Research Article

Patricia Daneghian and Asghar Rastbood*

GNSS-IR soil moisture estimation using deep learning with Bayesian optimization for hyperparameter tuning

https://doi.org/10.1515/jogs-2022-0172
received June 10, 2023; accepted February 29, 2024

Abstract: One of the ways for measuring water content is using global navigation satellite system (GNSS) multipath signals. By analyzing those signals, we will get useful information about the reflection surface. This technique is called GNSS interferometric reflectometry. Some receivers can record signal-to-noise ratio (SNR) data, which includes the SNR component of reflected signals and is strongly related to soil moisture. Here, we will use the data for station P038 in Mexico for 4 years, from 2017 until 2020. The calculation steps include extracting SNR data from RINEX files, estimating the prior reflector height and phase, calculating SNR metrics, and removing the vegetation effect to obtain volumetric water content (VWC). The results show that the VWC level has increased from 8.88 to 12.49% from 2017 to 2020. We have used long short-term memory neural networks with tuned hyperparameters by Bayesian optimization for predicting the time series of soil moisture. Our model is trained using 80% of the station observations. The accuracy of the network is checked using different metrics on the train, test, and all data. The mean absolute error, root mean square error, and a20-index of the test data are obtained as 0.002, 0.041, and 0.727, respectively. The modeling results will help farmers arrange their irrigation schedules more professionally.

Keywords: Bayesian optimization, deep learning, long short-term memory, multipath, reflection, soil moisture

1 Introduction

In the modern era, per capita cultivable land is declining due to the rapid increase in population. Additionally, the shortage and cost of fresh water have become serious challenges. For this reason, agricultural professionals should use innovative technologies and solutions to improve productivity, save on initial costs, and optimize the consumption of environmental resources, especially water. Therefore, measuring, monitoring, and estimating soil moisture are very important.

One of the main parameters for expressing the amount of water in the soil and the state of irrigation is the volumetric water content (VWC). The VWC is the ratio of the volume of water to the unit of soil volume. It can be expressed as a ratio, a percentage, or a depth of water at any soil depth (assuming a unit of surface). VWC indicates how much of the soil volume is filled with water. In this study, VWC is expressed as a percentage, with values ranging from zero to 100%. A value of zero indicates that the soil is completely dry and devoid of water, while a value of 100% indicates that all voids and empty soil spaces are filled with water.

Soil moisture measured in large areas shows how much rainfall has returned to the atmosphere. Therefore, soil moisture is an essential component of the hydrological cycle and a major observational quantity for optimizing irrigation management in agriculture. In addition to the main applications such as positioning, navigation, and timing, global navigation satellite systems (GNSSs) have other applications such as atmospheric remote sensing, reverse problem-solving related to earthquakes to determine the degree of fault dislocation, and crustal deformation monitoring (Chen et al., 2018; Li et al., 2015). The new application of GNSS is reflection measurement interference with GNSS, which is called GNSS interferometric reflectometry (GNSS-IR). It is an approach to approximating environmental and geophysical parameters around a GNSS station with the quality of other geodetic products.

* **Corresponding author: Asghar Rastbood,** Department of Surveying Engineering, Faculty of Civil Engineering, University of Tabriz, Tabriz, Iran, e-mail: arastbood@tabrizu.ac.ir
Patricia Daneghian: Department of Surveying Engineering, Faculty of Civil Engineering, University of Tabriz, Tabriz, Iran

Reflected signals are used for this purpose. Signal reflection is known as multipath. When the goal is to determine the position, multipath is considered one of the most important sources of error. This is because the reflected signals are bounced off the ground surface before being received by the antenna, resulting in a recorded distance between the satellite and the antenna that is greater than the actual value. This systematic error affects the accuracy of positioning.

In recent years, GNSS-IR has been proposed and tested as a possible solution to obtain continuous information on soil moisture at a local scale. The feasibility of this solution is due to the fact that GNSS satellites send L-band signals (microwave frequency), which are suitable for near-surface use. Signals that are reflected by adjacent surfaces and recorded by the antenna contain information about the antenna's surroundings on a scale of about 1,000 m^2. For these frequencies, the coefficient of signal reflection and soil permeability depend on soil moisture (Chew et al., 2013; Larson et al., 2009; Larson et al., 2008). Therefore, a signal reflected from the ground surface measured by a receiver can be used to understand the temporal changes in soil moisture near the ground surface. In this method, changes in the interference pattern of direct and reflected signals are analyzed. The main data recorded by GNSS receivers are code and carrier phase observations.

The third observation recorded by some receivers is the signal-to-noise ratio (SNR). SNR represents the ratio of signal power to noise measurement and is expressed using logarithmic decibel (dB) or decibel-Hertz (dB-Hz) units. The changes in the interference pattern of the direct and reflected signals, recorded in the SNR data, are plotted as interference diagrams. The temporal fluctuations of the interference diagrams indicate changes in the volume of soil moisture near the surface of the ground at a depth of approximately 5 cm.

Figure 1(a) represents the extracted SNR curve obtained from the reflection of bare soil. In Figure 1(b), the presence of snow covering the soil causes significant changes in the SNR curve compared to the bare soil. Figure 1(c) shows a layer of vegetation on the soil, which results in noticeable changes to the SNR curve. Figure 1(d) illustrates the SNR curve created by the reflection of wet soil. Soil moisture induces slight changes in the SNR curve, making it

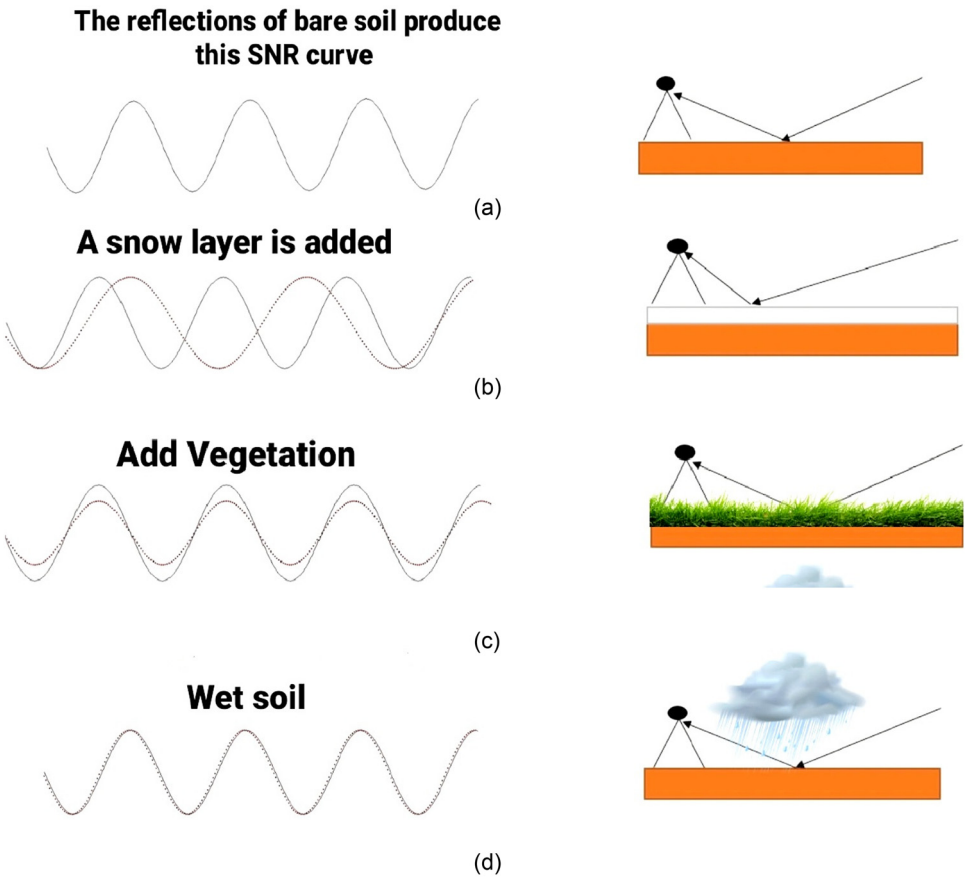

Figure 1: Changes occurred in the SNR curve in (a) bare soil, (b) snowy soil, (c) vegetated soil, and (d) wet soil.

challenging to obtain accurate information about soil moisture levels solely from SNR data analysis. However, recent studies have demonstrated that GNSS-IR has a strong capability to measure soil moisture content (SMC). For instance, SNR can be utilized to observe small changes in surface reflection and SMC (Larson et al., 2009; Mao et al., 2009). GNSS receivers receive both direct and reflected signals. GNSS-IR works based on the fact that SNR data values change as soil moisture changes (Dobson et al., 1985). Building on the relationship between soil reflection coefficient and soil moisture, Luo et al. (2016) developed an experimental model. Chew et al. (2014) proposed an estimation model to acquire soil moisture for a depth of 5 cm from the top of the uncoated soil.

Several studies have been conducted on soil volumetric moisture estimation using interferometry to measure the reflection of GNSSs. One such case study was published by the European Union for Geosciences and focused on GNSS interference reflection at the Cajamar testing center in Valencia, Spain. The study aimed to monitor soil moisture for 66 days, from December 3, 2018, to February 6, 2019. In this study, GNSS observations were conducted by integrating the satellite systems, global positioning system (GPS), GLONASS, and GALILEO. Simultaneously, local soil sampling at a depth of 5 cm was performed to determine soil moisture and create a validation dataset. The comparison of satellite reflection measurement interference data with field datasets revealed that the GNSS reflection interference measurement method has high potential for estimating soil moisture changes and assessing plant growth near the station (Martín et al., 2020).

Another study focused on changes in soil moisture volume and wheat growth on a farm in Beijing, China. The study utilized SNR data from the Beidou satellite system, in addition to GPS, and evaluated the results by comparing them with field observations. The study concluded that GNSS reflection interferometry technology has the desired ability and potential to monitor soil volumetric moisture and plant growth in agricultural fields (Zhang et al., 2021).

Accurately estimating future soil moisture holds practical significance for eco-hydrologists, agriculturists, and stakeholders involved in environmental health monitoring. It enables the development of a comprehensive understanding of hydro-physical and soil dynamic systems.

Artificial neural networks (ANNs) possess the capability to capture complex non-linear relationships among input variables and exhibit strong resilience toward data noise, as highlighted by the research conducted by Twarakavi et al. (2006). ANNs have found extensive application across diverse fields and disciplines. Their effectiveness and contributions are widely acknowledged in the field of water resources, hydrology, and structures (Ahmad and Simonovic, 2005; Armaghani et al., 2019; Asteris et al., 2021; Asteris and Nguyen, 2022). Over the past two decades, ANNs have emerged as a crucial methodology within the scientific community for comprehending intricate data patterns. Chai et al. (2009) employed ANNs to retrieve soil moisture by utilizing microwave data collected from a polarimetric L-band multi-beam radiometer. The findings of their study demonstrated significant enhancements in soil moisture retrieval through the utilization of the ANN model. Several authors have reported a multitude of studies focused on soil moisture retrieval utilizing microwave data (Alexakis et al., 2017; Gupta et al., 2017; Kumar et al., 2019; Paloscia et al., 2013). The application of the ANN model has also extended to the retrieval of soil moisture from optical data, including parameters such as land surface temperature and normalized difference vegetation index (Ahmad et al., 2010; Kumar et al., 2019). In a study conducted by Hassan-Esfahani et al. (2015), an evaluation was conducted on the performance of an ANN model in estimating soil surface moisture using spectral images.

Dabboor et al. (2023) introduced a deep learning framework for SMC prediction from satellite data. The framework incorporated various optimized machine learning models, including ANN, Gaussian process regression, support vector machines, decision trees, and elastic net regression. To enhance the prediction performance of the machine learning models, the Bayesian optimization strategy was utilized by the researchers for fine-tuning the hyperparameters. Augmenting the data resulted in improved performance of the five machine learning prediction models, as indicated by the outcomes of their study.

In a study conducted by Ahmed et al. (2021), the effectiveness of a hybridized long short-term memory (LSTM) predictive framework in emulating soil moisture patterns under global warming scenarios was demonstrated. The ability to simulate future soil moisture under the influence of climate change is demonstrated in their study through the utilization of the LSTM algorithm in conjunction with Boruta-random forest feature selection. Consequently, their approach holds promise for successful implementation across a range of domains, including hydrology, agriculture, soil use management, and environmental management.

Given that deep learning is often regarded as a black box approach, researchers have limited opportunities to manually fine-tune parameters due to the hidden layers and numerous hyperparameters associated with network structure and training algorithms. When it comes to optimizing the hyperparameters of a deep neural network, Bayesian optimization proves to be highly valuable, especially in scenarios where assessing the model's accuracy might require a long time. The objective of hyperparameter

optimization is to identify an algorithm that yields the best and most accurate performance based on results obtained from a validation set. The optimizer determines the optimal hyperparameters that result in the highest score on the test set (Victoria and Maragatham, 2021). A variety of design problems in different fields have benefited from the widespread application of Bayesian optimization. These fields include gait recognition (Feurer et al., 2019), environmental sustainability and monitoring (Marchant and Ramos, 2012), algorithm configuration through optimization (Brochu et al., 2010), automated machine learning (Bergstra et al., 2011; Hoffman et al., 2014; Snoek et al., 2012), reinforcement learning (Brochu et al., 2010), and big data applications (Shahriari et al., 2015).

The algorithm used in this study is implemented to estimate soil moisture across the entire EarthScope PBO (Plate Boundary Observation) H_2O network stations, mostly located in the western United States and with the largest variation in vegetation cover (Chew et al., 2013). The PBO H_2O project is defined to study the water cycle and uses GNSS reflective data related to the PBO network. All PBO H_2O network stations have choke-ring antennas, and almost all use Trimble NetRS receivers. Stations are located in areas with low topographic variations and in ecosystems known as grass fields or shrubs. Some stations record data in time intervals of 1 s and some in time intervals of 15 s.

In this article, the p038 station was selected from the network to conduct research. The reason for choosing this station is the presence of undefined and unchanged reflective observations of the station specifications for a period of 4 years during the years 2017–2020. SNR data were extracted from RINEX files, and the algorithm provided by Larson et al. (2009) was used to estimate soil moisture changes during 2017–2020. An ANN of LSTM with hyperparameters tuned using Bayesian optimization was also used to model and predict the time series of soil moisture, and the data were evaluated based on its results. The results of the estimates show that soil moisture in this area is on an upward trend.

2 Soil moisture estimation algorithm using GNSS

This algorithm is used to generate soil moisture estimations for all stations in the PBO network. The algorithm consists of four main steps. The first step involves selecting useful satellite tracks. In the second step, a prior reflection height is estimated for each track. The third step approximates the SNR metrics. Finally, in the fourth step, vegetation effects are quantified and removed.

2.1 Selection of useful satellite tracks

Not all satellite tracks are useful for our research. There are certain conditions that help us choose the appropriate cases. Satellite tracks should exhibit steady reflections within satellite elevation angles of 5°–25° or 5°–30°. When selecting the analysis area, we need to consider the presence of buildings and man-made surfaces, such as roads, as they can interfere with the tracks. Generally, sites with a higher number of satellite tracks used tend to produce more reliable soil moisture estimates.

2.2 A prior reflection height estimation for each track

Figure 2 depicts a simplified illustration of a reflected signal. This figure presents information regarding the distance between the reflecting surface and the receiver, referred to as the reflector height (RH). The initial height of the reflector (H_0) is not entirely known and needs to be estimated based on the SNR data.

The SNR of the direct signal, as displayed in equation (1), is a function of the voltage and the cosine of the signal phase:

$$\mathrm{SNR}_{\mathrm{dir}} = V\cos(\phi). \tag{1}$$

Compared to the direct signal, the reflected signal has a weakened voltage range and phase shift (equations (2) and (3)):

$$\mathrm{SNR}_{\mathrm{ref}} = \beta V\cos(\phi + \Delta\phi);\ 0 < \beta < 1, \tag{2}$$

$$A = \beta V. \tag{3}$$

As mentioned in equation (4), phase shift, which is the phase difference between direct and reflected signals, can be expressed based on the difference in length traveled by the two signals and the signal wavelength

$$\Delta\phi = 2\pi/\lambda \times L. \tag{4}$$

In equation (4), λ is the GNSS wavelength. In the right-angled triangle on the left side of Figure 2, as displayed in equation (5), the length difference can be written as a function of the satellite elevation angle e and length D

$$L = D\sin(e). \tag{5}$$

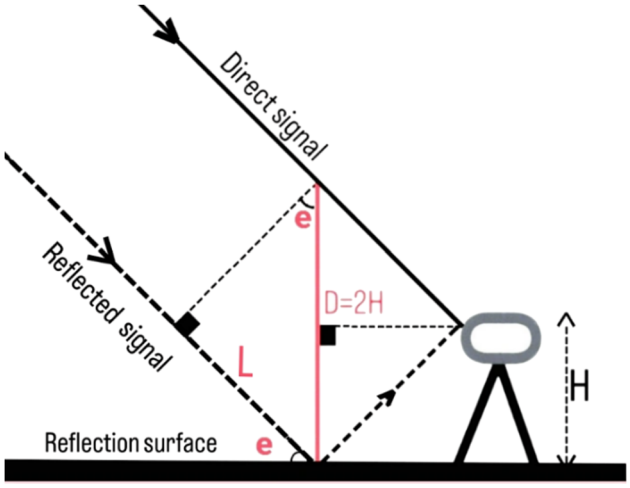

Figure 2: In interferometric reflection measurement using the GNSS, the received signals comprise a combination of direct and reflected signals. In this context, H represents the height of the antenna, e denotes the elevation angle of the satellite, and L represents the distance traveled by both the direct and reflected signals.

In the equilateral triangle on the left side of Figure 2, the barycentric of triangle D is twice the height of the antenna H. Therefore, by combining equations (4) and (5), we can write equation (6):

$$\Delta\phi = 4\pi H/\lambda \times \sin(e). \tag{6}$$

By placing equations (4) and (3) in equation (2), we will have equation (7):

$$\mathrm{SNR}_{\mathrm{ref}} = A\cos(4\pi H/\lambda \times \sin(e) + \phi). \tag{7}$$

As mentioned previously, it is assumed that the SNR data exhibit a constant frequency and amplitude in relation to the sine of the satellite elevation angle, which is a simplification. Simulations conducted by Larson et al. (2008) demonstrate that ϕ is directly correlated with the apparent reflection depth of the GPS signal. When the soil is wet, the apparent reflector is located close to the surface, while as the soil dries, the reflection depth increases by several centimeters. The initial height of the reflector is approximately equal to H, and it is estimated from the SNR data of the reflected signals using equation (7). The data from each satellite orbit can be utilized to estimate the initial height of the reflector for the station. The phase and amplitude are also estimated using the least squares method.

2.3 SNR metrics estimation

The Lomb-Scargle periodogram (LSP) can be used to estimate temporal variations in the dominant frequency of SNR interference. This method has been employed to estimate snow depth from GPS-IR (Larson and Nievinski, 2013). The LSP is similar to the fast Fourier transform, with the distinction that it can be applied to irregularly sampled data. The dominant frequency can be converted into the effective height of the reflector (H_{eff}) using equation (8)

$$H_{\mathrm{eff}(t)} = \frac{1}{2} f_{m,t}\lambda, \tag{8}$$

where f_m is the LSP peak frequency and t is the day counter. The change in the effective height of the reflector (ΔH_{eff}) during the year is defined as `equation (9)`:

$$\Delta H_{\mathrm{eff}(t)} = H_0 - H_{\mathrm{eff}(t)}. \tag{9}$$

The peak power of LSP frequency or LSP amplitude (A_{LSP}) is also affected by soil moisture, vegetation, and other station conditions such as topography. The metrics obtained from SNR data (ΔH_{eff}, A, A_{LSP}, ϕ) are influenced to varying degrees by changes in soil moisture (Chew et al., 2014; Larson et al., 2009; Larson et al., 2008), vegetation coverage (Wan et al., 2015), and snow depth (Larson and Nievinski, 2013). Two modeling studies have shown that phase is the most reliable indicator of soil moisture changes (Chew et al., 2014), and A is the metric most closely related to changes in vegetation permeability and canopy height (Chew et al., 2014). However, phase was also found to be significantly affected by changes in vegetation cover (Chew et al., 2014). Therefore, before estimating soil moisture from SNR data, it is necessary to quantify and remove the effect of vegetation, depending on the extent of vegetation in a station. SNR metrics are calculated for each day and each satellite orbit. The absolute magnitude of A and A_{LSP} is not related to soil moisture or vegetation changes and instead primarily depends on satellite transmit power, antenna gain, and receiver temperature. A and A_{LSP} are normalized as per equations (10) and (11)

$$A_{\mathrm{LSPnorm}(t)} = A_{\mathrm{LSP}(t)}/A_{\mathrm{LSP}_{20\%}}, \tag{10}$$

$$A_{\mathrm{norm}(t)} = A_{(t)}/A_{20\%}. \tag{11}$$

The lower index of 20% indicates the average of the upper 20% of A or A_{LSP} values in the time series for a single satellite orbit. The parameter of 20% may vary depending on the amount of noise in the time series or the total length of the time series. Any normalized value greater than one is set equal to one.

Changes in the H_{eff} time series are calculated using equation (9). Generally, ΔH_{eff} does not vary by more than 5 cm throughout the year, although a significant amount of snow or plant water at the station can cause larger changes. In the next step, the effects of vegetation are quantified using these criteria.

2.4 Vegetation effect quantification

In previous studies (Chew et al., 2014), it has been shown that an increase in soil moisture leads to a decrease in SNR amplitude. If the moisture content increases by 0.4 $(cm)^3$ $(cm)^{-3}$, A_{norm} will decrease from 1.00 to 0.78. Since the difference between residual soil and saturated soil for standard soil is approximately 0.4 $(cm)^3$ $(cm)^{-3}$, A_{norm}, there should not be a decrease in A_{norm} beyond 0.78, unless there is vegetation growth present. Therefore, if a satellite track's A_{norm} time series remains below 0.78, it indicates the effect of something other than soil moisture variations, such as changes in vegetation water content. Conversely, a track with a time series above 0.78 shows minor vegetation effects, making it difficult to determine whether the decrease in A_{norm} is due to vegetation or soil moisture changes. If there is a significant vegetation effect around the station, as determined by the A_{norm} time series, the vegetation effects should be removed before estimating soil moisture.

If there is a significant vegetation effect but it occurs for a short time period, the time phase data with A_{norm} less than 0.78 can be removed without using the vegetation filter. However, simply removing the phase data without implementing vegetation correction leads to errors in the periods around the affected interval, which are still influenced by vegetation growth. Figure 3 provides a summary of the soil moisture estimation algorithm for a single station. To remove the vegetation effect, a linear regression model is used. Linear regression is a statistical modeling technique employed to establish a linear relationship between a dependent variable and one or more independent variables. In the context of removing the vegetation effect before measuring soil moisture, linear regression can be utilized to model the relationship between the GNSS-IR measurements (dependent variable) and the vegetation parameter (independent variable). Equation (12) represents a general equation for a simple linear regression model

$$y = \beta_0 + \beta_1 x + \varepsilon, \tag{12}$$

where y represents the dependent variable (GNSS-IR measurement), x represents the independent variable (vegetation parameter), β_0 is the intercept, which represents the value of y when x is 0, β_1 stands for the coefficient of the independent variable, indicating the change in y for a unit change in x and ε represents the error term or residual, accounting for the variability in y that is not explained by the model. The goal of linear regression is to estimate the values of β_0 and β_1 that minimize the sum of squared residuals (errors) between the predicted values and the actual values of y. This process is typically done through a technique called ordinary least squares (OLS) regression, which calculates the best-fitting line by minimizing the sum of squared residuals. The estimated coefficients, denoted as b_0 and b_1 , are obtained using equations (13) and (14) derived from the OLS method:

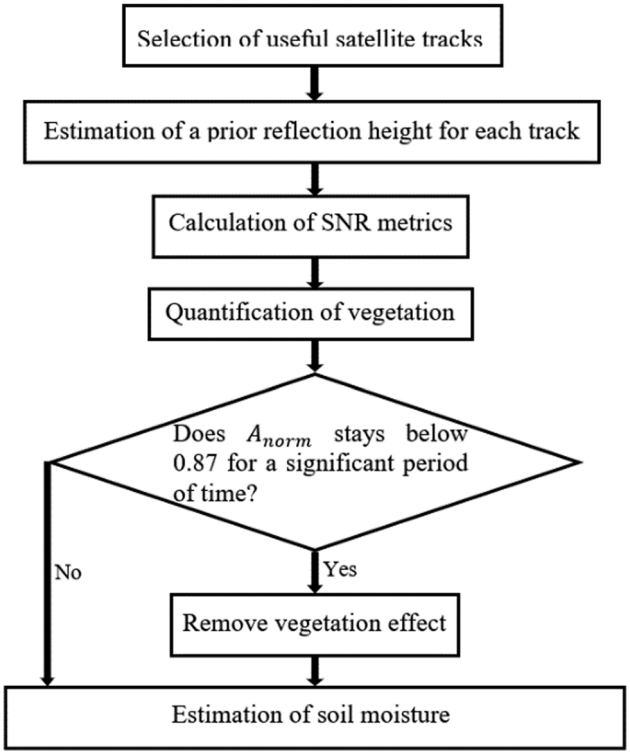

Figure 3: Flowchart displaying soil moisture estimating algorithm for a GPS site.

$$b_1 = \Sigma((x_i - \bar{x})(y_i - \bar{y}))/\sum((x_i - \bar{x})^2), \tag{13}$$

$$b_0 = \bar{y} - b_1\bar{x}, \tag{14}$$

where x_i and y_i represent the individual data points of the independent and dependent variables, respectively, and $\bar{x}$ and $\bar{y}$ represent the means of the independent and dependent variables, respectively.

Once the coefficients (b_0 and b_1) are estimated, the linear regression equation can be used to predict the GNSS-IR measurement (y) based on the vegetation parameter (x). It is important to note that linear regression assumes a linear relationship between the variables and certain underlying assumptions, including the independence of errors, constant variance of errors, and normally distributed errors. These assumptions should be validated or addressed during the model evaluation process.

3 LSTM neural network

LSTM is a type of recurrent neural network (RNN) architecture that was introduced by Hochreiter and Schmidhuber

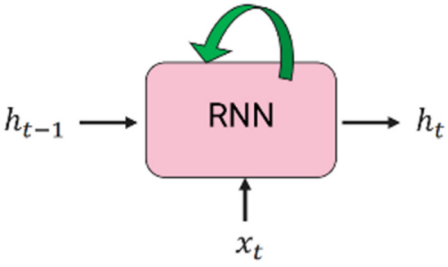

Figure 4: The RNN block, by looking back, indicates short-term memory.

(1997). It was later improved by Gers et al. (2000), as well as by Greff et al. (2017). The structure of a RNN is very similar to that of an LSTM network. In fact, LSTM can be seen as an updated version of RNN. As shown in Figure 4, the RNN block looks back and has short-term memory, just like LSTM. The network takes the output of the previous block (h_{t-1}) and the input vector (x_t) and produces the output of the current block (h_t). The order of input data (x_t) is important for this network.

The relationships of the RNN are defined in the following equations:

$$h_t = f_w(h_{t-1}, x_t), \tag{15}$$

$$h_t = \tanh(w_{hh}h_{t-1}, w_{xh}x_t), \tag{16}$$

$$y_t = w_{hy}h_{t-1}, \tag{17}$$

where w stands for weights and y_t for network output. Figure 5 shows the block diagram of the RNN, in which W shows weights and L displays the memories. The hyperbolic tangent function (tanh) is the main part of the blocks. A part of the outputs (y_i) is sent to the memories L_i.

3.1 Types of gates in the LSTM network

The LSTM network has three gates. The forget gate, which deletes and forgets information; the input gate, which receives information; and the output gate. Memory in the LSTM network is managed using these three gates (Gers et al., 2000). The gate-based block diagram of the LSTM network is shown in Figure 6.

Figure 7 shows the complete structure of the LSTM network, with two sets of weights w_i and w_h, as defined in equations (18) and (19) each containing four weight coefficients. The first index specifies the type of weight, and the second index specifies the type of gate

$$w_i = \{w_{if}, w_{ii}, w_{ig}, w_{io}\}, \tag{18}$$

$$w_h = \{w_{hf}, w_{hi}, w_{hg}, w_{ho}\}. \tag{19}$$

3.1.1 Forget gate

The forget gate itself is similar to a neural network (Figure 8). The relationship of the forget gate in the LSTM network is depicted by equations (20) and (21).

$$f_t = \sigma(w_{hf}h_{t-1} + w_{if}x_t + b_{hf} + b_{if}), \tag{20}$$

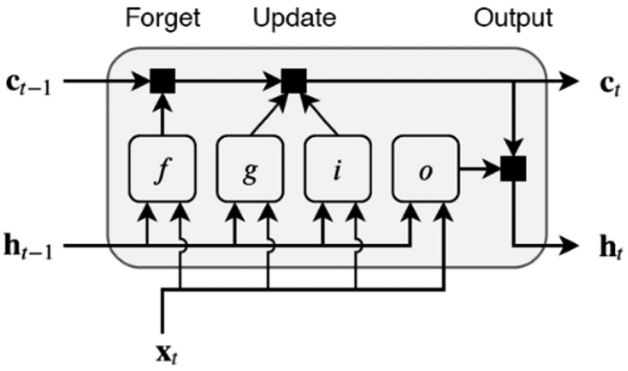

Figure 6: The gate-based block diagram of the LSTM network which includes three gates: forget gate (f), input gate (i), and output gate (o).

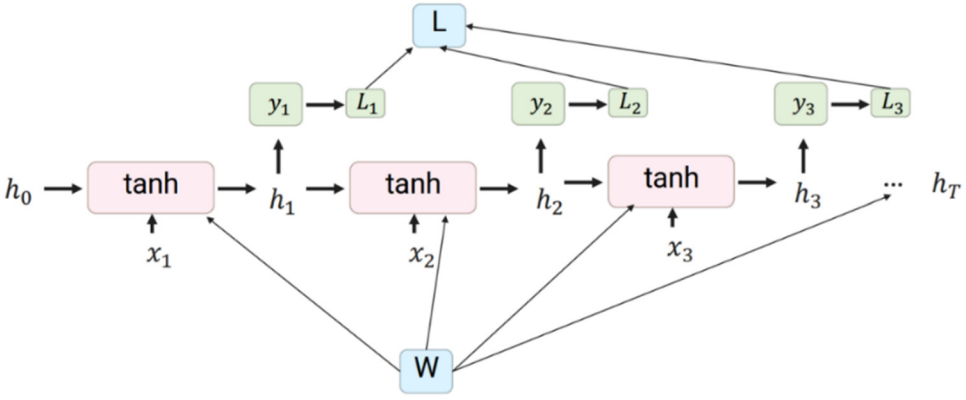

Figure 5: The block diagram of the RNN shows that the weight matrix W affects both the input vectors x and the hidden states h.

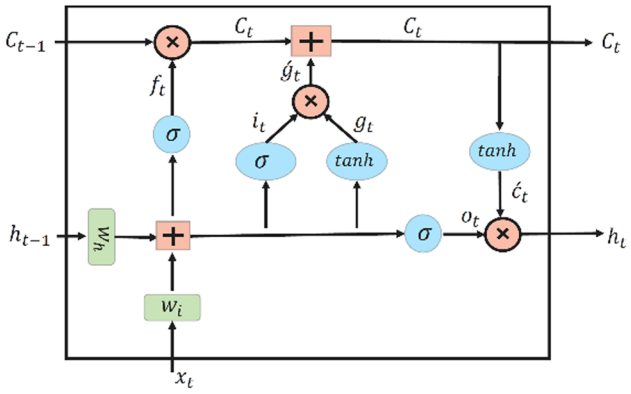

Figure 7: The complete structure of the LSTM network.

where f_t represents the vector output of the forget gate at time step t. It determines how much of the previous cell state should be forgotten or retained. σ symbol denotes the sigmoid activation function, which is commonly used in LSTM networks to squash the output values between zero and one, representing the strength of the gate. w_{hf} stands for the weight matrix associated with the previous hidden state (h) that is multiplied by the forget gate's weight. h_{t-1} displays the previous hidden state at the time step ($t-1$). It is the output of the LSTM cell at the previous time step. w_{if} represents the weight matrix associated with the current input (x_t) at time step t that is multiplied by the forget gate's weight. b_{hf} is the bias term associated with the previous hidden state (h), and b_{if} is the bias term associated with the current input. f_t is multiplied element-wise with c_{t-1}.

$$c_t = c_{t-1} \odot f_t, \tag{21}$$

where c_t represents the current cell state at time step t. It is the output of the forget gate and determines the memory information to be stored in the current time step. c_{t-1} denotes the previous cell state at time step ($t-1$). It is the cell state output of the LSTM at the previous time step, which carries the memory information from the past. The symbol $\odot$ represents the element-wise multiplication (also known as the Hadamard product) operation. It multiplies the elements of the forget gate output f_t and the previous cell state (c_{t-1}) element-wise.

If information is to be discarded, the value of the forget layer is set to zero. Conversely, if the information is to be retained in memory, the value of the layer is set to one. If the information is modified by the layer, the value of σ lies between zero and one. Thus, the role of the forget gate is to delete, retain, or modify information.

3.1.2 Input gate

Input gate refers to the mechanism for reviewing and storing new information in long-term memory. The relationships of the input gate are specified in equations (22)–(24):

$$i_t = \sigma(w_{hi}h_{t-1} + w_{ii}x_t + b_{hi} + b_{ii}), \tag{22}$$

$$g_t' = g_t \odot i_t, \tag{23}$$

$$c_t = c_t + g_t'. \tag{24}$$

Figure 9 highlights the position of the input gate in the overall structure of the LSTM network in yellow.

3.1.3 Output gate

There should be a gate that determines which pieces of information are valuable and should be retained, and which pieces are no longer needed and should be

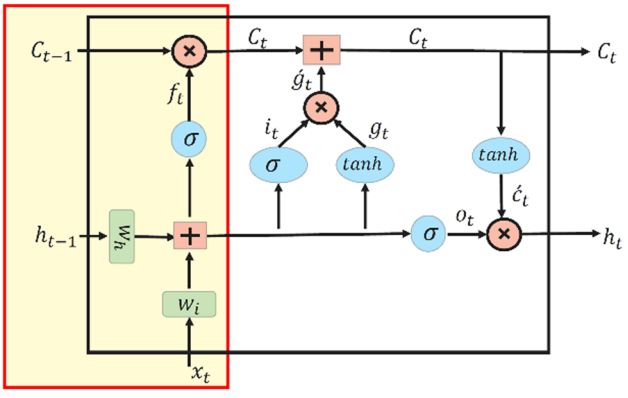

Figure 8: The position of the forget gate in the overall structure of the LSTM network is highlighted in yellow.

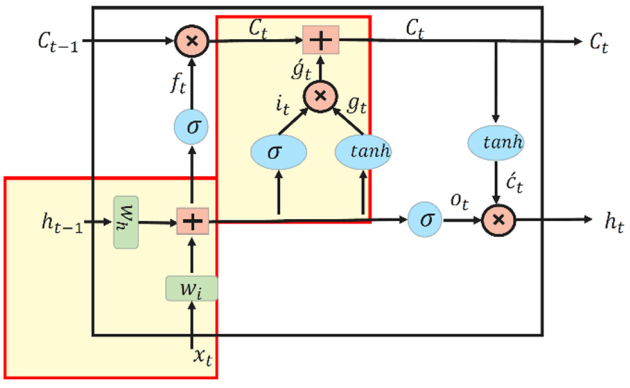

Figure 9: The position of the input gate in the overall structure of the LSTM network is colored in yellow.

discarded. This process is known as the memorization process, and it is performed by the output gate. The relationship for memorization in the LSTM network is defined by equation (25).

$$g_t = \tanh(w_{hg}h_{t-1} + w_{ig}x_t + b_{hg} + b_{ig}). \tag{25}$$

The value of g_t determines whether the information should be stored in memory or forgotten. The assessment of the importance or irrelevance of g_t is the responsibility of the output gate. The output gate is similar to the forget gate, as it is positioned along the path of g_t or the memorization process to determine its significance or irrelevance. In Figure 10, the placement of the memorization process within the overall structure of the LSTM network is highlighted in yellow.

So, the output gate determines how much of the information in long-term memory should be transferred to the output. The structure of the output gate is similar to other gates in the LSTM network. Figure 11 highlights the position of the output gate in the overall structure of the LSTM network, indicated in yellow.

$$o_t = \sigma(w_{ho}h_{t-1} + w_{io}x_t + b_{ho} + b_{io}), \tag{26}$$

$$h_t = o_t \odot \tanh(c_t). \tag{27}$$

In equations (26) and (27), c_t and h_t are the outputs of the output gate and the entire LSTM structure, respectively.

In this article, LSTM neural networks are used for modeling and predicting the time series of SMC.

3.2 Hyperparameter optimization

Hyperparameters encompass a collection of parameters utilized for both the training and testing phases to facilitate the learning process. The deep learning model is designed to learn distinct feature combinations and patterns through a diverse set of instances and weights. Typically, hyperparameters encompass a range of factors such as the learning rate, number of iterations, hidden layers, batch size, activation functions, momentum, and regularization. The selection of hyperparameters varies depending on the specific problem and the deep learning model being used. Different models do not have a universally applicable set of optimal hyperparameters. Effectively choosing hyperparameters greatly influences the enhancement of predictive performance. The objective of hyperparameter optimization is to select the values that yield the most optimal results during the validation process. The following equation provides a representation of hyperparameter optimization:

$$y^* = \text{argmin} f(y). \tag{28}$$

The variable y is an element of the domain space Y. The objective score, denoted as $f(y)$, corresponds to the evaluation of the error rate on the validation set, with the goal of minimizing it. Conversely, y denotes a set of hyperparameters that reside within a domain where the evaluation of hyperparameter optimization is characterized by higher costs. The training time for complex deep neural network models, coupled with a multitude of hyperparameters, exceeds human capacity and becomes infeasible to manage. Various techniques are employed for hyperparameter optimization, including grid search, random search, and Bayesian search.

3.2.1 Grid search

The most fundamental approach to hyperparameter tuning is grid search. In summary, the hyperparameters are partitioned into a discrete grid, simplifying their domain. Next,

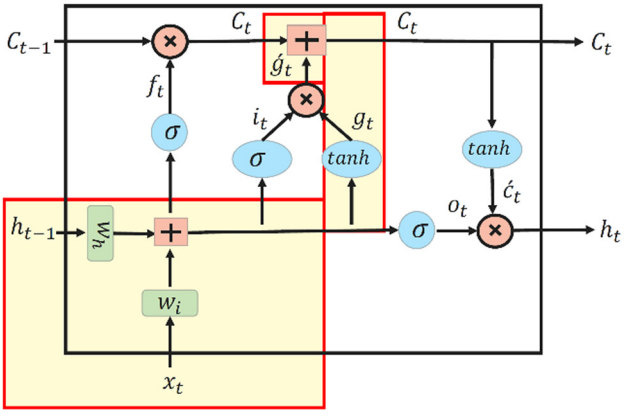

Figure 10: The position of the memorization process in the overall structure of the LSTM network is highlighted in yellow.

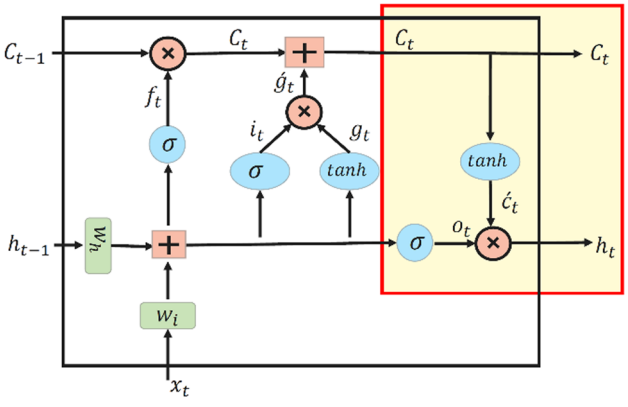

Figure 11: The position of the output gate in the overall structure of the LSTM network is highlighted in yellow.

employing cross-validation, we explore all potential combinations of values within the grid, evaluating multiple performance metrics in the process. The point on the grid that maximizes the average value in cross-validation represents the optimal combination of hyperparameter values. Grid search is a thorough and exhaustive approach that takes into account all conceivable combinations to identify the optimal point within the given domain. A notable drawback is its relatively slow speed, which is its primary disadvantage.

3.2.2 Random search

Random search involves defining distributions for each hyperparameter, which can be uniformly defined or sampled using a specific method. Random search differs from grid search in that it does not evaluate all values and instead randomly selects and tests values during the search process. In order to carry out the search, it is essential to specify the number of iterations that will be tested. Compared to grid search, random search demonstrates a relatively faster performance.

3.2.3 Bayesian search

Grid and random search are conventional hyperparameter optimization techniques that offer slight improvements over manual tuning. However, these methods lack the ability to incorporate past results, which hinders their capacity to select the subsequent set of hyperparameters during model training. Grid and random search, as optimization techniques, do not take into account previous evaluations, which is a significant limitation when utilizing these methods (Bergstra and Bengio, 2012). Bayesian model-based optimization provides a potential solution to overcome this limitation. In the context of hyperparameter optimization, Bayesian optimization, as discussed by Dewancker et al. (2015), effectively utilizes prior results to select the optimal hyperparameters for evaluation. The application of Bayesian optimization has been documented in the field of machine learning models and simulations, as explored in the work of Bergstra et al. (2013). It assists in the formulation of the laborious process of optimizing a considerable number of parameters.

In the realm of optimization, the utilization of Bayes' theorem within an equation is fundamental, as it establishes the relationship between a given model m and the corresponding observation d (Kramer et al., 2011):

$$P(m \mid d) = P(d \mid m)P(m)/P(d), \tag{29}$$

where $P(m|d)$ represents the posterior probability of m given d, $P(d|m)$ indicates the likelihood of d given m, $P(d)$ signifies the prior probability of d, and $P(m)$ denotes the marginal probability of m. Bayesian optimization is employed to search for the minimum value of a function, $f(y)$, within a specified bounded domain, Y, by iteratively evaluating different points in the parameter space.

Tuning the hyperparameters of an LSTM neural network using Bayesian optimization follows a similar process to that of any deep neural network. However, there are specific hyperparameters that are relevant to LSTM networks that can be optimized.

Some of the hyperparameters that can be tuned using Bayesian optimization for an LSTM network include

1. Number of LSTM units: This hyperparameter determines the number of memory cells in the LSTM layer.
2. Learning rate: The learning rate controls the step size during the optimization process.
3. Dropout rate: Dropout is a regularization technique used to prevent overfitting. The dropout rate determines the probability of dropping out units during training.
4. Batch size: The batch size determines the number of training samples used in each iteration of the optimization process.
5. Number of LSTM layers: This hyperparameter determines the depth of the LSTM network by specifying the number of LSTM layers.
6. Activation function: The activation function defines the non-linear transformation applied to the LSTM units.

By defining the search space for these hyperparameters and using Bayesian optimization, the process can efficiently explore and find the optimal set of hyperparameters for an LSTM neural network, leading to improved performance and generalization.

Tuning the hyperparameters of a neural network using Bayesian optimization involves several steps:

1. Define the search space: First, we need to define the hyperparameters we want to optimize and specify their ranges or distributions.
2. Choose an objective metric: Select an objective metric that quantifies the performance of the model. This can be accuracy, loss, or any other suitable metric that reflects the quality of the model's predictions.
3. Select a probabilistic model: Bayesian optimization uses probabilistic models to model the unknown objective function, which maps hyperparameter configurations to their corresponding metric values. Gaussian processes (GPs) are commonly used as the probabilistic model in Bayesian optimization. GPs define a distribution over functions and provide a flexible way to model the unknown objective function mapping hyperparameter configurations to metric values. GPs capture the

uncertainty in the function estimates and allow for efficient posterior inference.

4. Specify an acquisition function: Acquisition functions guide the search process by evaluating the utility or potential benefit of evaluating a specific hyperparameter configuration. These functions balance exploration and exploitation to effectively search the hyperparameter space. Common acquisition functions include Expected Improvement (EI), Probability of Improvement (PI), and Upper Confidence Bound (UCB).

– EI is a popular acquisition function that quantifies the expected improvement over the current best metric value. It considers both the mean and variance estimates from the probabilistic model and focuses on regions where the expected improvement is high.
– PI measures the probability of finding a hyperparameter configuration with a metric value better than the current best. It aims to maximize the probability of finding an improvement.
– UCB balances exploration and exploitation by adding an exploration term to the mean estimate of the probabilistic model. The exploration term encourages sampling from regions with higher uncertainty.

5. Iterative search process: The Bayesian optimization process involves an iterative search procedure. It starts with an initial set of hyperparameter configurations, often selected randomly or using heuristics. The model is trained and evaluated using these configurations to obtain the corresponding metric values.

Bayesian optimization can be computationally expensive for large and complex neural networks. Several techniques can be employed to make the search process more efficient:

– Parallelization: Evaluating multiple hyperparameter configurations in parallel can speed up the search process. This can be achieved by utilizing multiple compute resources or distributed computing.
– Warm-starting: Warm-starting involves using the knowledge gained from previous optimization runs to initialize the probabilistic model. This can help accelerate the convergence of the optimization process.
– Sequential model-based optimization (SMBO): SMBO combines Bayesian optimization with other optimization techniques, such as random search or grid search, in the early stages of the search. This approach balances exploration and exploitation, leveraging the strengths of both methods.
– Bayesian optimization can handle constraints on hyperparameters by constraining the search space or using constraint-aware acquisition functions. For example, if a hyperparameter has a range constraint, the search space can be restricted to that range.

6. Update the probabilistic model: The probabilistic model is updated with the observed hyperparameter configurations and their metric values. The GP model, for example, is updated using a process called Bayesian inference, which uses the observed data to update the prior distribution and obtain a posterior distribution.
7. Select the next hyperparameter configuration: The acquisition function is used to determine the next hyperparameter configuration to evaluate based on the current probabilistic model. The acquisition function evaluates the potential benefit of sampling different hyperparameter configurations, considering both exploration and exploitation.
8. Evaluate the model with the selected configuration: The model is trained and evaluated using the selected hyperparameter configuration. The corresponding metric value is then used to update the probabilistic model.
9. Repeat iterations: Steps 7 and 8 are repeated for a specified number of iterations or until a termination criterion is met. The iterations progressively refine the probabilistic model and guide the search toward hyperparameter configurations that are likely to yield better results.
10. Select the best hyperparameters: Once the search process is complete, the hyperparameter configuration corresponding to the best metric value is selected as the optimal configuration. This configuration can then be used to train the final model.

Bayesian optimization efficiently explores the hyperparameter space by iteratively updating the probabilistic model based on the observed data. It intelligently selects hyperparameter configurations to evaluate, focusing on regions likely to yield good results. However, it is important to note that the effectiveness of Bayesian optimization depends on factors such as the complexity of the problem, the size of the search space, and the availability of computational resources. Experimentation and tuning of the optimization process itself may be necessary to achieve the best results.

3.3 Performance evaluation

Numerous performance statistical indices have been suggested in scholarly works to aid in evaluating the predictive capability and accuracy of diverse models.

In this article, root mean square error (RMSE), normalized root mean square error (NRMSE), Pearson's rank

correlation coefficient (ρ), and coefficient of determination (R^2) were employed to assess the performance of the derived model.

In principle, a lower RMSE value signifies more precise prediction outcomes (a null value indicates a perfect fit), whereas higher R^2 values indicate a stronger correspondence between the analytical and predicted values (a value of one signifies a perfect fit).

The predictive ability of a model is assessed by Pearson's correlation coefficient, R, which quantifies the variance explained by the model. R values range from zero to one, with values close to one indicating a strong predictive ability and values close to zero indicating inadequacy.

These performance metrics provide a reliable assessment of the overall predictive accuracy of an ANN. The mentioned statistical parameters have been computed using equations (30)–(33) (Apostolopoulou et al., 2020; Armaghani and Asteris, 2021):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{30}$$

$$\text{NRMSE} = \frac{\text{RMSE}}{\bar{x}}, \tag{31}$$

$$R^2 = 1 - \left\{\frac{\sum_{i=1}^{n}(x_i - y_i)^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right\}, \tag{32}$$

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2 - 1)}, \tag{33}$$

where n is the number of observations, x_i and y_i represent the predicted and target values, and d_i is the difference between the two ranks of each observation.

In addition to the aforementioned indices, the authors have introduced a novel performance index, referred to as the a20-index, specifically designed to assess the reliability of the developed soft computing techniques:

$$a20\text{-index} = \frac{m20}{M}. \tag{34}$$

In equation (34), M represents the total count of dataset samples, and m20 denotes the count of samples where the ratio of "Experimental value" to "Predicted value" falls between 0.80 and 1.20. It should be noted that in an ideal predictive model, the a20-index is anticipated to be one. The proposed a20-index signifies the percentage of samples with predicted values that deviate within ±20% of the corresponding experimental values.

4 Case study

Our discussion focuses on PBO station P038, located in Portales, New Mexico, at 34.14726°N and −103.40734°W, as shown in Figure 12(a). The monument at P038 is of the deep-drilled braced type. Both the station and its monument have been installed since 2015 and have been operational up to the present day. For our study, we will analyze the data collected from 2017 to 2020.

Our analysis consists of two main steps. In the first step, we will analyze the data using the normal RH (GNSS-IR). In the second step, we will utilize the results obtained from the first step to run the soil moisture code.

Before we start our study, we have to calculate the GNSS reflection zones to make sure that we can sense

(a) (b)

Figure 12: (a) Photograph of a GPS site, called P038, in Portales, New Mexico. The antenna phase center at this site is approximately 2 m above the soil surface. The station has been installed since 2015. (b) Google Earth image for PBO site p038 and reflection zones for 2-m RH and satellites with elevation angles between 5° and 15°.

the surface we want to measure. For this, we need to check the position of our site and get the gaps in sensing zones and barriers exact position that may cause distortion in our work so if there is a need we can have an azimuth or elevation angle mask. As displayed in Figure 12(b), there is a gap in the reflection zone for our site. These are the areas where GPS signals are weak or unavailable, this can happen for several reasons such as obstructions, atmospheric conditions, or receiver limitations. Sometimes GPS receives have limitations in terms of their sensitivity and accuracy, which can result in weak or inaccurate signals in certain areas. The reflective areas are constant over time and is actually not a function of time, but it is a function of the position of the receiver. The inclination angle of the global positioning system (GPS) satellites' orbit relative to the equator is different depending on the type of satellite. The orbital inclination angle for USA GPS satellites is about 55°, Russian GLONASS satellites are about 65°, EU Galileo satellites are about 56°, and China BeiDou satellites are about 55°. Therefore, the usual inclination angle of the orbit of GNSS satellites varies between 55° and 65°, and from this inclination angle to the North Pole, practically no satellite rises or sets, and for this reason, in a range around the north, the reflected signal to be received by the receiver practically does not exist.

The gap below shows the area that satellites have not been able to cover; as there is not any data recorded in that azimuths, there is no need to use azimuth mask. According to Figure 12(b) all chosen elevation angles and azimuths are suitable. After analyzing the reflection zones of our site and choosing the best elevation angle and azimuths, we will generate SNR data from RINEX files; SNR data are converted from dB-Hz to a linear scale (volts/volts) for each rising and setting satellite track. We are going to use multipath signals to generate SNR data. We can take a quick look at our data to quickly test various options (elevation angles, frequencies, azimuths, and quality control [QC] parameters) of our site so we can choose the most suitable ones.

Figure 13 illustrates the LSP of the SNR data extracted from the L2C observations received by the P038 station for the 100th day of 2019. The periodogram is presented for the four quadrants: northeast, northwest, southeast, and southwest.

In the case of a flat reflector, the LSP diagram would exhibit a single peak indicating the height of the reflector, and any variations in the peak would indicate changes in the reflector's height over time. However, if the monitored area has topography, multiple peaks will be present in the LSP diagram.

It is important to note that the *x*-axis of the diagram does not exceed 6 m, and the results on the *x*-axis start at

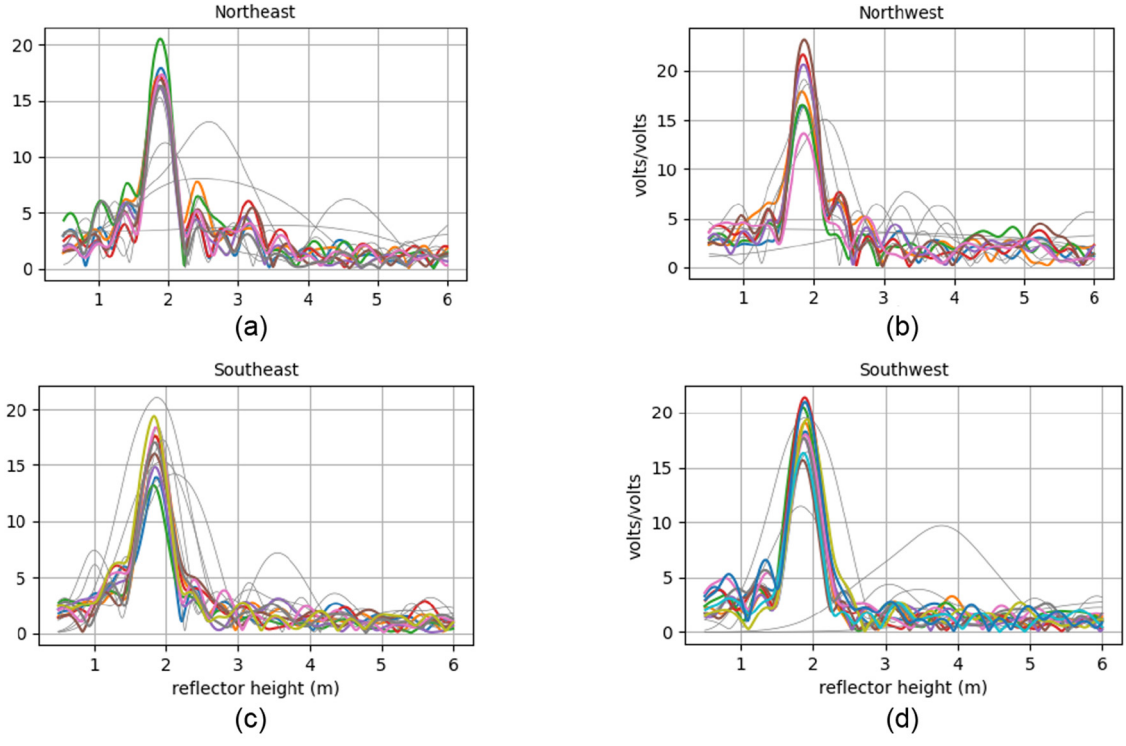

Figure 13: LSP periodogram of SNR data extracted from L2C observations in four quadrants (a) northeast, (b) northwest, (c) southeast, and (d) southwest for station P038 on the 100th day of 2019.

0.5 m. This means that RHs below 0.5 m cannot be resolved using this method.

The LSP periodograms enable us to determine whether there is a planar reflector beneath our antenna. In this case, the peaks in the periodograms cluster around 2 m, suggesting that the antenna phase center is approximately 2 m above the ground at this site. Each color in the diagram represents a different satellite. The data plotted in gray indicate failed reflections. The quadrants referenced are the northwest, northeast, southeast, and southwest quadrants.

According to Figure 14(a), the blue dots represent successful retrievals of reflector height (RH), while the gray dots indicate unsuccessful retrievals. In Figure 14(b), the peak-to-noise ratios are shown, and Figure 14(c) displays the amplitude of the spectral peak. The dashed lines in the figures represent the QC metrics used for the quick look analysis.

From Figure 14(a), we observe that the retrieved RHs are relatively stable at azimuths between 130° and 250° for our station. Figure 14(b) suggests that a peak-to-noise QC metric of 3 is acceptable. Additionally, as shown in Figure 14(c), the amplitudes generally exceed 11, so a minimum value of 8 can be accepted.

Next, we need to estimate prior RHs for our site. The soil moisture algorithm utilizes GPS satellites, and for accurate soil moisture measurements, it is preferable to use a flat area. In our case, L2C data are more effective as it provides higher quality compared to other GPS signals. L2C refers to a specific frequency used in GNSS, such as GPS. The L2C frequency is transmitted by GPS satellites and is primarily utilized for civilian applications.

When employing GNSS-IR for measuring soil moisture, the L2C frequency offers several advantages. First, it has a longer wavelength compared to higher GNSS frequencies. Longer wavelengths are less affected by vegetation and can penetrate through vegetation canopy to some extent. This characteristic enables GNSS-IR to provide soil moisture measurements beneath vegetation cover, which is valuable for agricultural and hydrological applications.

Another reason for considering L2C frequency as more suitable in our study is its sensitivity to changes in the soil's dielectric properties, which are influenced by SMC. By analyzing the reflected L2C signals from the ground, it is possible to estimate soil moisture with reasonable accuracy. This sensitivity to soil moisture variations is exploited in GNSS-IR techniques.

Availability and compatibility of L2C data are additional factors to consider. L2C frequency is widely available in modern GPS receivers and is designed to enhance positioning accuracy and reliability. This makes it convenient to utilize L2C signals for GNSS-IR soil moisture measurements, as compatible receivers are more accessible.

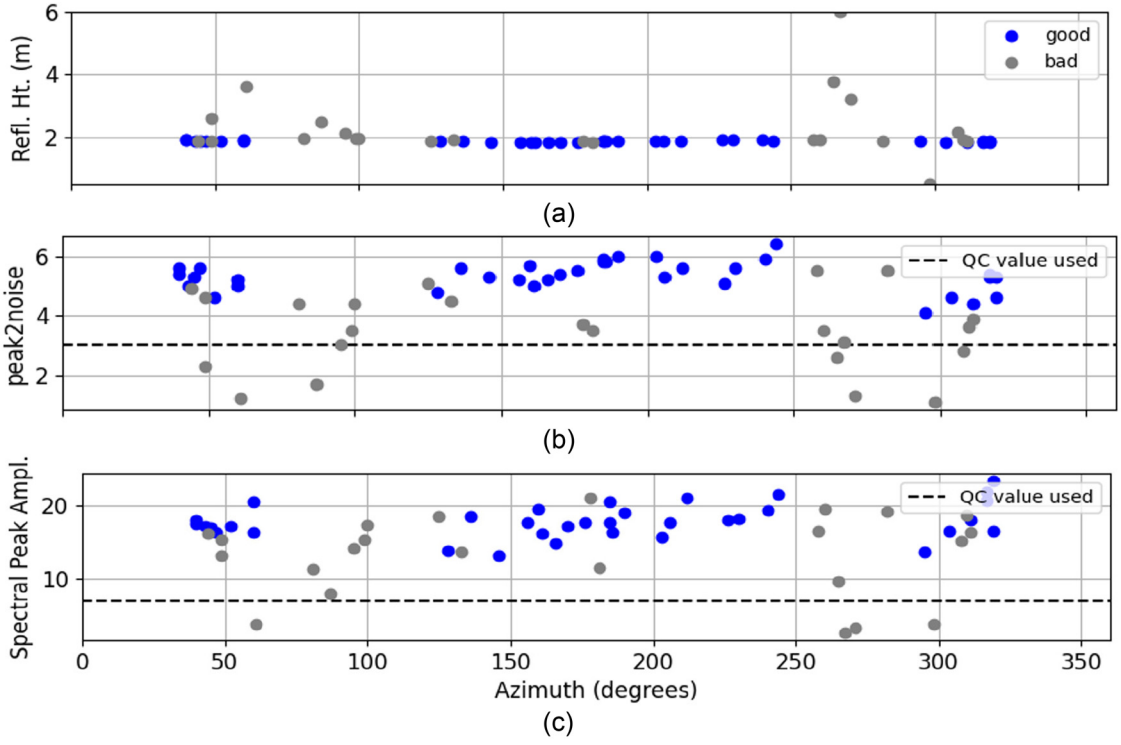

Figure 14: A summary of various QC metrics: (a) RH, (b) peak-to-noise ratio, and (c) spectral peak amplitude.

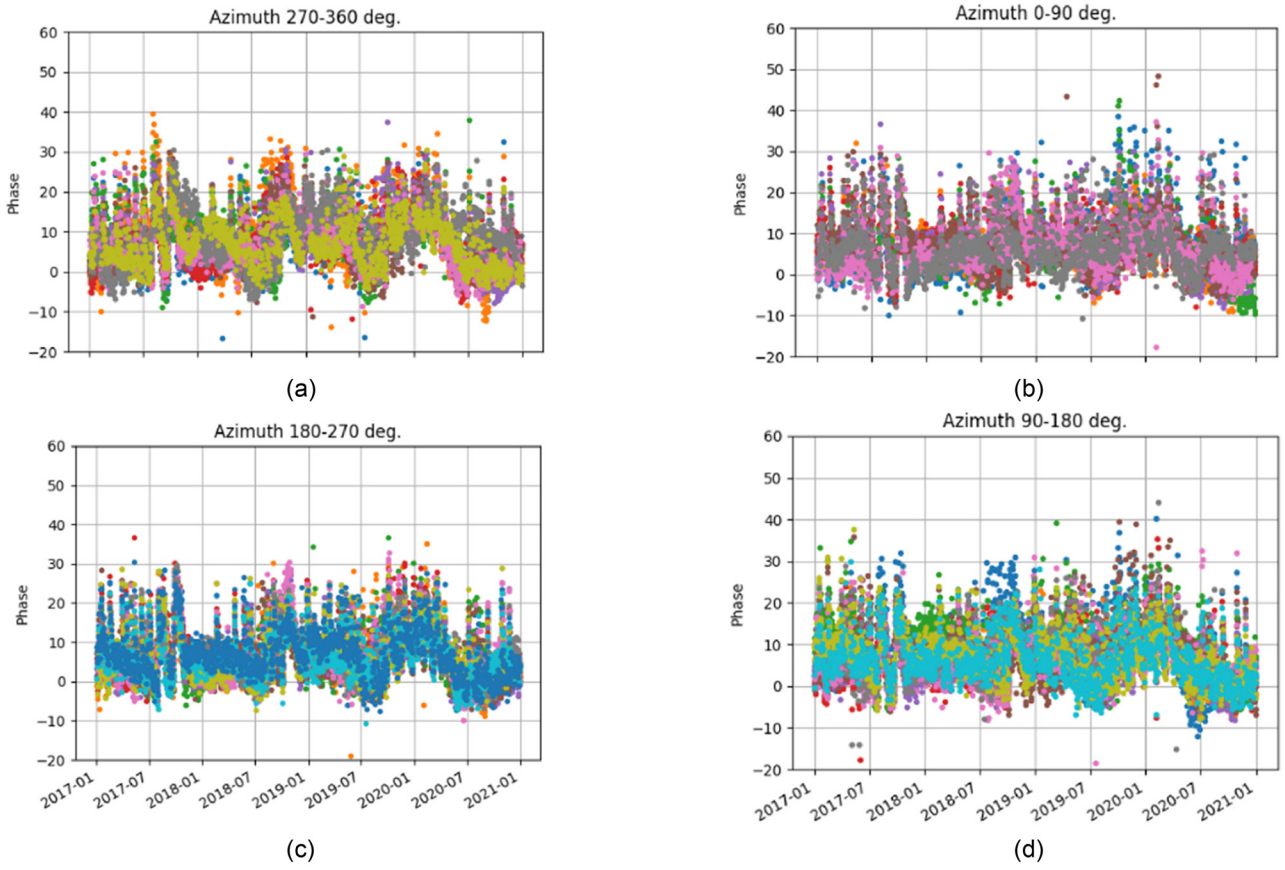

Figure 15: Phase results of all satellites for each day for four geographic quadrates: (a) northwest, (b) northeast, (c) southwest, and (d) southeast.

To summarize, using the L2C frequency in GNSS-IR can be effective for measuring soil moisture due to its longer wavelength, sensitivity to soil moisture changes, and compatibility with modern GPS receivers. However, it is important to note that GNSS-IR is a developing technology, and further research and advancements are needed to improve its accuracy and expand its applications.

In this case, the sites supported are those that are 1–3 m above the soil. In station P038, the antenna is positioned 2 m above the ground. High-rate data is not necessary, so it can be decimated to 15 s when translating the RINEX file to the SNR file format. The elevation angle is limited between 5° and 30°, as data below 5° may be obstructed by trees or tall buildings. However, the elevation angles used here are primarily affected by multipath.

In the second step, the best satellite tracks are identified, with the default being all rising and setting L2C satellite arcs. It is advisable, if possible, to choose the latest year's data to ensure the availability of the most L2C satellites. L2C data are used because the case pertains to soil moisture. To ensure access to L2C data, RINEX 3 files can be utilized. Following this, the phase estimation for each satellite track on each day is performed. The results of the phase estimation will be plotted for four geographic quadrants, as depicted in Figure 15.

The phase results obtained in the previous stage, as displayed in Figure 16, will be converted to VWC. However, vegetation has a significant impact on the phase results, and it is necessary to remove this effect to achieve accurate soil moisture estimates. Therefore, before changing the units from phase (degrees) to VWC, we will model and eliminate the vegetation effects using spectral altitude. Figure 17(a) illustrates the phase results with and without vegetation correction. The vegetation correction helps to mitigate the influence of vegetation and improve the accuracy of the soil moisture estimates. Finally, by incorporating soil texture profiles specific to our site, we can normalize the data shown in Figure 17(b). It is important to note that nonsensical soil moisture values, such as negative values, are not permissible in this context.

As depicted in Figure 17(b), the VWC level was 8.88% in 2017, which increased to 11.74% in 2018. There was a slight decrease in the following year, with the level reaching 10.88% in 2019. In 2020, the level rose to 12.49%.

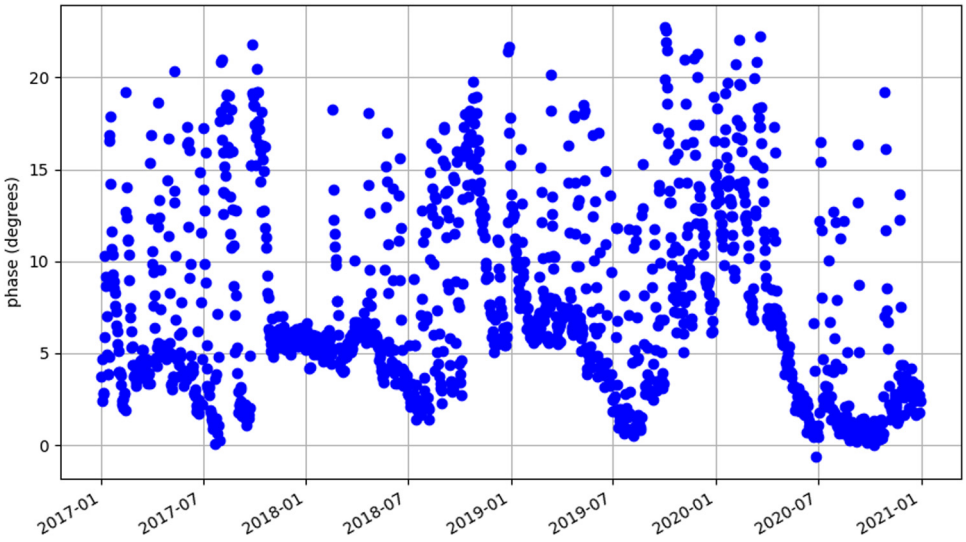

Figure 16: Daily L2C phase results.

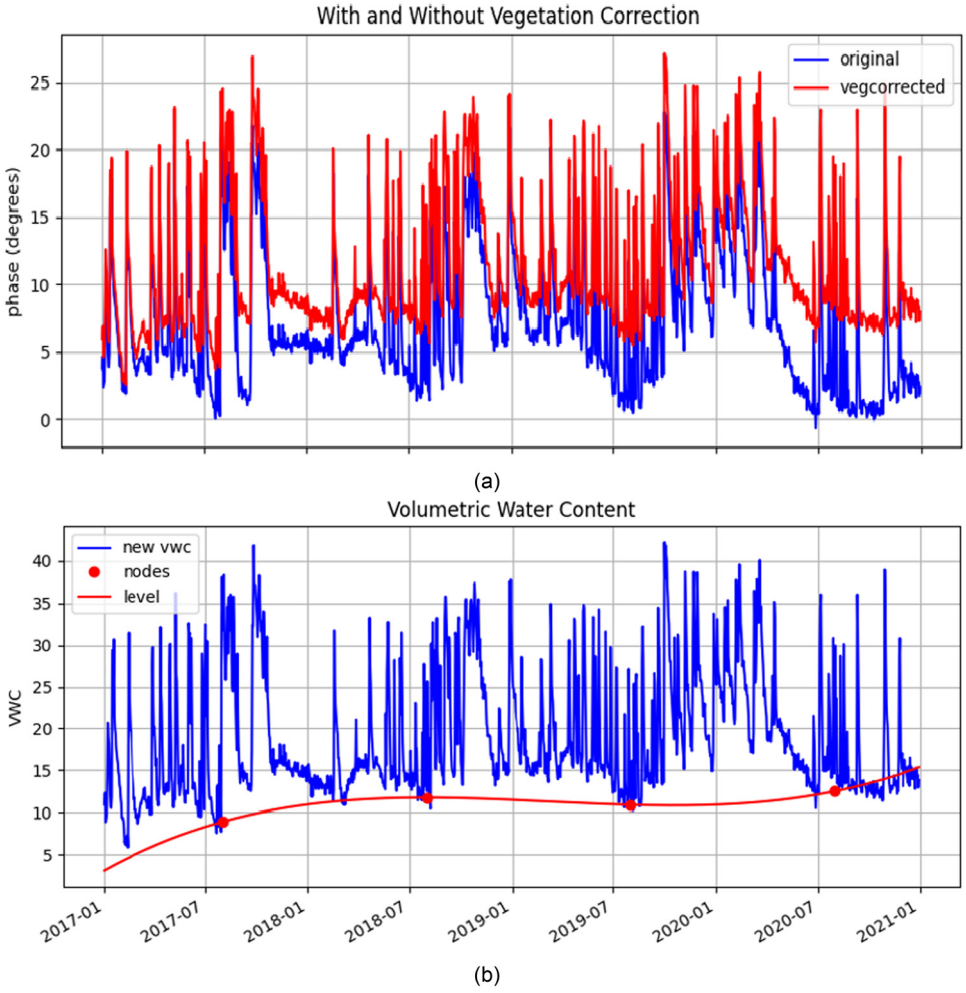

Figure 17: (a) Phase results with and without vegetation correction. (b) VWC and its leveling.

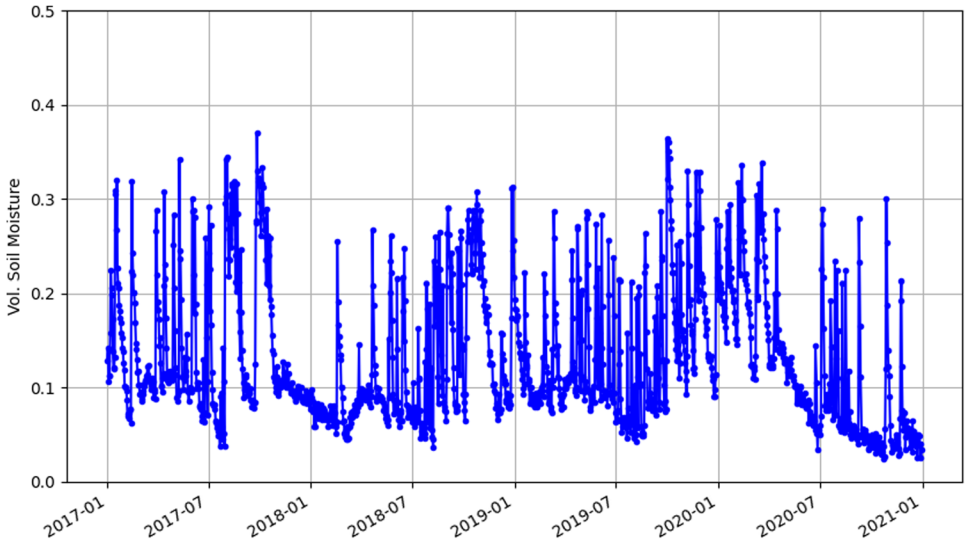

Figure 18: Final VWC for P038.

Figure 18 presents the final result of our work. The horizontal axis represents the dates, spanning from 2017 to the end of 2020, reflecting the duration of our study. The vertical axis represents the volumetric soil moisture on each day during this time period.

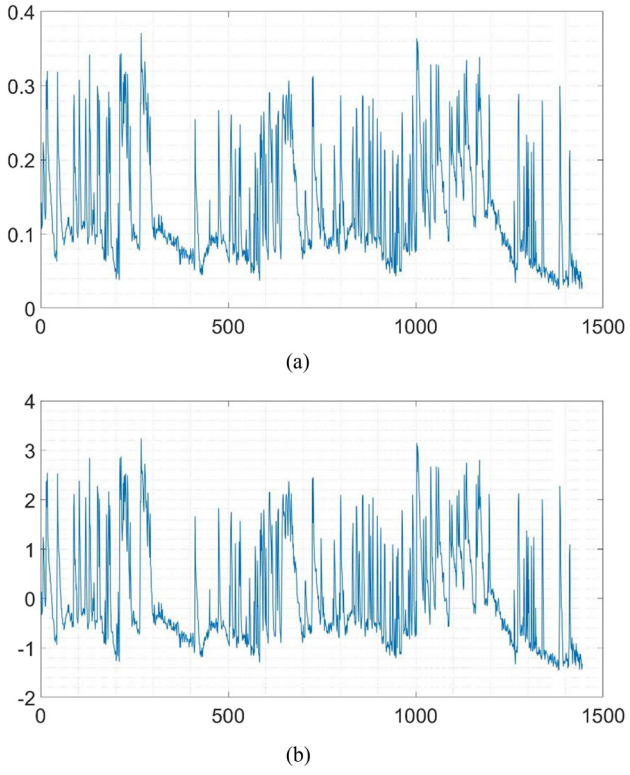

Figure 19: (a) Original, mean = 0.1322, std = 0.073749 versus (b) normalized, SMC data.

VWC is a measure of the ratio of water volume to soil volume unit. It represents the volume of water contained within a given volume of soil. VWC can be expressed as a ratio, percentage, or depth of water in each soil depth, assuming a unit surface. It indicates the proportion of the soil volume that is filled with water. Typically, VWC is expressed as a percentage and ranges from 0 to 100%. A VWC of 0 signifies completely dry soil with no water, while a VWC of 100% indicates that all voids in the soil are filled with water.

In the final step of this article, the objective is to accurately predict soil moisture several days in advance. This prediction solution will assist farmers in optimizing their irrigation schedules more efficiently. For predicting soil moisture time series, the assumption is made that the behavior of the time series remains relatively consistent over different time periods. By considering a limited history of the time series, future values can be predicted using LSTM neural networks with hyperparameters tuned using Bayesian optimization.

Figure 19 displays the original SMC data with a mean of 0.132 and a standard deviation of 0.074, along with the normalized data. The normalization process is used to transform the data to a common scale, typically between

Table 1: Optimum architecture LSTM-tuned hyperparameters using Bayesian optimization

Number of layers	Number of units	Initial learning rate	L2 regularization
1	111	0.011481	0.0096801

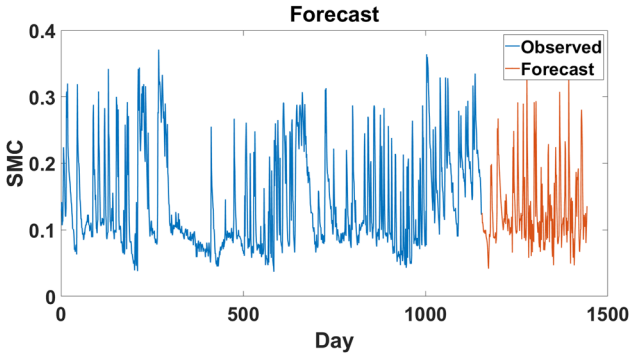

Figure 20: Observed (blue) and forecasted (brown) SMC data.

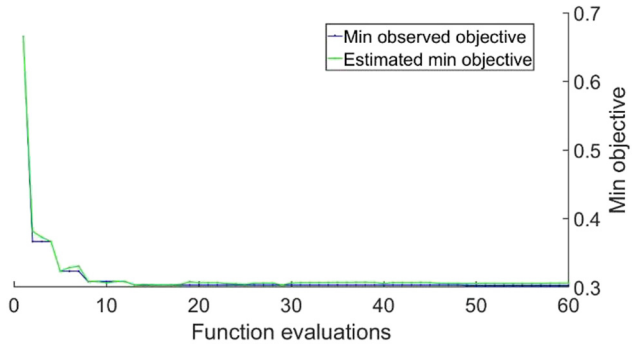

Figure 21: Minimum objective versus number of function evaluations.

Table 2: Statistical indexes of the optimal LSTM model with hyperparameters tuned by Bayesian algorithm

Data	Indices				
	RMSE	**NRMSE**	R^2	ρ	**a20-index**
Train	0.040854	0.29302	0.54427	0.829	0.7719
Test	0.040636	0.41276	0.43659	0.857	0.7273
All	0.040886	0.31158	0.55004	0.849	0.7362

0 and 1, to facilitate the training and performance of the LSTM neural network.

Splitting the dataset into separate training, validation, and test sets is crucial for evaluating the model's performance and preventing overfitting. The training set is utilized to train the model, while the validation set is employed to fine-tune hyperparameters such as the number of layers, number of units, initial learning rate, and L2 regularization. Additionally, the validation set helps monitor the model's performance during training. Finally, the test set is used to assess the model's final performance after completing the training and hyperparameter tuning processes. It is essential to ensure that the test set remains completely separate throughout the training and validation procedures to obtain an unbiased evaluation.

Table 1 shows the optimum architecture of LSTM-tuned hyperparameters using Bayesian optimization. Observed and forecasted SMC data after LSTM hyperparameters optimization with the Bayesian algorithm is shown in Figure 20. The performance is evaluated in terms of RMSE, NRMSE, R^2, ρ, and a20-index during training, validation and testing of models. Table 2 presents the results of the optimum LSTM model developed for SMC modeling. The proximity of statistical indices between train and test data shows no overfitting in the modeling.

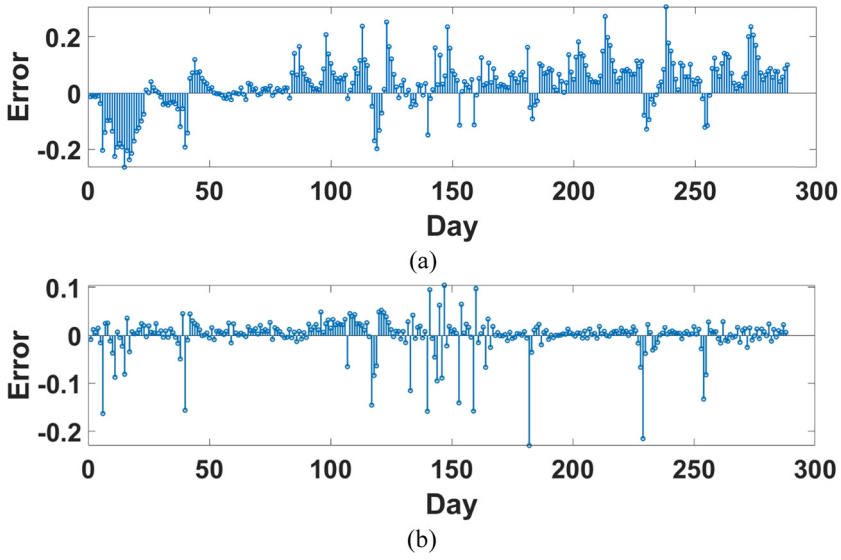

Figure 22: Error evaluation for test data by (a) hand tuning and (b) Bayesian optimization.

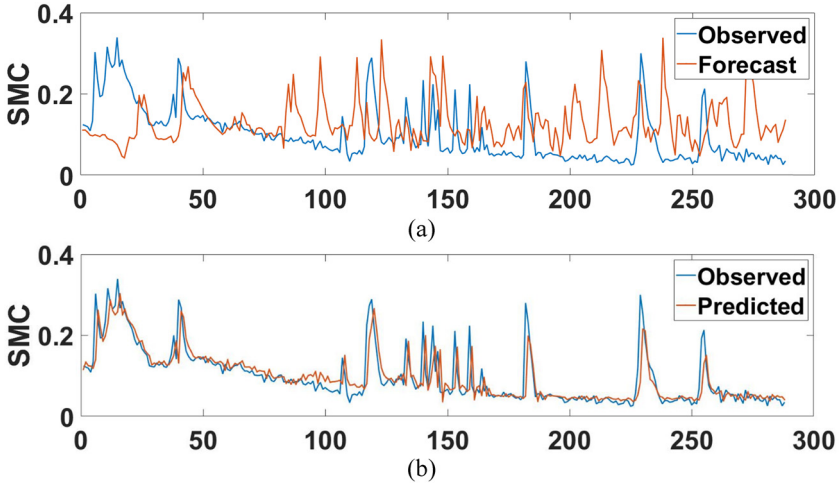

Figure 23: Scatter plot of the target (observed, in blue) versus modeled (outputs, in brown) for test data by (a) hand tuning and (b) Bayesian optimization.

The evaluation of the LSTM neural network function is both resource-intensive and time-consuming. To mitigate this, an acquisition function is employed to efficiently explore the search space and limit the evaluation of the expensive function to specific regions within that space. The objective function, represented as a GP model, trains the model by evaluating the objective function. The Gaussian function is chosen for analytical tracking as it can induce a posterior distribution over the objective function.

Figure 21 illustrates the evaluation of the function, demonstrating how the minimum objective is achieved through iterations. In each iteration, samples generated from the acquisition function are evaluated by the objective function. The evaluated samples are then incorporated into the data to update its posterior distribution using Bayes' theorem. This process optimizes the hyperparameters across the layers and performs global optimization on the validation set, leading to reduced time consumption and improved model performance.

It is evident that there exists a tradeoff between the time required for model training and hyperparameter tuning. This tradeoff proves valuable when the trained model is applied to diverse domains beyond its initial training context.

For train data, mean absolute error (MSE) = 0.0016691, RMSE = 0.040854, NRMSE = 0.29302, for test data, MSE = 0.0016513, RMSE = 0.040636, NRMSE = 0.41276 and for all data, MSE = 0.0016717, RMSE = 0.040886, and NRMSE = 0.31158. Figure 22 shows the error evaluation for test data by hand tuning and Bayesian optimization. The process of reproducing hand tuning is exceptionally challenging due to its reliance on the trial and error approach, making it a formidable procedure to replicate.

For train data, mean = 0.00021974, StD = 0.040949, for test data, mean = −0.0074623, StD = 0.040085 and for all data, mean = −0.0013178, StD = 0.04088. In all cases, the error mean is around zero. For train data, R^2 = 0.54427, for test data, R^2 = 0.43659, and for all data R^2 = 0.55004. Increasing the number of data can lead to improved regression results. Figure 23 displays the Scatter plot of the target (observed) and modeled (outputs) for SMC test data by hand tuning and Bayesian optimization. Rank correlation for train, test, and all data equals 0.829, 0.857, and 0.849, respectively.

5 Conclusions

This article presents a multi-stage process for estimating near-surface soil moisture around a PBO (Plate Boundary Observatory) station named P038 from 2017 to 2020. The required data are obtained from the special archive section in UNAVCO. The Chew et al. (2014) algorithm is utilized to model and remove the vegetation effects in the estimated soil moisture time series. Since the focus of the research is on soil moisture, only L2C data are used. The utilization of SNR data from the new L2C signals is attributed to their higher data quality in comparison to both the legacy L1 and L2P signals. The maximum depth of penetration, up to 5 cm, corresponds to the frequency of the L2C signal.

The initial step involves generating SNR data from RINEX files. Through several stages, the phase for each satellite on each day is obtained. The vegetation effect is then mitigated, and the resulting data are converted into VWC. According to the results, the VWC level has increased from 8.88 to 12.49% between 2017 and 2020.

According to the research results, GNSS-IR can be considered as another remote sensing method for monitoring soil moisture continuously and of course on a local scale, which can be used in different weather conditions such as rainy and foggy conditions and in different lighting conditions such as day and night are applicable.

Furthermore, soil moisture prediction is conducted for multiple days in advance, which can assist farmers in preparing irrigation schedules more efficiently. To achieve the best results, the hyperparameters of the LSTM model are tuned using the Bayesian optimization algorithm. The accuracy of the network is evaluated using various metrics on the train, test, and all data. The RMSE, NRMSE, R^2, ρ, and a20-index of test data are obtained as 0.041, 0.413, 0.436, 0.857, and 0.727, respectively.

GNSS-IR has certain limitations. First, it provides coarse-scale measurements with limited spatial resolutions, making it difficult to capture fine-scale variability in soil moisture within a specific area. Vegetation can also pose challenges as it obstructs direct signals from satellites, leading to less accurate soil moisture data. Atmospheric conditions like rain, snow, and fog can attenuate GNSS signals, introducing errors in the estimation. Moreover, surface roughness can affect the reflection of GNSS signals, causing uncertainties in soil moisture measurements.

Future research in GNSS-IR should focus on addressing the limitations mentioned earlier. This can be achieved through various approaches, such as multi-sensor fusion, data assimilation techniques, improved signal processing, and ground truth validation. Models and correction methods should be developed to account for surface roughness and minimize errors. Increasing the number of samples and expanding the study area can enhance the robustness and generalizability of findings. Additionally, conducting experiments on a larger scale will facilitate the practical application of the research in real-world agricultural irrigation settings. Considerations regarding data acquisition, resource allocation, and methodology design will be explored to ensure reliable and accurate results.

Author contributions: Only the authors mentioned in the paper contributed to this paper and there were no other contributors. All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Patricia Daneghian, and Asghar Rastbood. The first draft of the manuscript was written by Patricia Daneghian and Asghar Rastbood commented on previous versions of the manuscript. All authors read and approved the final manuscript. The authors have not received funding or financial assistance from any organization.

Conflict of interest: Authors state no conflict of interest.

Data availability statement and supplementary materials: Datasets related to this article can be found in the "special" archive section on UNAVCO's website at https://data.unavco.org/archive/gnss/products/reflectometry/. Additionally, you can access our GitHub account at https://github.com/arastbood2023/soil-moisture-modelling/. It should be noted that all figures and calculations related to GNSS-IR for soil moisture content estimation were performed using an open-source package called gnssrefl code, which can be found at https://github.com/kristinemlarson/gnssrefl. The processing for LSTM and Bayesian optimization for hyperparameter tuning calculations was done using Optuna, a Python library for hyperparameter optimization, available at https://github.com/optuna/optuna. For further information about the P038 site, you can visit the following link: https://www.unavco.org/instrumentation/networks/status/nota/overview/P038.

References

Ahmad, S., A. Kalra, and H. Stephen, 2010. "Estimating soil moisture using remote sensing data: A machine learning approach." *Advances in Water Resources* 33(1), 69–80.

Ahmad, S. and S. P. Simonovic, 2005. "An artificial neural network model for generating hydrograph from hydro-meteorological parameters." *Journal of Hydrology* 315(1–4), 236–51.

Ahmed, A. M., R. C. Deo, A. Ghahramani, N. Raj, Q. Feng, Z. Yin, and L. Yang. 2021. "LSTM integrated with Boruta-random forest optimiser for soil moisture estimation under RCP4. 5 and RCP8. 5 global warming scenarios." *Stochastic Environmental Research and Risk Assessment* 35(9), 1851–81.

Alexakis, D. D., F. D. K. Mexis, A. E. K. Vozinaki, I. N. Daliakopoulos, and I. K. Tsanis. 2017. "Soil moisture content estimation based on Sentinel-1 and auxiliary earth observation products. A hydrological approach." *Sensors* 17(6), 1455.

Apostolopoulou, M., P. G. Asteris, D. J. Armaghani, M. G. Douvika, P. B. Lourenço, L. Cavaleri, A. Bakolas, and A. Moropoulou. 2020. "Mapping and holistic design of natural hydraulic lime mortars." *Cement and Concrete Research* 136, 106167.

Armaghani, D. J., and P. G. Asteris. 2021. "A comparative study of ANN and ANFIS models for the prediction of cement-based mortar materials compressive strength." *Neural Computing and Applications* 33(9), 4501–32.

Armaghani, D. J., G. D. Hatzigeorgiou, C. Karamani, A. Skentou, I. Zoumpoulaki, and P. G. Asteris. 2019. "Soft computing-based techniques for concrete beams shear strength." *Procedia Structural Integrity* 17, 924–33.

Asteris, P. G., A. Mamou, M. Hajihassani, M. Hasanipanah, M. Koopialipoor, T.-T. Le, N. Kardani, and D. J. Armaghani. 2021. "Soft computing based closed form equations correlating L and N-type Schmidt hammer rebound numbers of rocks." *Transportation Geotechnics* 29, 100588.

Asteris, P. G., and T. A. Nguyen. 2022. “Prediction of shear strength of corrosion reinforced concrete beams using Artificial Neural Network.” *Journal of Science and Transport Technology* 2, 1–12.

Bergstra, J., R. Bardenet, Y. Bengio, and B. Kégl. 2011. “Algorithms for hyper-parameter optimization.” *Advances in Neural Information Processing Systems* 24, 2546–54.

Bergstra, J., and Y. Bengio. 2012. “Random search for hyper-parameter optimization.” *Journal of Machine Learning Research* 13, 2.

Bergstra, J., D. Yamins, and D. D. Cox. 2013. “Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures.” *In Proceedings of the 30th International Conference on Machine learning, Atlanta, Georgia, USA*. doi: 10.5555/3042817.3042832.

Brochu, E., V. M. Cora, and N. De Freitas. 2010. “A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning.” *arXiv preprint arXiv:1012.2599*.

Chai, S. S., J. P. Walker, O. Makarynskyy, M. Kuhn, B. Veenendaal, and G. West. 2009. “Use of soil moisture variability in artificial neural network retrieval of soil moisture.” *Remote Sensing* 2(1), 166–90.

Chen, Q., W. Jiang, X. Meng, P. Jiang, K. Wang, Y. Xie, and J. Ye. 2018. “Vertical deformation monitoring of the suspension bridge tower using GNSS: A case study of the forth road bridge in the UK.” *Remote Sensing* 10(3), 364.

Chew, C. C., E. E. Small, K. M. Larson, and V. U. Zavorotny. 2013. “Effects of near-surface soil moisture on GPS SNR data: Development of a retrieval algorithm for soil moisture.” *IEEE Transactions on Geoscience and Remote Sensing* 52(1), 537–43.

Chew, C. C., E. E. Small, K. M. Larson, and V. U. Zavorotny. 2014. “Vegetation sensing using GPS-interferometric reflectometry: Theoretical effects of canopy parameters on signal-to-noise ratio data.” *IEEE Transactions on Geoscience and Remote Sensing* 53(5), 2755–64.

Dabboor, M., G. Atteia, S. Meshoul, and W. Alayed. 2023. “Deep learning-based framework for soil moisture content retrieval of bare soil from satellite data.” *Remote Sensing* 15(7), 1916.

Dewancker, I., M. McCourt, and S. Clark. 2015. *Bayesian optimization primer*. https://app.sigopt.com/static/pdf/SigOpt_Bayesian_Optimization_Primer.pdf.

Dobson, M. C., F. T. Ulaby, M. T. Hallikainen, and M. A. El-Rayes. 1985 “Microwave dielectric behavior of wet soil-Part II: Dielectric mixing models.” *IEEE Transactions on Geoscience and Remote Sensing* 1, 35–46.

Feurer, M., A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter. 2019. *Auto-sklearn: efficient and robust automated machine learning, part of the springer series on challenges in machine learning book series (SSCML)*. Berlin, Germany: Spinger. Vol. 10, pp. 973–8.

Gers, F. A., J. Schmidhuber, and F. Cummins. 2000. “Learning to forget: Continual prediction with LSTM.” *Neural Computation* 12(10), 2451–71.

Greff, K., R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber. 2017. “LSTM: A Search Space Odyssey.” *IEEE Transactions on Neural Networks and Learning Systems* 28(10), 2222–32. doi: 10.1109/TNNLS.2016.2582924.

Gupta, D. K., R. Prasad, P. Kumar, and A. K. Vishwakarma. 2017. “Soil moisture retrieval using ground based bistatic scatterometer data at X-band.” *Advances in Space Research* 59(4), 996–1007.

Hassan-Esfahani, L., A. Torres-Rua, A. Jensen, and M. McKee. 2015. “Assessment of surface soil moisture using high-resolution multi-spectral imagery and artificial neural networks.” *Remote Sensing* 7(3), 2627–46.

Hochreiter, S. and J. Schmidhuber. 1997. LSTM can solve hard long time lag problems. *Advances in Neural Information Processing Systems 9*. Denver, CO, USA: NIPS, p. 473–9.

Hoffman, M., B. Shahriari, and N. Freitas. 2014. “On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning.” *Artificial Intelligence and Statistics*. Reykjavic, Iceland: PMLR, pp. 365–74.

Kramer, O., D. E. Ciaurri, and S. Koziel. 2011. “Derivative-free optimization.” *In Computational optimization, methods and algorithms*, Heidelberg, Berlin: Springer, pp. 61–83. doi: doi: 10.1007/978-3-642-20859-1_4.

Kumar, P., R. Prasad, A. Choudhary, D. Gupta, V. Mishra, A. Vishwakarma, A. Singh, and P. Srivastava. 2019. “Comprehensive evaluation of soil moisture retrieval models under different crop cover types using C-band synthetic aperture radar data.” *Geocarto International* 34(9), 1022–41.

Larson, K. M., J. J. Braun, E. E. Small, V. U. Zavorotny, E. D. Gutmann, and A. L. Bilich. 2009. “GPS multipath and its relation to near-surface soil moisture content.” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 3(1), 91–9.

Larson, K. M., and F. G. Nievinski. 2013. “GPS snow sensing: results from the EarthScope Plate Boundary Observatory.” *GPS solutions* 17, 41–52.

Larson, K. M., E. E. Small, E. Gutmann, A. Bilich, P. Axelrad, and J. Braun. 2008. “Using GPS multipath to measure soil moisture fluctuations: Initial results.” *GPS Solutions* 12(3), 173–7.

Li, X., G. Dick, C. Lu, M. Ge, T. Nilsson, T. Ning, J. Wickert, and H. Schuh. 2015. “Multi-GNSS meteorology: Real-time retrieving of atmospheric water vapor from BeiDou, Galileo, GLONASS, and GPS observations.” *IEEE Transactions on Geoscience and Remote Sensing* 53(12), 6385–93.

Luo, X., S. Yan, J. Shan, H. Yan, and H. Wang. 2016. “Using the BDS-R signal for soil moisture estimation.” *China Satellite Navigation Conference (CSNC) Proceedings: Volume I*. Springer, pp. 175–85.

Mao, K., J. Wang, and M. Zhang. 2009. “The study of soil moisture retrieval from GNSS_R signals based on AIEM model and experiment data.” *High Tech Lett* 3, 295–301.

Marchant, R., and F. Ramos. 2012. “Bayesian optimisation for intelligent environmental monitoring.” *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 2242–9.

Martín, A., S. Ibáñez, C. Baixauli, S. Blanc, and A. B. Anquela. 2020. “Multi-constellation GNSS interferometric reflectometry with mass-market sensors as a solution for soil moisture monitoring.” *Hydrology and Earth System Sciences* 24(7), 3573–82.

Paloscia, S., S. Pettinato, E. Santi, C. Notarnicola, L. Pasolli, and A. Reppucci. 2013. “Soil moisture mapping using Sentinel-1 images: Algorithm and preliminary validation.” *Remote Sensing of Environment* 134, 234–48.

Shahriari, B., K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. 2015. “Taking the human out of the loop: A review of Bayesian optimization.” *Proceedings of the IEEE* 104(1), 148–75.

Snoek, J., H. Larochelle, and R. P. Adams. 2012. “Practical bayesian optimization of machine learning algorithms.” *Advances in Neural Information Processing Systems* 25, 2951–9.

Twarakavi, N. K., D. Misra, and S. Bandopadhyay. 2006. "Prediction of arsenic in bedrock derived stream sediments at a gold mine site under conditions of sparse data." *Natural Resources Research* 15, 15–26.

Victoria, A. H. and G. Maragatham. 2021. "Automatic tuning of hyper-parameters using Bayesian optimization." *Evolving Systems* 12, 217–23.

Wan, W., K. M. Larson, E. E. Small, C. C. Chew, and J. J. Braun. 2015. "Using geodetic GPS receivers to measure vegetation water content." *GPS Solutions* 19, 237–48.

Zhang, S., J. Peng, C. Zhang, J. Zhang, L. Wang, T. Wang, and Liu Q. 2021. "GiRsnow: An open-source software for snow depth retrievals using GNSS interferometric reflectometry." *GPS Solutions* 25, 1–8.