



HAL
open science

Machine Learning-Based Lightning Localization Algorithm Using Lightning-Induced Voltages on Transmission Lines

Hamidreza Karami, Amirhossein Mostajabi, Mohammad Azadifar, Marcos Rubinstein, Chijie Zhuang, Farhad Rachidi

► **To cite this version:**

Hamidreza Karami, Amirhossein Mostajabi, Mohammad Azadifar, Marcos Rubinstein, Chijie Zhuang, et al.. Machine Learning-Based Lightning Localization Algorithm Using Lightning-Induced Voltages on Transmission Lines. IEEE Transactions on Electromagnetic Compatibility, 2020, 62 (6), pp.2512-2519. 10.1109/TEMC.2020.2978429 . hal-03589923

HAL Id: hal-03589923

<https://hal.science/hal-03589923>

Submitted on 26 Feb 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Machine Learning Based Lightning Localization Algorithm Using Lightning-Induced Voltages on Transmission Lines

Hamidreza Karami, Amirhossein Mostajabi, Mohammad Azadifar, *Member, IEEE*, Marcos Rubinstein, *Fellow, IEEE*, Chijie Zhuang, *Member, IEEE* and Farhad Rachidi, *Fellow, IEEE*

Abstract—In this study, we present a Machine Learning based method to locate lightning flashes using calculations of lightning-induced voltages on a transmission line. The proposed approach takes advantage of the preinstalled voltage measurement systems on power transmission lines to get the data. Hence, it does not require the installation of additional sensors such as ELF, VLF, or VHF. The proposed model is shown to yield reasonable accuracy in estimating 2D geolocations for lightning strike points in a grid of 10x10 km². The median location error obtained is less than 90 m when the sensors are 2 km away from each other. The algorithm is shown to be flexible when it comes to choosing the distance between the two voltage sensors. Furthermore, the changes in the risetime of the return stroke currents had negligible effect on the geolocation accuracies.

Keywords— *Lightning Localization; Machine Learning; Transients on Transmission Lines; Gradient Boosting Algorithms*

I. INTRODUCTION

Knowing the exact geolocation of a lightning strike is important in a wide range of research and application domains, including geophysical research, lightning warning, aviation/air traffic, weather services, insurance claims, power transmission and distribution, etc. For example, some lightning warning systems rely on such data to indicate approaching thunderstorms [1], [2] and thus to prevent catastrophic effects of lightning strikes to critical infrastructure, sensitive equipment or systems, and outdoor facilities. Even in high-energy atmospheric physics, lightning location data are used to seek the lightning flashes associated with Terrestrial Gamma-ray Flashes (TGFs) seen from space [3].

Given its fundamental role in many aspects of lightning studies, appreciable attention has been given to accurate lightning localization. The most widely used lightning location techniques are the time-of-arrival method (ToA) and the magnetic direction finding method [4]–[6]. In [7]–[10], a method was proposed to locate the lightning strike point based on the Electromagnetic Time Reversal (EMTR) theory. In this technique, first the electric field waveforms of the lightning strike are measured by multiple sensors (this is the so-called

“forward” or direct-time phase). Second, a time-reversed version of these waveforms is back-injected from the sensor points into the location domain, also called the computational domain since this phase is carried out by simulation (the so-called “backpropagation” or reverse-time phase). Finally, a criterion is used to determine the location of the lightning strike point. It is reported in [7] that at least 3 sensors are needed to locate the lightning strike using EMTR. Recently, Qin et al. [11] introduced a GPU-based algorithm to increase the performance of lightning geolocation networks. The algorithm has been effectively applied in, respectively, a six-station and a five-station networks for two-dimensional (2-D) and 3-D geolocation estimation of lightning flashes. All the above-mentioned approaches require the installation of appropriate sensor modules (ELF, VLF, or VHF).

A methodology based on the difference of the time of arrival of the induced voltages on a transmission line was presented in [12] to obtain the coordinates of the lightning strike. However, in the derivation, the direct field from the lightning strike was ignored, which is not an acceptable assumption in most cases.

This paper presents a new machine learning based lightning localization algorithm in 2D that utilizes data from the preinstalled voltage measurement systems on power transmission lines. The algorithm requires at least two sensors to operate. It should be noted that using only two sensors can lead to an ambiguity in the lightning location on both sides of the transmission line. The ambiguity can be removed either by considering another voltage or current sensor on another transmission line, or by considering the terrain topography or other objects in the environment that could remove the symmetry of the problem.

Machine learning algorithms can give computers the ability to learn a skill (such as the prediction of the geographic coordinates of a passive or active object) from sets of archived data, with the final goal of applying the skill to new cases. They do that by automatically extracting an unknown underlying mapping function from the inputs to the outputs. In supervised learning, they are designed to learn a target function that best maps input features to the outputs given a dataset with an already known output (the so-called training set). They would use the learned function in future to estimate the outputs for new

unseen examples of the input features (the so-called testing set). Recently, machine learning has been also shown to be a useful method for source localization. For example, Huang et al. [13] applied deep neural networks to acoustic source localization in shallow water environments. Vera-Diaz et al. [14] used deep learning to directly estimate the three-dimensional position of a single acoustic source using raw data from microphone arrays. Regarding the application of Machine Learning to lightning localization, we present in this paper a proof of concept on how machine learning algorithms could be used to locate lightning flashes by looking at their associated lightning-induced voltages on transmission lines. To achieve this, a machine-learning-based model is trained to estimate the 2D geolocation of a lightning strike point, given the real-time measured values of lightning-induced voltages measured by two sensors on the transmission line.

The paper is organized as follows. Section II explains different steps in the proposed algorithm including building the database, machine learning model selection and generation. A discussion is then given in Section III on how the generated model is trained and tested to estimate the lightning strike point using numerical simulation results. Finally, conclusions and a final discussion are given in Section IV.

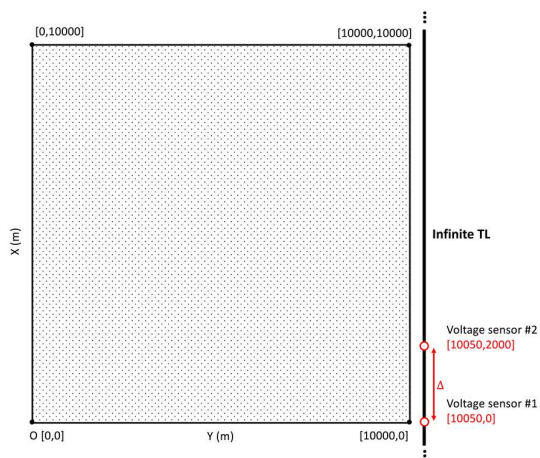


Fig. 1. Sketch diagram of the detection region. The infinite transmission line is located 50 m beside the detection region. The two voltage sensors are located on the line and 2 km away from each other ($\Delta=2$ km). The coordinates of the sensor positions are annotated in red.

II. METHODOLOGY

A. Numerical Simulation and Data Acquisition

In this section, we aim to train a machine learning model to estimate the geolocation of an electromagnetic source, given the data of the lightning-induced voltages obtained by two sensors located on a power transmission line. The geometry of the problem, shown in Fig. 1, consists of an infinite transmission line with two sensors that capture the lightning induced voltages on it. We defined 200 uniformly random positions for the

source within the presented geometry in Fig. 1. To build the required database for training and testing procedures, the lightning-induced voltages from lightning striking at each of these source positions are calculated, without loss of generality, using Rusck's formula [15], which is written as follows,

$$v(x, t) = \frac{Z_0 I_0 \beta}{4\pi} \cdot \left[\frac{\gamma_-}{(d^2 + \beta^2 \gamma_-^2)} \left(1 + \left(r + \frac{\beta^2 \gamma_-^2}{\sqrt{(\beta ct)^2 + \rho^2 / \delta}} \right) \right) + \frac{\gamma_+}{(d^2 + \beta^2 \gamma_+^2)} \left(1 + \left(r + \frac{\beta^2 \gamma_+^2}{\sqrt{(\beta ct)^2 + \rho^2 / \delta}} \right) \right) \right] \quad (1)$$

where,

$$\gamma_- = (ct - x), \gamma_+ = (ct + x), \delta = \frac{1}{1 - \beta^2}, \text{ and } \rho = x^2 + d^2.$$

The amplitude of the return stroke current is I_0 and its velocity is equal to βc , in which c is the velocity of light in free space and β , set to 0.4 in this paper, is the ratio of the return-stroke speed to the speed of light. t denotes the time and it should be greater than $\sqrt{x^2 + d^2}/c$. d is the horizontal distance between the lightning channel and the transmission line and x is the distance along the transmission line (Fig. 1). According to [16], we chose the peak channel base current to be $I_0 = 12$ kA, consistent with the median of subsequent return stroke current peak. Z_0 is the characteristic impedance of free space. In Rusck's formula, the lightning-induced voltages are calculated for a lossless, single-wire transmission line above a perfectly-conducting ground. The excitation source (i.e., the lightning flash) is a step current ascending along the lightning strike channel.

To calculate the induced-voltages at the nearest point of the transmission line ($x = 0$) to the lightning stroke, the following formula is used,

$$v(0, t) = \frac{Z_0 I_0 h}{4\pi d^2} \cdot \frac{2\beta ct}{1 + \left(\frac{\beta ct}{d}\right)^2} \cdot \left(1 + \beta^2 \left(\frac{ct/d}{\sqrt{1 + \beta^2((ct/d)^2 - 1)}} \right) \right), \quad (2)$$

valid only for $t \geq d/c$. All parameters have been defined before.

It should be noted that other, more accurate induced-voltage calculation approaches, either based on transmission line theory (e.g., [17]) or full-wave simulations (e.g., [18]) can be easily substituted in the proposed method. Using (1), we calculated the lightning-induced voltages in the time domain at the locations of the two sensors. For each of the sensors, the calculation of the signal was continued until the amplitude of the voltage reached 10^{-7} of the maximum recorded value by

that sensor. Fig. 2 shows a zoomed view of two sample induced voltages recorded by the two sensors.

1) Data Preprocessing

Once all random positions were considered, we ended up with 2 x 200 different waveforms of the ‘measured’ transient voltages with each of them having a different number of samples. We performed two preprocessing steps before feeding the ‘measured’ signals into the model:

i. Since the model will treat these samples as the features for its learning process, the differing number of samples would make the number of features be different from one observation to another. However, for the machine learning model, the number of features in the training set must be the same as the number of features in the test set. Therefore, a preprocessing step was required to make all transient signals be of the same length. To do that, we extended the length of the shorter signals by appending zeros to the end of each one of the shorter signals until it reached the maximum length, N_{\max} , among the recorded signals for sensors 1 and 2.

ii. The second preprocessing stage relates to how the data from the two sensors were merged before being used as the input features. We tried several linear and non-linear combinations using both, original and time-reversed signals. After several tries, the linear combination in (3) was seen to yield significantly higher accuracy:

$$V(t) = V_1(t) - V_2(T - t) \quad (3)$$

where $V(t)$ is the new combined signal, $V_1(t)$ is the signal measured by sensor #1 after the first preprocessing step was applied (i.e. with the length of the signal extended to N_{\max}), T is the time window, and $V_2(T - t)$ is the time-reversed version of the signal measured by sensor #2 after the first preprocessing step was applied.

After the above preprocessing steps, we formed a tabular database with each row containing the values of $V(t)$ for each of the 200 iterations mentioned above. In each row, the N_{\max} samples of $V(t)$ were used as the predictors and the x and y coordinates of the source were used as the response.

B. Selection of the Machine Learning Model

Once the database is formed, a machine learning algorithm needs to be employed to identify regularities between predictors and responses using a portion of the data which, as mentioned above, is called the training set. The trained model can then use the explored correlations to predict the response for new cases (testing set). A model search process was performed to choose the most appropriate machine learning regression model. The model search process repeatedly looks for the best-fit model through (i) several regression types including regression trees, support vector machines, and gaussian process regression models, and (ii) different ensemble methods such as bagging and boosting. The results indicated that the best performance

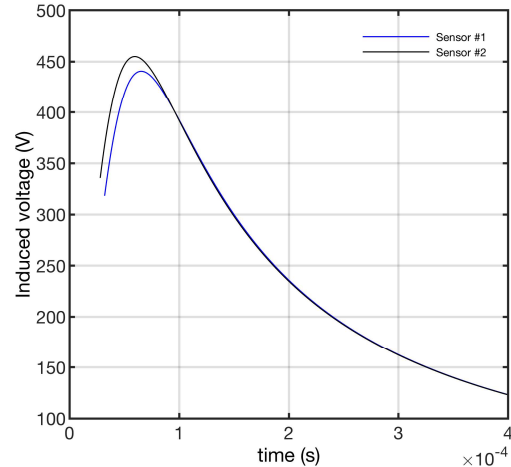


Fig. 2. A zoomed view of a sample of the induced voltages measured by the two sensors. The 2D coordinates of the lightning strike point is [3722 m, 6786 m] and the excitation source is a step current ascending along the lightning strike channel. The voltages are calculated using the Rusck’s formula.

would be achieved using the XGBoost algorithm. XGBoost stands for “Extreme Gradient Boosting” and it is a variant of the gradient boosting machine which uses a more regularized model formalization to control overfitting [19]. More information on the XGBoost model description and generation is given in the following section.

C. Model generation

In this study, we generated an ensemble learner out of individual classification trees using a scalable tree boosting system. Ensemble learners use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone called weak learners [20], [21]. A weak learner is an algorithm that generates classifiers that can merely do better than random guessing. What follows briefly describes the framework for the ensemble learning used in this study.

1) Preparing the weak learners

In this study, we used decision trees as the weak learners. Decision trees predict a response following the decisions in the tree from the root node down to the leaf nodes where the responses are. The flow of data points is split at each node based on the condition at each internal node. Each data point flows to one of the leaves following the direction on each node. When a data point reaches a leaf, a value is assigned to it as the prediction score. The predictive algorithm then combines the prediction scores that each data point gains from the ensemble members to generate the output.

2) Ensemble aggregation method

Boosting is a machine learning ensemble algorithm that is based on the idea that a weak learner can be turned into a strong

learner. Most boosting algorithms consist of iteratively learning weak learners and adding them to a final strong learner. At each iteration, the algorithm attempts to construct a new model that corrects the errors of its predecessor. Hence, the next weak learner will learn from an updated version of residual errors.

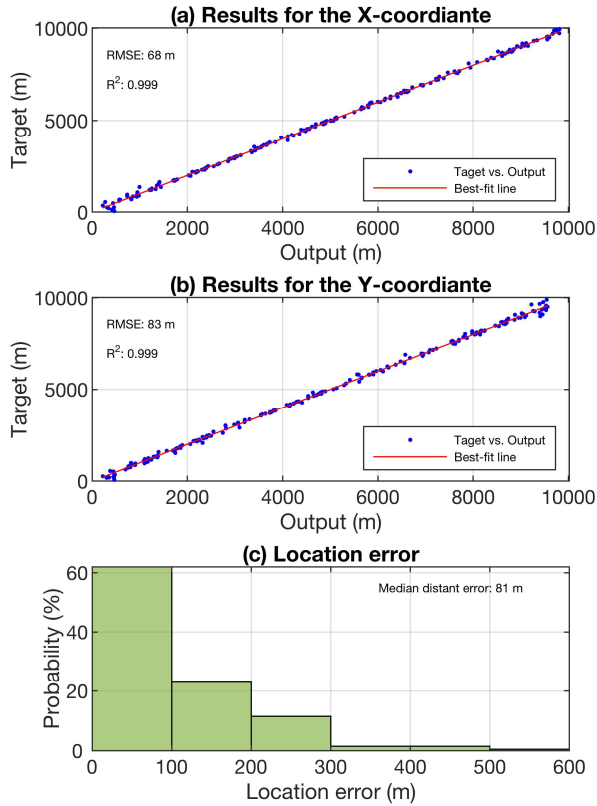


Fig. 3. Model evaluation results for (a) the x-coordiante and (b) the y-coordiante. (c) is the histogram of the location error considering 200 randomly selected lightning strike points inside.

The XGBoost algorithm is called gradient boosting since the objective function is optimized using the gradient descent algorithm before each new model is added. The objective function consists of two terms: The loss function, which is put as a measure of the predictive power, and the regularization factor, which controls the complexity of the model which helps to avoid overfitting. At each iteration, the algorithm needs to solve two key problems: (i) How to define the structure of the next weak learner (decision tree) in the ensemble so that it improves the prediction along the gradient, and (ii) how to assign the prediction scores to the leaves. The algorithm uses gradient descent to solve these two problems. To build a tree, the algorithm greedily enumerates the features and finds the best splitting point by calculating the split gains. After each split, it assigns the weight to the two new leaves grown on the tree. This process continues repeatedly until the maximum depth is reached. The algorithm then starts pruning the tree

backwards and removes nodes with a negative gain.

More information about the XGBoost algorithm, including the definition and calculation of the loss function, regularization function, and split gain can be found in Chen and Guestrin [19] and Chen and He [22].

III. EVALUATION OF THE MACHINE LEARNING MODEL

In this study, the predictive ML model was evaluated using a 5-fold cross-validation described in what follows. First, the dataset was shuffled and split into five different groups. As a result, each observation in the dataset was assigned to an individual group and remained there for the duration of the training and testing process. Each unique group was held out from the dataset as the test set and the remaining four groups were used as the training set. The model was then fitted on the training set and evaluated on the test set. The model estimation results on the test set were retained and the trained model was discarded. The process was repeated until each individual group had been taken once as the test set. The model outputs were combined over the rounds. This splitting method helps to eliminate the leakage of correlated samples from the training set into the test set. Moreover, it avoids overfitting to a specific testing set since each one of the observations is considered as part of the testing set in one of the five rounds. Once the five rounds of training and testing were done, the model's prediction skill was evaluated by comparing the outputs with the target values for the x - and y -coordinates. The results presented in Fig. 3a and Fig. 3b, are the scatter plots of the target versus output values where the blue dots correspond to the 200 observations in the database. The best-fit lines are calculated using the least-squares regression method. The very high values of the coefficient of determination (R^2) indicate that a high proportion

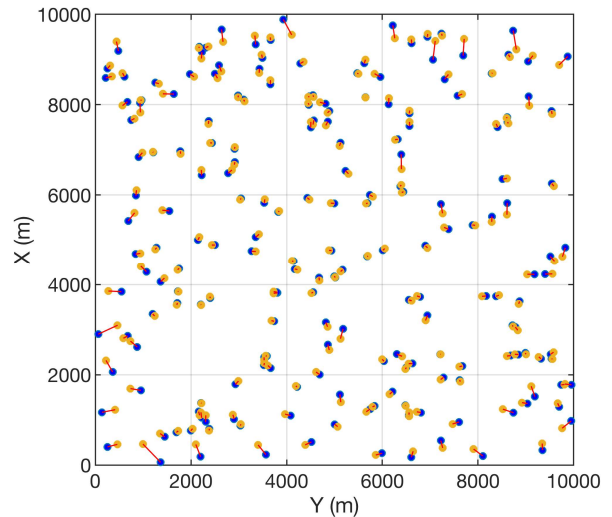


Fig. 4. Scatter plot of the target (blue dots) versus estimated (yellow dots) 2D geolocations for the N=200 guest lightning strike points using the proposed ML based approach.

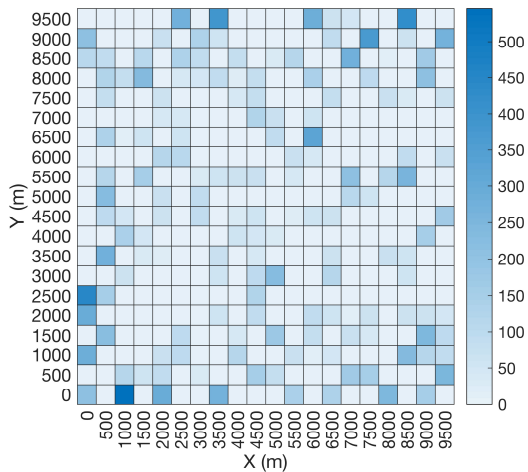


Fig. 5. Average location error presented as heatmap chart inside the detection region. The colormap represent the location errors in m. The (0,0) point corresponds to the coordinate center (O) shown in Fig. 1. The x and y labels for each of the cells are the coordinates of the left-bottom corner of the grid cell.

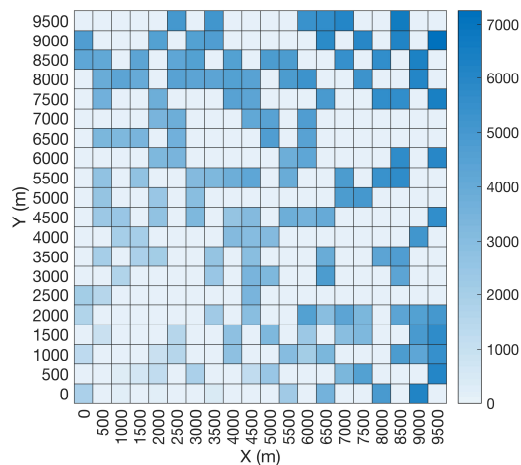


Fig. 6. As of Fig. 5 when only the data from sensor #1 is used as the input of the ML model.

of the variance in the data are explained by the fitting line. Looking at the results in Fig. 3a and Fig. 3b, the Root Mean Squared Errors (RMSE) are 68 m and 83 m for the x and y coordinates, respectively. These low error values mean that by just looking at the lightning-induced voltage waveforms measured by the two sensors on the transmission line, the model was able to accurately predict the location of the nearby lightning along the x and y axes. Finally, the relative probability of the location errors between the ground truth 2D source position and the estimated one by the model for each of the 200 considered source positions is given in Fig. 3c. The height of the bar in each bin is the relative number of observations that falls into that bin. According to the results, in more than 60% of the cases, the model was able to predict the source with less than 100 m location error.

Fig. 4 is the scatter plot of the true (blue dots) vs. estimated (yellow dots) lightning locations for the 200 studied samples. In order to visualize the location error density based on the position relative to the transmission line, the heatmap of the location errors is shown in Fig. 5. The figure shows, for each grid cell, the average of the location errors estimated by the model for the samples that fell into that grid. The colormap is the location error in meters. The figure shows that the maximum errors are attributed to the grid cells that are near the border of the detection region.

A. One-sensor Analysis

In this section, we investigate how eliminating one of the two sensors would affect the location accuracy. To do that, we followed the same procedure as the one stated in Section II, this time replacing (3) with the following equation:

$$V(t) = V_1(t) \tag{4}$$

where $V(t)$ is the signal used as the input for the ML model and $V_1(t)$ is the signal measured by the sensor #1 after the first preprocessing step (see II.A.1) is applied (i.e., with the length equal to N_{\max}). The guest lightning strike points and the procedures for training, testing, and for the evaluation of the model were the same as the ones used in the two-sensor analysis. The model performance results (Fig. 6 and Fig. 7) show a high median of near 4 km for the location errors. They also show that in 95% of the cases the model output was between 1000-7000 m away from the ground-truth geolocation. We also changed the input data used in (3) to $V_2(t)$ and the results were roughly the same. These results imply that one-sensor data were not enough for the model to make the estimation of the lightning strike point and at least two sensors need to be used.

B. The Impact of the Sensors' Positions

In this subsection, we examine the effect of the distance separating the two sensors on the performance of the system. In this Section, we investigate how the geolocation accuracy changes when the distance between the two sensors varies along the transmission line. To do that, we changed the distance between the two sensors from 2 km to 5 km, 8 km, and 10 km. We did that by moving sensor #2 from [100050, 2000] to [10050, 5000], [10050, 8000], and [10050, 10000] in the detection region shown in Fig. 1. For each of these new distances, we followed the same procedure described for the case of 2 km-distance in Section II using the same 200 guest locations of lightning strike points. Fig. 8 presents the cumulative probability and the Probability Density Function (PDF) of the location errors for the four different sensor positions. Looking at the results for the studied sensor positions, one can see that the median location error varies in a narrow

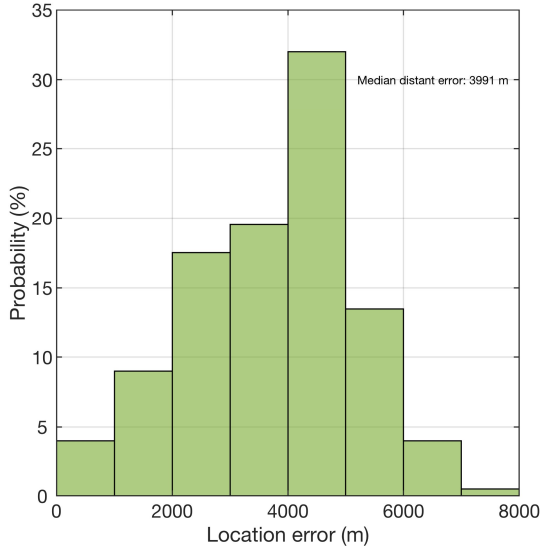


Fig. 7. As of Fig. 3c when only the data from sensor #1 is used as the input of the ML model.

range (between 65 m to 97 m) compared to the size of the detection region and the maximum location error is less than 600 m. Moreover, probability distribution functions are seen to have similar widths at the four sensors' positions. Given this, the proposed ML model seems to have low sensitivity to the distance between the deployed sensors. The reason comes back to the adaptability of the ML model to the change in the input data. In other words, once the sensors are moved to the new position, the ML model would then learn the new underlying relationships using the new input data and, hence, it would be able to deliver similar performance results.

C. Sensitivity to the Risetime of the Lightning Current

As stated in Section II.A, so far, the lightning current is modeled using a simple step function. To investigate the effect of the risetime on the performance of the model, we redid the analyses using a linearly rising ramp function as the excitation pulse (i.e. the lightning current). As a result, the transmission line (TL) model was used to represent the lightning return stroke, with a channel-base current expressed as follows

$$i(t) = \alpha t u(t) - \alpha(t - t_r)u(t - t_r) \quad (6)$$

where $u(t)$ and t_r are, respectively, the Heaviside function and the risetime of the current. α is the slope of the linearly rising current.

At each of the 200 guest locations, a random value for the risetime (t_r) was chosen and used to form the excitation source. The range for the rise time values was derived from the direct current measurements reported in [16]. The induced voltages at the two sensors (Fig. 1) were calculated using the following analytical formula [23]:

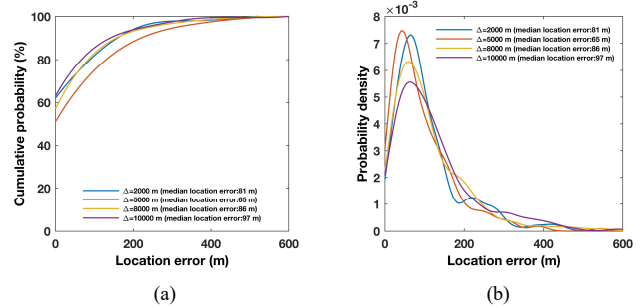


Fig. 8. (a) Cumulative probability and (b) probability density estimate of the location errors for four different sensors' positions. Δ is the distance between the two deployed voltage sensors on the transmission line (see Fig. 1).

$$v(x, t) = \alpha \frac{30h}{\beta c} \text{Log} \left(1 + \left(\frac{\beta}{t_d} \frac{t^2 - t_0^2}{t + \beta t'} \right)^2 + 2\beta \text{Log} \left(\frac{t + t'}{t_0 + t_0/\beta} \right) \right) \quad (5)$$

where,

$$t_d = \frac{d}{c}, t_0 = \sqrt{t_x^2 + t_y^2 + t_z^2}, t_x = \frac{x}{c}, t_z = \frac{h}{c}, t' = \sqrt{t^2 + \tau t_0^2}, \tau = (1 - \beta^2)/\beta^2.$$

Doing this, we investigated how the ML model performs when the lightning currents used during the training and testing procedures can have different risetimes, which is what happens in the real case.

We also moved sensor #2 along the transmission line and at the exact locations mentioned in Section III.B to verify any dependencies of the results on the sensors' positions. The obtained results (Fig. 9) reveal very similar location accuracies with the ones achieved in the case of a step function excitation source (Fig. 8). The results show that the ML model still yields excellent performance even when the excitation sources have random values for the risetime.

IV. CONCLUSIONS

A Machine Learning (ML) based method was proposed to locate lightning flashes using measurements of lightning-induced voltages on a transmission line. The algorithm builds up a database based on the solutions for lightning-induced voltages on transmission lines. It then uses part of the data to learn the underlying target function that best maps the inputs (measured lightning-induced voltages) to the outputs (geolocation of the lightning strike point). Once trained, the model can estimate the lightning strike point for cases that were not used in the training phase.

The model yields reasonable accuracy for 2D geolocations in a grid of 10x10 km². The median location error was observed to be 81 m and the model performance was observed to become much better when the lightning strike point is farther from the

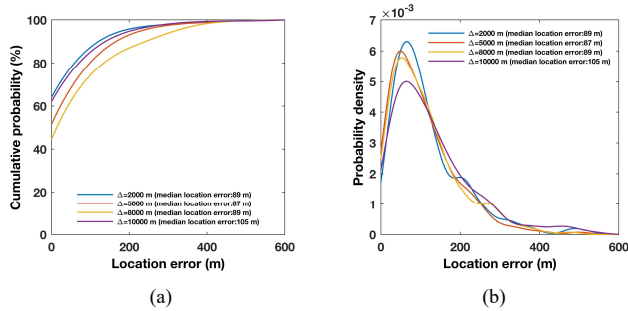


Fig. 9. As of Fig. 8 when the excitation source is changed to the linearly rising ramp function.

border of the detection region. To find the optimum number of required sensors, the data from each of the two sensors were separately fed into the model. It was seen that using data from only one of the sensors is not enough to reach reasonable source location estimations. Moreover, the sensor’s relative position along the transmission line was seen to have a negligible impact on the estimation results, with the model being able to deliver sufficient accuracy both when the sensors are moved closer or farther away from each other.

In this study, which is only intended to be a proof of concept, the conditions were highly idealized. The lightning return stroke current was modeled using a step function and linearly rising ramp function, the transmission line had an infinite length, was lossless and the ground was a perfectly conducting plane. Therefore, the model does not account for the effects of finite-length, lossy multiconductor transmission lines, the existence of the shield wire, a lossy ground, or the earth topography.

Research is in progress by the authors to investigate the performance of the model for finite length multi conductor transmission lines using advanced models based either on the transmission line theory or full-wave approaches.

ACKNOWLEDGMENT

This work was supported by Sichuan Energy and Internet Research Institute (Grant Nr. X), and also in part by the Swiss National Science Foundation (Project No. 200020_175594) and the European Union's Horizon 2020 research and innovation program under grant agreement No 737033-LLR.

REFERENCES

[1] M. J. Murphy, N. W. S. Demetriades, and K. L. Cummins, “Probabilistic early warning of cloud-to-ground lightning at an airport,” in *16th Conference on Probability and Statistics in the Atmospheric Sciences*, 2000, pp. 126–131.
 [2] A. Karagiannidis, K. Lagouvardos, and V. Kotroni, “The use of lightning data and Meteosat infrared imagery for the nowcasting of lightning activity,” *Atmos. Res.*, vol. 168, pp. 57–69, Feb. 2016.
 [3] D. M. Smith *et al.*, “The rarity of terrestrial gamma-ray flashes,” *Geophys.*

Res. Lett., vol. 38, no. 8, p. n/a-n/a, Apr. 2011.
 [4] V. A. Rakov, “Electromagnetic Methods of Lightning Detection,” *Surv. Geophys.*, vol. 34, no. 6, pp. 731–753, Nov. 2013.
 [5] K. L. Cummins, M. J. Murphy, E. A. Bardo, W. L. Hiscox, R. B. Pyle, and A. E. Pifer, “A Combined TOA/MDF Technology Upgrade of the U.S. National Lightning Detection Network,” *J. Geophys. Res. Atmos.*, vol. 103, no. D8, pp. 9035–9044, Apr. 1998.
 [6] K. L. Cummins and M. J. Murphy, “An Overview of Lightning Locating Systems: History, Techniques, and Data Uses, With an In-Depth Look at the U.S. NLDN,” *IEEE Trans. Electromagn. Compat.*, vol. 51, no. 3, pp. 499–518, Aug. 2009.
 [7] N. Mora, F. Rachidi, and M. Rubinstein, “Application of the time reversal of electromagnetic fields to locate lightning discharges,” *Atmos. Res.*, vol. 117, pp. 78–85, Nov. 2012.
 [8] G. Lugrin, N. M. Parra, F. Rachidi, M. Rubinstein, and G. Diendorfer, “On the Location of Lightning Discharges Using Time Reversal of Electromagnetic Fields,” *IEEE Trans. Electromagn. Compat.*, vol. 56, no. 1, pp. 149–158, Feb. 2014.
 [9] H. Karami, F. Rachidi, M. Rubinstein, “On Practical Implementation of Electromagnetic Time Reversal to Locate Lightning,” in *23rd International Lightning Detection Conference (ILDC)*, Tucson, Arizona, 2014.
 [10] G. Lugrin, N. Mora, F. Rachidi, M. Rubinstein, and G. Diendorfer, “On the use of the Time Reversal of Electromagnetic fields to locate lightning discharges,” in *2012 International Conference on Lightning Protection (ICLP)*, 2012, pp. 1–4.
 [11] Z. Qin, M. Chen, F. Lyu, ... S. C.-I. T., and U. 2019, “A GPU-Based Grid Traverse Algorithm for Accelerating Lightning Geolocation Process,” *IEEE Trans. Electromagn. Compat.*, 2019.
 [12] Z. Abdul-Malek, Aulia, N. Bashir, and Novizo, “Lightning Location and Mapping System Using Time Difference of Arrival (TDOA) Technique,” in *Practical Applications and Solutions Using LabVIEW Software*, InTech, 2011.
 [13] Z. Huang, J. Xu, Z. Gong, H. Wang, and Y. Yan, “Source localization using deep neural networks in a shallow water environment,” *J. Acoust. Soc. Am.*, vol. 143, no. 5, pp. 2922–2932, May 2018.
 [14] J. M. Vera-Diaz, D. Pizarro, and J. Macias-Guarasa, “Towards End-to-End Acoustic Localization Using Deep Learning: From Audio Signals to Source Position Coordinates,” *Sensors*, vol. 18, no. 10, p. 3418, Oct. 2018.
 [15] S. Rusck, *Induced lightning over-voltages on power-transmission lines with special reference to the over-voltage protection of low-voltage networks*. Stockholm: KTH, 1958.
 [16] “Lightning Parameters for Engineering Application,” no. 269. CIGRE WG C4.407, pp. 65–102, 2013.
 [17] F. Rachidi, “A review of field-to-transmission line coupling models with special emphasis to lightning-induced voltages on overhead lines,” *IEEE Trans. Electromagn. Compat.*, vol. 54, no. 4, pp. 898–911, 2012.
 [18] F. Napolitano, A. Borghetti, C. A. Nucci, F. Rachidi, and M. Paolone, “Use of the full-wave finite element method for the numerical electromagnetic analysis of LEMP and its coupling to overhead lines,” *Electr. Power Syst. Res.*, vol. 94, pp. 24–29, 2013.
 [19] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
 [20] R. Polikar, “Ensemble based systems in decision making,” *Circuits Syst. Mag. IEEE*, vol. 6, no. 3, pp. 21–45, 2006.
 [21] L. Rokach, “Ensemble-based classifiers,” *Artif Intell Rev*, vol. 33, pp. 1–39, 2010.
 [22] T. Chen and T. He, “Higgs Boson Discovery with Boosted Trees,” in *HEPML '14 Proceedings of the 2014 International Conference on High-Energy Physics and Machine Learning*, 2015, vol. 42, pp. 69–80.
 [23] S. Sekioka, “An Equivalent Circuit for Analysis of Lightning-Induced Voltages on Multiconductor System Using an Analytical Expression,” in *International Conference on Power Systems Transients (IPST'05)*, 2005.