empirical Saastamoinen formula (3).

Therefore, the same data as provided to the machine learning algorithms are also used to compute pathdelays with the Saastamoinen model. Thus, a fair comparison in terms of accuracy can be made. It is important to point out that with this method the computation of ZTDs took several hours (quite a lot compared to machine learning approach). Although this could still be optimized, it also indicates a potential advantage of the proposed machine learning approaches; especially because already trained models are able to make predictions even for very large datasets in a much shorter amount of time.

5. Results

In this section, the results obtained by the two methods described in Section 4 are presented and compared. It must be pointed out that the minimal targeted accuracy for the developed neural network was two cm, after preliminary tests (not shown here) using basic machine learning algorithms (such as random forest).

Machine learning prediction of ZTDs using meteo data

Figure 4 shows the residuals of the measurements from the neural network prediction. The results show values below 2 cm in terms of root mean square error or mean absolute error. Additionally, it is interesting to note that the residual distribution is centered at almost exactly zero, with a mean value of less than a tenth of a millimeter. Because of the individual circumstances of each GNSS station, it is expected that the achieved accuracy will vary quite a lot depending on the station. The difference in station height as well as the distribution of available meteorological stations are after all quite vast. It is important to note, however, that no station has an rmse higher than the minimum goal of 2 cm. In addition, the average residual for no station is higher than 0.69 % of the respective delays. Fig. 5 displays the respective rmse as well as the yearly mean of water vapor partial pressure with respect to the height of the stations. This figure indicates

that the predictions for stations above a certain altitude generally result in a much lower error. The correlation of -0.75 between the rmse and the stations height is quite high, whilst the correlation of water vapor partial pressure and height is almost -1. The error is considerably lower for very high stations. This result is expected as the higher the stations the smaller the wet delay (which is directly connected to water vapor partial pressure) and its variability get; and therefore the error of the prediction is lower. However, below a station height of 1000 m the altitude does not seem to have as much of an impact anymore.

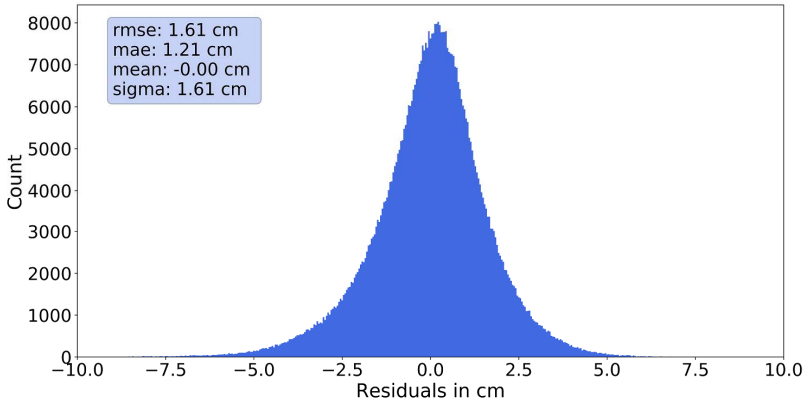

Figure 4 Residuals for the neural networks prediction.

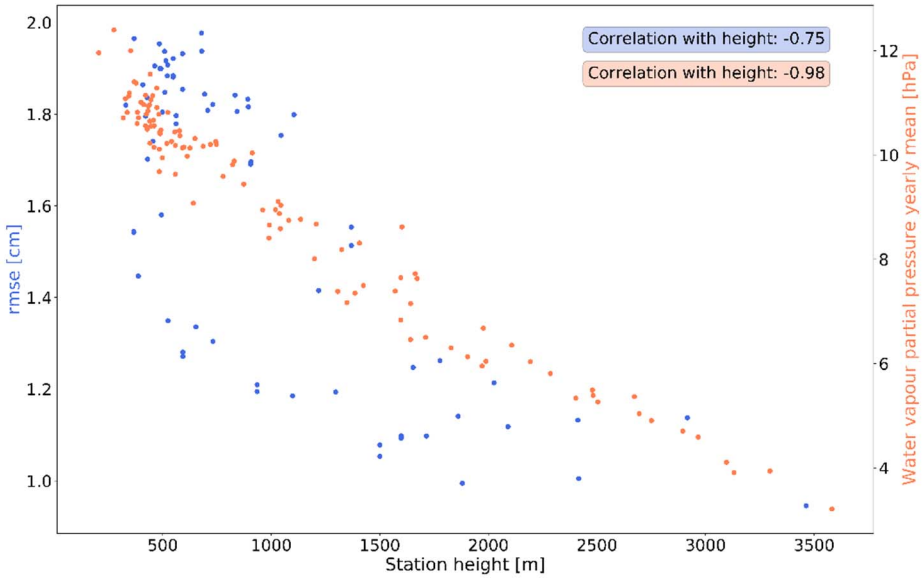

Figure 5 Rmse of residuals for the neural network prediction (blue) and yearly mean of water vapor partial pressure (orange), with respect to height.

Comparison with empirical models

To further assess the accuracy achieved by the neural network predictions, the ZTDs are now compared to the Saastamoinen model as explained in Section 4. Tab. 3 shows several properties of the residuals of the two methods.

	Neural network	Saastamoinen
Root mean square error (rmse) [cm]	1.61	2.09
Mean absolute error (mae) [cm]	1.21	1.59
Mean relative error (mre) [%]	0.55	0.74
Residuals above 2 cm [%]	18.57	30.06
Mean of residuals [cm]	0.00	0.34

Table 3 Comparison between neural network and Saastamoinen predictions.

According to Tab. 3, the neural network predictions are generally better than the ones of the Saastamoinen approach. For rmse and mae the resulting error is roughly 4 mm lower, which corresponds to a reduction of the error of about 19 %. The improvement in the mre is (with a relative reduction of about 27 %) ca. 0.2 % in an absolute sense; for instance if the ZTD is 2 m, the Saastamoinen computed ZTD is estimated about 4 mm worse than the neural network prediction. Additionally, the amount of residuals that lie above the threshold of 2 cm is considerably lower for the machine learning approach. In relative terms, this corresponds to a value that is almost 40 % lower than that of Saastamoinen model. Moreover, there is a bias of few mm in the estimation of the delay produced by the Saastamoinen approach, whilst the mean of the residuals for the neural network is almost zero. This indicates that the Saastmoinen approach usually tends to overpredict the delay, while the machine learning algorithm has no bias for the complete set of stations.

Analyzing the result with respect to different times of the year, Fig. 6 shows a monthly comparison of the rmse for both approaches. Generally, regardless of the approach, the error is much lower during the winter and spring months, while the highest errors occur during August, September and October. This is likely again caused by the increased wet delay during this time period, which is reflected in a higher uncertainty of the prediction. Moreover, from this figure we see that the neural network approach deals better with this problem than the Saastamoinen model, as the difference in rmse between them is larger during the summer and autumn months than for the winter period.

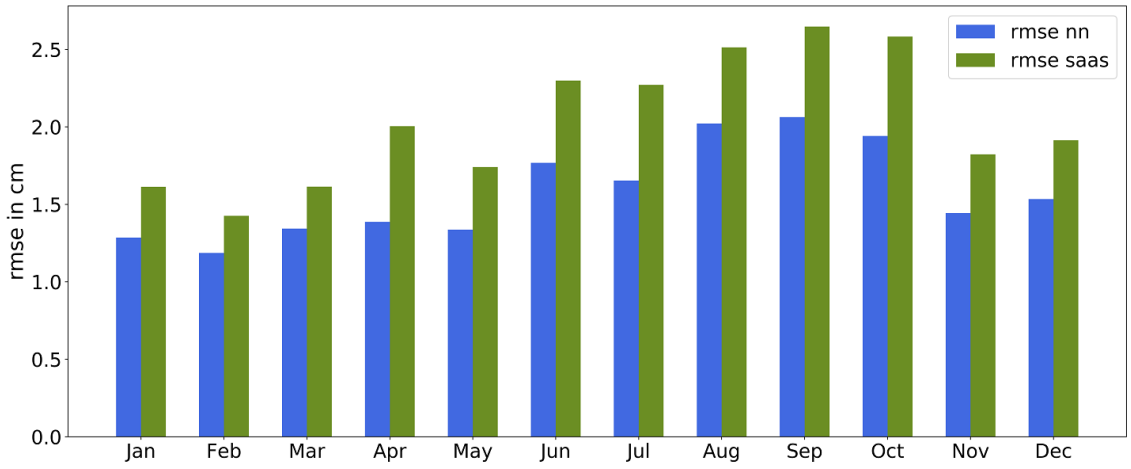

Figure 6 Root mean square errors for the neural network and the Saastamoinen predictions for each month.

Furthermore, to have a closer look at individual predictions, the WEHO station delays were analyzed. This station at 2’916 m elevation is the second highest station of the network. It is located in the Canton of Bern right next to the border to the Valais. Fig. 7 shows the monthly comparison of the rmse for this station.

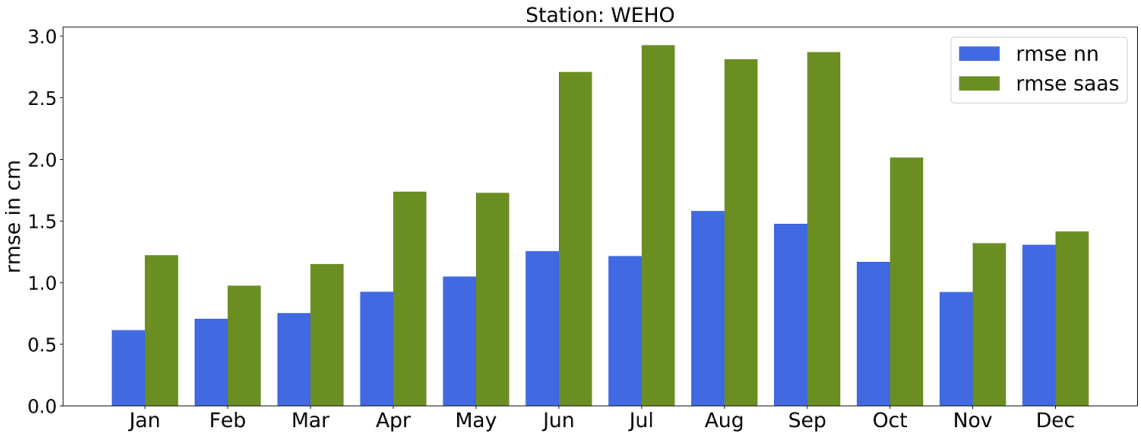

Figure 7 Root mean square errors for the neural network and the Saastamoinen predictions for each month, only for GNSS station WEHO.

As can be seen, the monthly distribution of the rmse shows very similar properties as the error for all stations. Again, the error is generally much lower during winter. Especially, the vast difference during the summer months is noticeable, where in some cases the rmse of the neural network predictions is less than half of that of its Saastamoinen counterpart. This is also visible when looking at the time series of predictions and their corresponding residuals, displayed in Fig. 8. As already mentioned, the Saastamoinen model tends to overpredict the delay. This is especially visible here during the summer months as can be seen in the residuals. The neural network on the other hand seems to fail to capture the extreme spikes of the delay and tends to smooth its predictions.

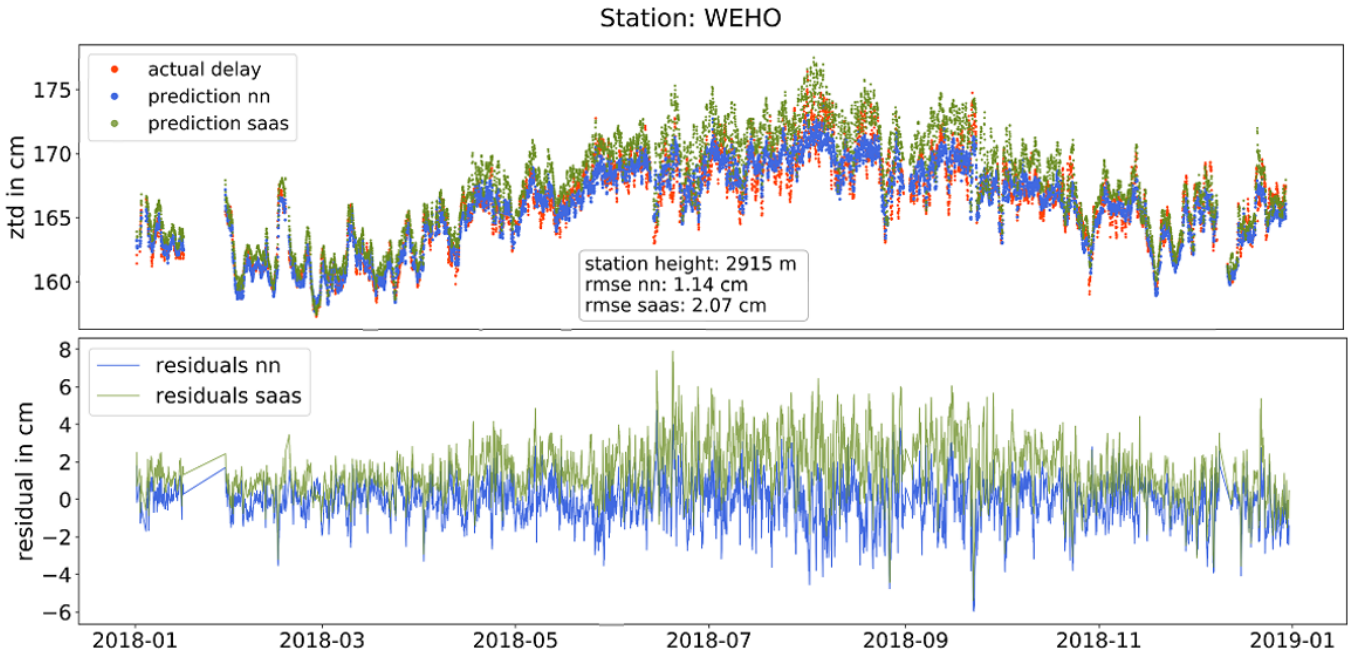

Figure 8 Actual estimated delay and the respective neural network and Saastamoinen predictions (top plot), residuals of predictions (bottom plot); only for GNSS station WEHO.

6. Conclusions

In this work, we presented the estimation of tropospheric (zenith) pathdelays for GNSS using meteorological parameters from a network all over Switzerland. On the one hand, a machine learning approach based on artificial neural networks was developed to train the prediction of GNSS pathdelays from the meteorological data. On the other hand, meteorological parameters were interpolated at GNSS station locations using a collocation approach and the Saastamoinen model to compute tropospheric pathdelays. From the machine learning approach, the root mean square error was never bigger than 2 cm; it averaged at 1.6 cm and had almost a zero bias. A dependency of the rmse was found with respect to height, where the higher the GNSS station the lower the overall error. This confirmed the fact that at higher altitudes, the amount of water vapor is smaller, resulting in better predictability of the delays. From comparisons with the computed delays using the Saastamoinen formula, an overall improvement of about 20% was achieved with the neural network. The improvement was more obvious in summer periods, where the amount of wet delay is also larger. Furthermore, the predicted delays by Saastamoinen experienced a few mm bias from the reference delays, whilst the machine learning predictions are almost bias-free. Overall, it was shown in this work that machine learning can be used to model tropospheric delays by using meteorological data. The increasing amount of time series of ZTDs and meteorological data is a good indication that empirical models can be overcome; however, this clearly needs further testing and datasets to be compared. Especially for applications in mobile devices, this approach may have a large impact, as pre-trained models can make accurate predictions without the need for large computational power.

ACKNOWLEDGMENTS

The authors would like to thank swisstopo for providing the GNSS AGNES/COGEAR measurements and MeteoSwiss for the SwissMetNet data.

References

[1] Kaplan ED und Hegarty CJ, Understanding GPS principles and applications, Artech House, 2016.

[2] Saastamoinen J, "Contributions to the theory of atmospheric refraction: Part II. Refraction corrections in satellite geodesy.," *J. Geod.,* pp. 107, 13-34., 1973.

[3] H. S. Hopfield, "Tropospheric effect on electromagnetically measured range: Prediction from surface weather data," *Radi Science,* pp. Vol. 6(3), 357-367, 1971.

[4] J. Collins, Assessment and Development of a Tropospheric Delay Model for Aircraft Users of the Global Positioning System, Department of Geodesy and Geomatics Engineering, Technical Report No. 203, University of New Brunswick, Fredericton, New Brunswick, Canada, 1999.

[5] R. F. Leandro, M. C. Santos and L. R. B., "A North America Wide Area Neutral Atmosphere Model for GNSS Applications," *NAVIGATION, Journal of The Institute of Navigation,* pp. Vol 56(1), 57 - 71, 2009.

[6] G. Möller, R. Weber and J. Böhm, "Improved Troposphere Blind Models Based on Numerical Weather Data," *NAVIGATION, Journal of The Institute of Navigation,* pp. Vol. 61(3), 203-211, 2014.

[7] A. E. Hassanien, Machine Learning Paradigms: Theory and Application, Springer, 2018.

[8] Kitpracha C, Modiri S, Asgarimchr M, Heinkelmann R und Schuh H, *Machine Learning based prediction of atmospheric zenith wet delay: A study using GNSS measurements in Wettzell and co-located VLBI observations,* Vol. 21, EGU2019-4127, 2019.

[9] Shamshiri R, Motagh M und Nahavandchi H, *A machine learning-based regression technique for prediction of tropospheric phase delay on large-scale Sentinel-1 InSAR time-series,* EGU, 2019.

[10] Eckert V, Cocard M and Geiger A, "COMEDIE:(Collocation of Meteorological Data for Interpretation and Estimation of Tropospheric Pathdelays) Teil I: Konzepte, Teil II: Resultate; Technical Report 194," ETH Zürich. Grauer Bericht., 1992a.

[11] Eckert V, Cocard M and Geiger A, "COMEDIE :(Collocation of Meteorological Data for Interpretation and Estimation of Tropospheric Pathdelays) Teil III: Software; Technical Report 195," ETH Zürich. Grauer Bericht., 1992b.

[12] L. Essen and K. Froome, "The Refractive Indices and Dielectric Constants of Air and its Principal Constituents at 24,000 Mc/s," in *Proceedings of the Physical Society*, 1951.

[13] J. Rueger, "Refractive Index Formulae for Radio Waves," in *Proceedings of the FIG XXII International Congress*, Washington, D.C. USA, 2002.

[14] Meteoswiss, «Automatic monitoring network: SwissMetNet project,» 2020. [Online]. Available: https://www.meteoswiss.admin.ch/home/measurement-and-forecasting-systems/land-based-stations/automatisches-messnetz.html. [Zugriff am 30 January 2020].

[15] Swisstopo, 2019. [Online]. Available: http://pnac.swisstopo.admin.ch/pages/en/agnes-status.html#, last visited: June 2019.

[16] Hurter F, GNSS meteorology in spatially dense networks, Geodätisch-geophysikalische Arbeiten in der Schweiz, Swiss Geodetic Commission., 2014.

[17] Teunissen PJG and Montenbruck O, Springer Handbook of Global Navigation Satellite Systems, Springer, 2017.

[18] Nwankpa C, Ijomah W, Gachagan A und Marshall S, «Activation Functions: Comparison of Trends in Practice and Research for Deep Learning,» *arXiv:1811.03378 [cs.LG],* 2018.

[19] Haykin SO, Neural Networks and Learning Machines, Pearson, 2009.

[20] Troller M, GPS based Determination of the Integrated and Spatially Distributed Water Vapor in the Troposphere, Geodätisch-geophysikalische Arbeiten in der Schweiz, Swiss Geodetic Commission., 2004.