Article - 75 years - Special Edition

TrackingStorm: Visualization Tool for a Storm Detector Network (SDN) in the LF Spectrum

Augusto Mathias Adams[1]
https://orcid.org/0000-0002-2683-6098

Edson Masao Odake Junior[1]
https://orcid.org/0000-0002-7149-4111

Armando Heilmann[1]*
https://orcid.org/0000-0002-0883-8778

Horácio Tertuliano Filho[1]
https://orcid.org/0000-0002-9683-4119

Anselmo Daniel Adams[2]
https://orcid.org/0000-0003-1872-5301

Luis Augusto Cordeiro dos Santos[1]
https://orcid.org/0000-0002-1193-2693

César Augusto Dartora[1]
https://orcid.org/0000-0001-5110-5214

[1] Federal University of Paraná (UFPR), Group of Signal Propagation Systems (SPS) and Atmospheric Electricity Phenomena (FEA) from University Federal of Paraná – (SPS/FEA-UFPR), Curitiba, Paraná, Brazil; [2] Eaatech Development LTDA, Department of Information Technologies, Curitiba, Paraná, Brazil.

HIGHLIGHTS

- Storm Dynamics (path/velocity/heading) coherent with the simulation values.

- Alerts: 86.57% effective alarms for short/long time storms for cells ahead the storm.

- WAP: 75-99% of lightning's in area of probability predictions.

Abstract: During the last year the Group of Atmospheric Electricity Phenomena (FEA/UFPR) developed a short range lightning location network based on a sensor device called Storm Detector Network (SDN), along with a set of algorithms that enables to track storms, determining the Wide Area Probability (WAP) of lightning occurrence, risk level of lightning and Density Extension of the Flashes (DEF), using the geo-located lightning information as input data. These algorithms compose a Dashboard called Tracking Storm Interface (TSI), which is the visualization tool for an experimental short range Storm Detector network prototype in use on the region of Curitiba-Paraná, Brazil. The algorithms make use of Geopandas and clustering algorithms to locate storms, estimate centroids, determine dynamic storm displacement and compute parameters of thunderstorms like velocity, head edge of electrified cloud, Mean Stroke Rate, and tracking information, which are important parameters to improve the alert system which is subject of this research. To validate these algorithms we made use of a simple storm simulation, which enabled to test the system with huge amounts of data. We found that, for long duration storms, the tracking results, velocity and directions of the storms are coherent with the values of simulation and can be used to improve an alert system for the Storm Detector network. WAP can reach at least 75% of prediction efficiency when used 6 past WAP data, but can reach

98.86% efficiency when more data is available. We use storm dynamics to make improved alert predictions, reaching an efficiency of ~87%.

Keywords: lightning; detection efficiency; location accuracy; short range detector; lightning risk alert system.

GRAPHICAL ABSTRACT

INTRODUCTION

Last year the Group of Atmospheric Electricity Phenomena (FEA/UFPR) developed a short range network prototype using a sensor device called Storm Detector device (SD) [1], which was formerly used mainly for research purposes, equipped with an AS3935 sensor [4] for lightning detection, controlled by an ESP32 [5], a System-On-Chip (SoC) microprocessor. It also embeds a GPS device for location/time synchronization and a small storage device for data logging. The ESP32 also has several embedded features, like WiFi and Bluetooth, which makes it suitable for standard Internet of Things (IoT) applications [5]. The Storm Detector Device firmware now supports network services, like remote lightning detection and geolocation data gathering through Message Queueing Telemetry Transport (MQTT) protocol [6]. The network, called Storm Detector Network (SDN) is now composed by 5 Storm Detector devices distributed in the region of Curitiba-Paraná, Brazil and it is in the homologation phase of development.

Lightning location network (LLN) data is used to access storm risks and compute lightning flash density (N_g) [7], It is also used at real-time by meteorological communities to forecast severe weather [11, 12]. The LLN performance parameters are defined to at least Detection Efficiency (DE), Location Accuracy (LA) and Classification Accuracy (CA) [7]. The Storm Detector Device was designed to be a low-cost alternative to access storm risks and alerts in large areas where communication and/or financial resources are a major problem, such as rural areas, remote electricity facilities, oil refineries, airports, tourist areas, electrical systems and populations close to mountainous regions, offering acceptable performance for such tasks, such as reasonable LA and real-time operation [1].

The Storm Detector Network algorithms are used to select detection events belonging to the same lightning and geo-locate them using Levenberg-Marquardt Algorithm (LMA) [13-15] as the main algorithm and True Range Lateration (TRL) [8-10] as an initial solution provider to determine the lightning geo-location. The detection events separation algorithms make use of Density-Based Spatial Clustering of Application with Noise (DBSCAN) algorithm [3] to separate the detected events in both time and space, and uses a combination of an algorithm inspired in Secure Positioning of Wireless Devices with Application to Sensor Networks Algorithm (SECPOS) [16] and raster scan to separate lightning events in the same area and time, eventually eliminating outliers and strange measures. Storm Detector Network is now capable of locating lightning events with 97% of Total Detection Efficiency (TDE) and a LA of 2 km of 90% confidence interval, with a mean LA of 532 m.

Lightning risk warnings are based on the nearest lightning location, which, without any more information or processing, is the main reason for low effective alarm rates. The effort to collect storm dynamics is aimed to at least partially solve the problem of false alarms, because it is possible to say that locations on the leading edge of the storm are likely to be effective under a storm at the moment or in the next minutes, while in the same way the locations on the rear edge of a storm are likely to be on a low alert level. The velocity and direction of storm dynamics are keys to develop better alert systems, helping to put more locations where the

alert is likely to be effective knowing these parameters. Moreover, storm tracking information and lightning density estimates are important topics regarding lightning risk analysis.

TrackingStorm Interface (this case a web environment) is based on Storm Detector Network and it shows processed data from network lightning locations, which at first are grouped in density areas called, by the way, storms. The thunderstorm areas are processed through a pipeline to give some prospects about the lightning behavior, and estimates movements of the electric charge center (that is, the centroids of thunderstorm area), determine its velocity and direction, the average of events in the area (Mean Stroke Rate), Wide Area of Probability of lightning occurrence (WAP) and computes densities estimates, like Density Extension of the Flashes (DEF) and Risk Level of lightning occurrence. The main goal of this pipeline is to gather information about the storms to help make an improved alert system which, along TrackingStorm Interface, will inform and recommend actions based on more data than only lightning. This article describes the TrackingStorm Pipeline (TSP) process and the tests made using a simple storm simulation process to input huge amounts of lightning data into Storm Detector Network, using a hypothetical sensor network deployed in the area of Paraná state as a lightning detection network.

MATERIAL AND METHODS

Programming Languages

The simulation script, the TrackingStorm Pipeline and server side TrackingStorm Interface were implemented using Python language, version 3.8 [17]. It also uses Numba JIT Compiler [18] to improve performance when possible. We made use of Flask microweb Framework [19] for server side interface, Scikit-Learn [20] library for clustering algorithms and Geopandas library [2] for geographic data manipulation (intersection, overlays, joins, unary unions and convex hull).

The TrackingStorm Interface frontend make use of JavaScript, HTML and Mapbox for presenting data.

We use the MySQL database engine to store lightning information by Storm Detector Network and to keep track of storms in TrackingStorm Pipeline. We also use MongoDB database engine to store the geographic layers produced by TrackingStorm Pipeline and to present processed information to the TrackingStorm Interface frontend by the server side system.

TrackingStorm Tools

Lightnings

The lightning location is considered as a basis for all the tools described in this section. By default it is selected a snapshot from the last 15 minutes of all lightning locations from Storm Detector Network. The snapshot is, then used to compute other tools described in this section. It shows the last lightning in the user interface.

Storm Areas

We define a storm area with polygons that surround the lightning of a given group/storm and is the first extracted information of the lightning location. A storm is defined as a group high density of lightning in a spatial domain. From the lightning locations, it is obtained by clustering the lightning using the distance between them as a grouping criterion. It shows an area encircled by a black line in the user interface and represents the identified storms.

Wide Area Probability of Lightning Occurrence (WAP)

The movement of electrically active storms cannot be predicted directly by the SD Device, however, we can consider the use of grouped and synchronized CG lightning detection measures to establish a geometric perimeter around the subsequent lightning, taking the preceding lightning as a reference. Therefore, as the electrical storm evolves at the time and the frequency of CG lightning increases, the algorithm begins to process the data every minute, thus, wider areas appear in the geo-located overlay on the map.

As the clustering considers multivariate space and time of lightning data, using numerical methods, is created a vertex geometry that coincides with all precursor lightning.

As lightning strikes are associated with the number of electrically active storms and the electric charges center of that storm, new wide areas can appear in real time, thus creating wide areas where lightning strikes, called Wide Area Probability, can occur. In this way, Wide Area Probability acts as a dynamic response

support that pre-determines specific (wide) areas in which lightning strikes can occur in the following minutes. It shows orange ellipses in the user interface, where the first WAP areas are faded and the last are vivid, representing the evolution of the storm according to this tool.

Storm Tracking

A representative point of a polygon is the mass center from its area, also called a centroid. The tracking tool keeps the historical information of the storm centroids for use in storm dynamics computation. It also shows a path in the user interface where the identified storm has moved from the beginning to the current time in the user interface, represented by a thin blue line inside the identified storm, from the first to the last centroid of the storm.

Storm Dynamics

An important information extracted from the tracking tool historical data is the storm dynamics. By dynamics we mean the movement of the electric charges center of a storm, which is useful to predict where the storm is heading, allowing the system to make alert predictions in a pro-active way. The velocity and dynamic direction of an active electrical thunderstorm are computed using the last two tracking centroids, if available. Also, are computed and recorded the Mean Stroke Rate (MSR), defined as the count of the occurred lightning within the storm area per 5 minutes.

The actual parameter values are in fact a moving average of the past 5 results of the computed ones, which TrackingStorm Pipeline also maintains an historical record to make it possible. It shows a blue dot, a blue dashed faded line and a blue arrow in the user interface, indicating the last storm centroid, the predicted path where the storm will be in the near future and heading of the storm. It also shows storm duration, mean stroke rate and velocity, when the mouse pointer is over the blue dot.

Density Extension of the Flashes (DEF)

The Density Extension of the Flashes (DEF) is a method used in the monitoring of lightning by satellite and that we integrate its methodology as another product of TrackingStorm Pipeline. We quantify the intensity of the storm according to cloud-to-ground lightning, triangulated by a network of Storm Detector sensors. This method is based on the density of flashes on the ground per minute of storm (flashes x km^{-2} x min^{-1}) computed assuming fixed squares of 1 km x 1 km. Since the size of the square roughly corresponds to the electric charges center of a typical storm, the density of flashes for a square defines a unitary storm, accumulated from 1 to 5 minutes. The Density of Extension of the Flashes assists in the temporal analysis of the accumulation of lightning strikes that occur per km^2, that is, within a Wide Area Probability per unit of time (minutes), allowing observing the evolution of the frequency of lightning over that area. It shows a colored grid in the user interface, from green to red, where green indicates a low density area and red indicates a high density area.

Risk Level of Lightning Occurrence

The National Oceanic and Atmospheric Administration (NOAA) [22] defines Lightning Alert Levels (LAL) in a scale of 6 levels, given as:

- LAL 1: No thunderstorms
- LAL2: Isolated thunderstorms. Light rain will occasionally reach the ground. Lightning is very infrequent, 1 to 5 cloud to ground strikes in a five minute period.
- LAL 3: Widely scattered thunderstorms. Light to moderate rain will reach the ground. Lightning is infrequent, 6 to 10 cloud to ground strikes in a 5 minute period.
- LAL 4: Scattered thunderstorms. Moderate rain is commonly produced Lightning is frequent, 11 to 15 cloud to ground strikes in a 5 minute period.
- LAL 5: Numerous thunderstorms. Rainfall is moderate to heavy. Lightning is frequent and intense, greater then 15 cloud to ground strikes in a 5 minute period.
- LAL 6: Dry lightning (same as LAL 3 but without rain). This type of lightning has the potential for extreme fire activity and is normally highlighted in fire weather forecasts with a Red Flag Warning.

The TrackingStorm also shows lightning alert level but since TrackingStorm Pipeline has no data about rainfalls, only lightning, using NOAA's scale is particularly difficult. TrackingStorm Pipeline defines lightning risk level by cells like Density Extension of the Flashes, not by storms, and assumes no hypothesis about rainfalls. It defines a scale of 4 risks levels (RL), given as:

- RL 0 (No risk): no lightning in the cell in a 5 minute period.
- RL 1 (Light risk): 1 or 2 lightning in the cell in a 5 minute period.
- RL 2 (Moderate risk): 3 to 4 lightning in the cell in a 5 minute period.
- RL 3 (Severe Risk): more than 4 lightning in the cell in a 5 minute period.

Although the storm area, from an alert perspective, is under alert of lightning events, the risk level definition shows how sparse is the thunderstorm. In other words, Risk Level shows how Mean Stroke Rate or NOAA's lightning activity is distributed along the space. It shows a grid in the user interface, like Density Extension of the Flashes, under the colors of yellow (Light risk), orange (Moderate risk) and red (Severe risk). Risk Level also shows a list of cities under risk of lightning occurrence, defined from the user location, indicating, when possible, the cities around the user point that are on alerts.

Alerts

Alerts is a user location based tool, which indicates alert predictions in user location, taking in consideration the storm dynamics and storm area. It uses a 2.5 km x 2.5 km grid of the coverage area from the entire Storm Detector Network hypothetical sensor network. Using the current storm dynamics (specially velocity and heading) and Mean Stroke Rate, it can make a projection of the alert cell candidates in the next 15 minutes.

For initiating storms and short time lightning events, the strategy of use the storm dynamics will fail. For these cases, the alert system marks cells that are under alert using only lightning as an input data. However, for well stablished storm dynamics, that is, for storms that a are more than 15 minutes old, it takes the following approach:

- The first derivative of Mean Stroke Rate gives insights about storm activity and so, used to predict storm lifespan in the future. The lifespan prediction tells if a storm can endure more 15 minutes or not. It returns a prediction horizon of 15 minutes if lifespan prediction is more than 15 minutes, the computed lifespan otherwise;
- For recently detected storm areas, prediction horizon will be the storm lifespan until it reaches 15 minutes;
- Project the storm area along the projected lifespan, to mark alert cell candidates;
- The alert level is given by computing the distance of each cell to the nearest lightning event in the head of the storm.

The TrackingStorm Interface tells the server about the user location and wraps this position to the nearest system coverage cell. To the user, alerts appear as a button where the color icons – yellow, orange and red – indicates the alert level as light, moderate and severe alert, based on the distance of the nearest lightning location, given as:

- Cell – Nearest Lightning distance >= 45 km: no alert, the lightning is too far to represent risk to the user.
- 45 km >= Cell – Nearest Lightning distance >= 30 km: yellow flag (light alert), tells the user to stay alert about lightning risk (a storm is approaching or, in case of short time events or not well stablished storm dynamics, a lightning occurred within 45 and 30km from the user cell).
- 30 km >= Cell – Nearest Lightning distance >= 15 km: orange flag (moderate alert), tells the user to prepare to find some cover (storm approaching and is near the user cell).
- Cell – Nearest Lightning distance <= 15 km: red flag (severe alarm), tells the user to find cover immediately (storm/lightning overhead).

Lightning Data

Lightning data is generated by a simulation script that, considering a selected random areas called storms (are circles with a given radius), inserts lightning in storms and inserts detection events obtained from lightning into Storm Detector Network , using an hypothetical sensor network deployed on the area of Paraná state. The simulation takes in consideration the parameters found in [1] for sensor distance estimation, given as:

$$L_E(\delta) = -1.667 \times 10^{-5} \delta^3 + 0.003501 \delta^2 - 0.2563 \delta + 13.5, \qquad (1)$$

Where, L_E represents the natural logarithm of the sensor energy parameter and δ is the distance in km. The simulation also takes advantage of the Absolute Detection Efficiency (ADE) curve also found by [1] to compute the detection probability of each hypothetical sensor, given as:

$$ADE_{SD}(\delta) = 88.11 (1 - 1/(1 + 309.6 \, EXP(-0.2888 \, \delta)) . \qquad (2)$$

For each lightning event, the simulation computes the distance to each sensor, then computes the detection probability also for each sensor and determine the sensors in detection state doing a simple Bernoulli trial, using the output of a random generator to compare with the probabilities given by (2). The sensor location error [1] of +/- 3km is inserted to each sensor computed distance as an additive gaussian noise. The simulation sends the data of each sensor to Storm Detector Network server using MQTT protocol, using a JavaScript-Object Notation (JSON) package defined by Storm Detector Network as a detection package. Storm Detector Network processes the sensor events and gives the lightning location.

Lightning Processing

The main goal of TrackingStorm Pipeline is to classify the incoming lightning events to group it in areas of lightning occurrence or storms were this task, is make by a clustering algorithm: DBSCAN.

Density-Based Spatial Clustering of Application with Noise (DBSCAN) is a clustering method that builds areas based on its density connected. DBSCAN is a type of partition clustering where high density areas are considered clusters whereas low-density or incompatible clusters are regarded as noise [21]. It is designed to find clusters and noise in a spatial dataset [3] using 2 parameters: minimum number of points to form a cluster (min_points) and maximum distance to consider two points as connected (eps) [3]. DBSCAN works well in 2D and 3D euclidean spaces, as well with any distance function. In our case, the main goal of DBSCAN is to find dense regions of lightning occurrence, to form storm areas that enables to compute storm parameters. We use the Scikit-Learn [20] DBSCAN implementation, along with a haversine function wrapper to implement geographic distance required by the algorithm. By default, min_points is set to 2 points and eps is set to 5 km to density reachability. As input data for the clustering process, each minute we select a snapshot from the last 15 minutes of lightning activity. Storm identification is make by detecting lightning events that are already grouped in storms any time within 15 minutes. If none of the lightning events were grouped by past runs, a new group of lightnings is formed and properly identified.

The storm areas are formed using geometric manipulations on a geographic dataframe from Geopandas (geopandas. GeoDataframe). By the way, the storm area is the polygon that surrounds all the lightning points for a given cluster/storm. Geopandas geometric manipulations gives the storm area with a little code, once the lighting locations of a given cluster are inserted in a geographic dataframe, all there is to do is a unary union of all geometry field followed by a convex_hull procedure.

Once the storm areas are defined and identified, the TrackingStorm Pipeline computes the Wide Area Probability of lightning occurrence (WAP) for each storm area. The TrackingStorm Pipeline also computes the mass center or centroids of all storm areas. The storm centroids are useful to keep track of storm movements in space and are used as a basis for storm dynamics computation (velocity, direction, heading and Mean Stroke Rate).

Historical data of storm centroids are collected and form the Tracking tool, which is the path where the storm moves through space. Storm Dynamics is computed using the distance between the 2 last centroids in the historical data, if available. TrackingStorm Pipeline also keeps historical data from dynamics and then determine the velocity, direction, heading and Mean Stroke Rate from the moving average of the last 5 elements of historical unfiltered dynamics data.

Density Extension of The Flashes, Risk Level and Alerts computations are pretty straightforward, once the storm area is determined, TrackingStorm Pipeline puts a grid of 1 km x 1 km inside the storm and counts the events in each cell. This is done by Geopandas geometric manipulation procedures (join and dissolve),

to count the events inside the same cell. All there is to do is divide by area and time window (Density Extension of the Flashes) and divide by time window / 5 (Risk Level). The alerts are computed at the end of pipeline because TrackingStorm Pipeline needs to know storm dynamics and areas to give alerts, projecting the storm evolution into the future and computing alerts using the distance of the nearest lightning location to each cell as parameter.

RESULTS

Data Generation

We simulated 40 storms in the simulation scenario, to test TrackingStorm Pipeline and TrackingStorm user the web interface. The thunderstorms parameters are:

- Storm radius: 1 to 5 km, randomly chosen.
- Storm Velocity: 5 to 50 km/h, randomly chosen.
- Storm Stroke Ratio: 10 to 40 lightnings/min, randomly chosen.
- Storm Duration: 40 to 60 minutes, randomly chosen.

A total of 36952 lightnings locations are inserted on the database by Storm Detector Network. We keep record of the storm in a separate file to compare with storm detection, tracking and storm dynamics. Values like velocity, Mean Stroke Rate, heading and centroid location are kept to compare with the computed values of TrackingStorm Pipeline.

Storm Identification

TrackingStorm Pipeline identified and tracked all of the 40 storms inserted by the simulator. However, there are some lighting events (1848 in total) that has no correspondence with any storm. This is expected because:

- Storm Detector Network data are with ~5 % of misplaced lightning locations (Location error >= 6 km).
- TrackingStorm Pipeline Storm Identification algorithms eliminates data that don't fit the eps criteria (Two lightnings are to be 5 km from each other to be considered a storm).

Then this lightning loss (~5%) can be considered acceptable for good performance.
In the same way, storm areas generated from identified storms are slightly lower that the actual area from the original storm (~2% lower), which is coherent with the lightning loss of storm identification. It also indicated that misplaced lightning locations are, by default, outside the storm area. However, TrackingStorm Pipeline still identifies the storm area with a mean accuracy of 98%.

Tracking and Storm Dynamics

Figure 1 resumes The Tracking and Storm Dynamics tools of TrackingStorm for a chosen representative storm:

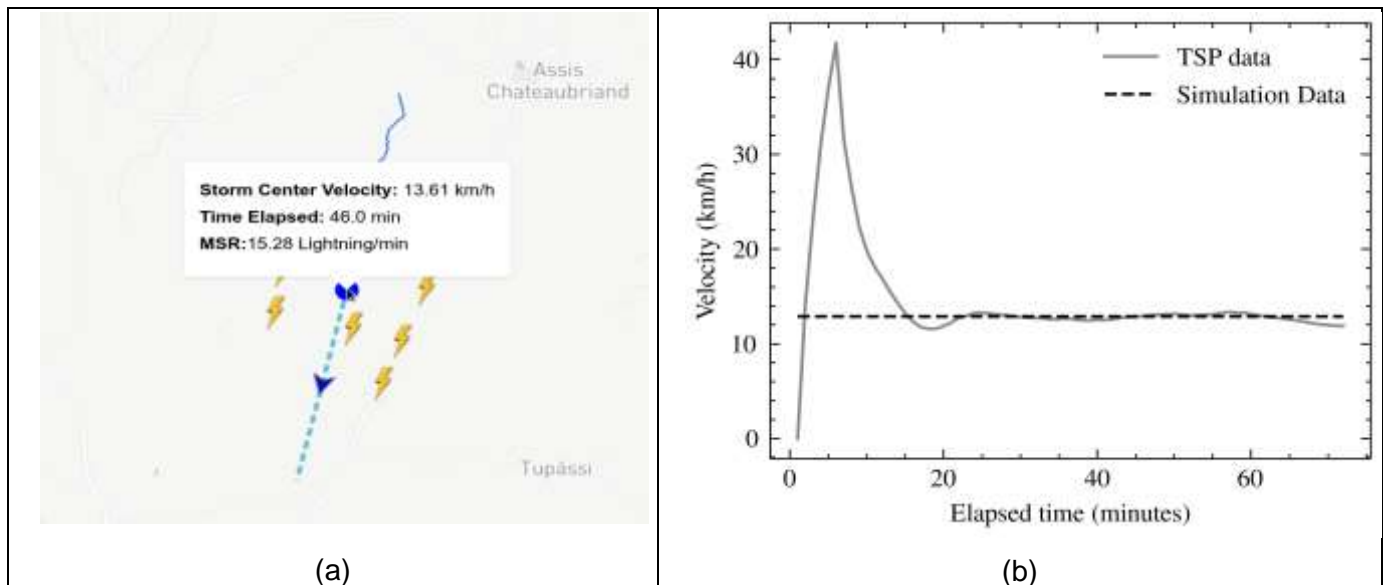(a)   (b)

Figure 1. Storm Dynamics parameters: (a) Snapshot of the Tracking and Storm Dynamics tools in action, for the chosen representative storm. The thin blue line represents the path the storm has moved, the blue dashed line is the last movement direction, the blue arrow is the heading of the storm and the dark blue dot is the last centroid location. Storm Dynamics also shows velocity, Mean Stroke Rate and storm elapsed time when the pointer passes over the last known centroid location (b) Evolution of storm velocity for the entire storm lifespan.

In Figures 1 and 2, the simulated values are shown as dashed lines. Storm Dynamics parameters determination relies on Tracking tool who gathers information about the storm centroids location over the time. The thin blue line in Figure 1(a) is the path formed by the past centroids location in the storm history. The centroids aren't well determined at the storm starts, so is the velocity and heading of the storm. This occurs because at the start TrackingStorm Pipeline don't have enough data to determine storm dynamics and any data collected by this tool is unreliable from the start until approximately 10 to 20 minutes. After that the parameters values are reliable for alert predictions.
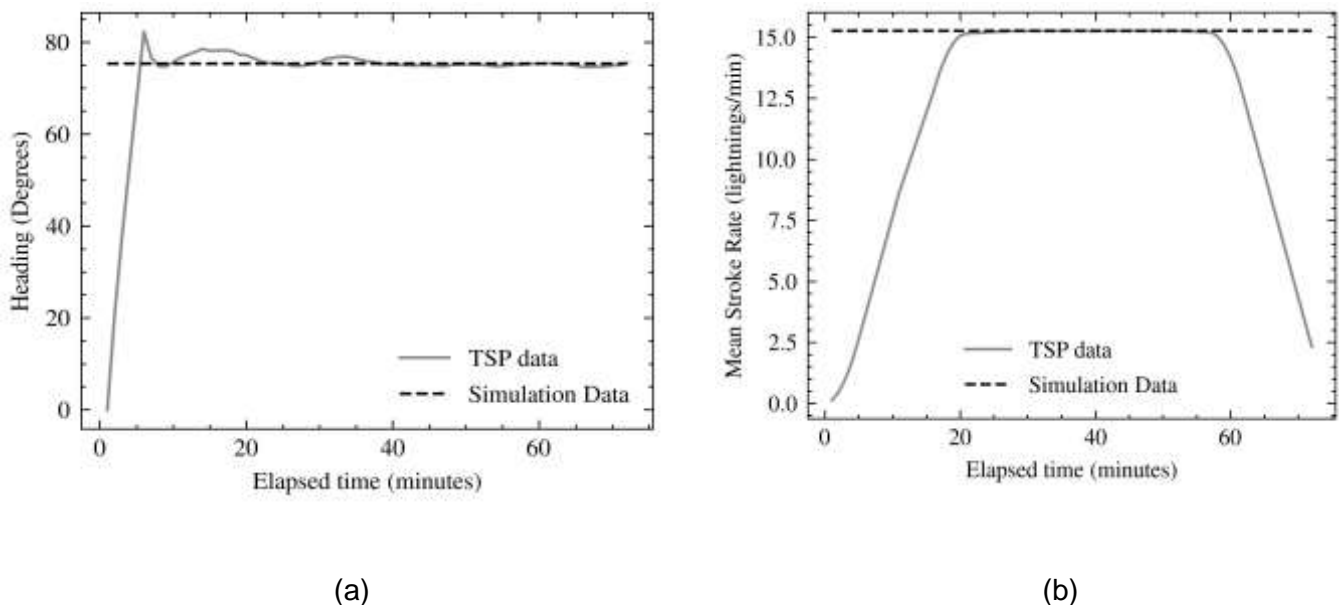

(a)   (b)

Figure 2. Storm Dynamics parameters: (a) Evolution of the storm heading for the entire storm lifespan (b) Evolution of Mean Stroke Rate for the entire storm lifespan.

The heading parameter, in the most cases analyzed, is the first parameter to settle, taking about 4-6 minutes to converge to the simulated value. The velocity, also important for alert predictions, takes approximately 15 minutes to converge. Mean Stroke Rate presents a positive slope at storm beginning, converging to simulated value after 20 minutes, and has a negative slope after the storm stops producing lightning data. Mean Stroke Rate slopes are determined by lightning snapshot time window (15 minutes) and

by moving average filter (5 raw measures to settle). After settlement, the velocity, heading and Mean Stroke Rate stays nearly the simulation values until the storm stops.

Wide Area Probability (WAP)

Figure 3 shows the Wide Area Probability Tool in action and the Wide Area Probability analysis made using entire lightning dataset of a chosen representative storm:
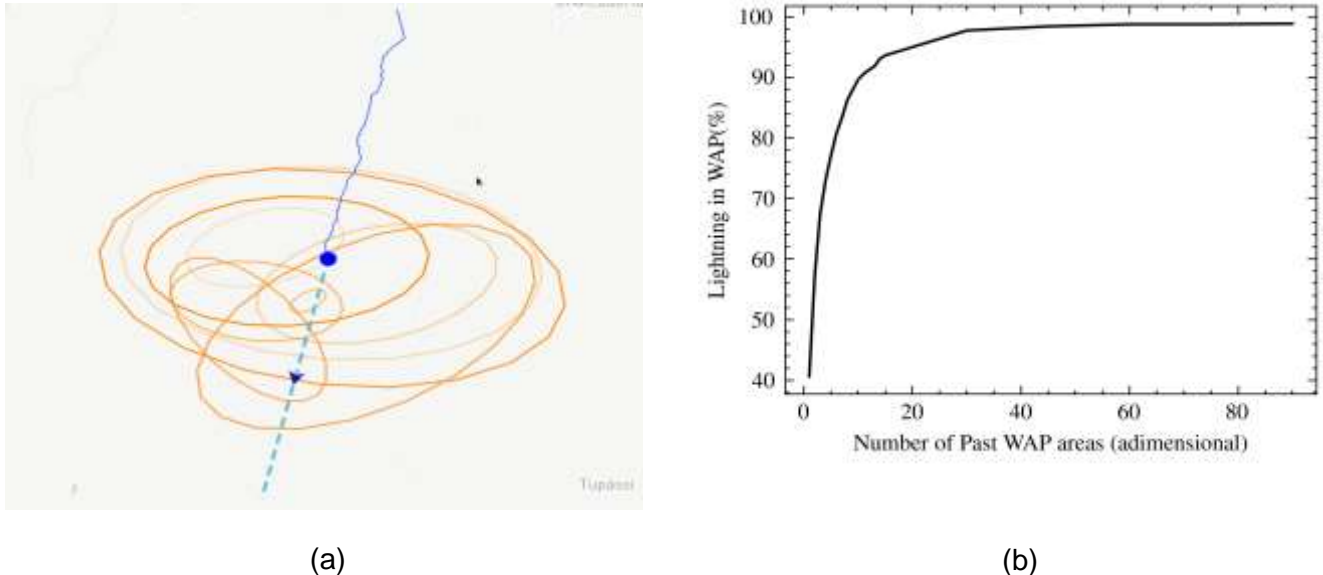

(a) (b)

Figure 3. Wide Area Probability Tool in action: (a) A snapshot of the Wide Area Probability tool in action. Wide Area Probability are represented by orange ellipsis in TrackingStorm Interface showing where the lightning are more likely to occur. (b) Efficiency of lightning occurrence in the already defined Wide Area Probability versus the amount of past Wide Area Probability that defines the delimiting area. It shows that it takes approximately 6 past Wide Area Probability data to have at least 75% efficiency and it can reach near 99% as the storm evolves and TSP uses more Wide Area Probability data.

Wide Area Probability tool shows orange ellipsis in TrackingStorm Interface, which the vivid ones are the last Wide Area Probability data. Wide Area Probability analysis shows that, for a single Wide Area Probability data, the next lightning will occur in this area with a probability of ~40%, increasing when we add more data. It takes approximately 6 past Wide Area Probability data to reach a probability of 75%, increasing fast to 95% when we have 20 past Wide Area Probability data to analyze, until it reaches 98.86% for more than 30 past Wide Area Probability data. The explanation is, as the storm evolves, Wide Area Probability wraps around the storm enlarging its area, delimiting the region where the lightnings should occur.

Density Extension of Flashes (DEF)

Figure 4 shows Density Extension of Flashes (DEF) tool in two different storms, with different radios and Mean Stroke Rates. Figure 4(a) shows a sparse storm, where Density Extension of Flashes is diluted among many cells and rarely reaches high values, while Figure 4(b) shows denser areas where Density Extension of Flashes has high values.
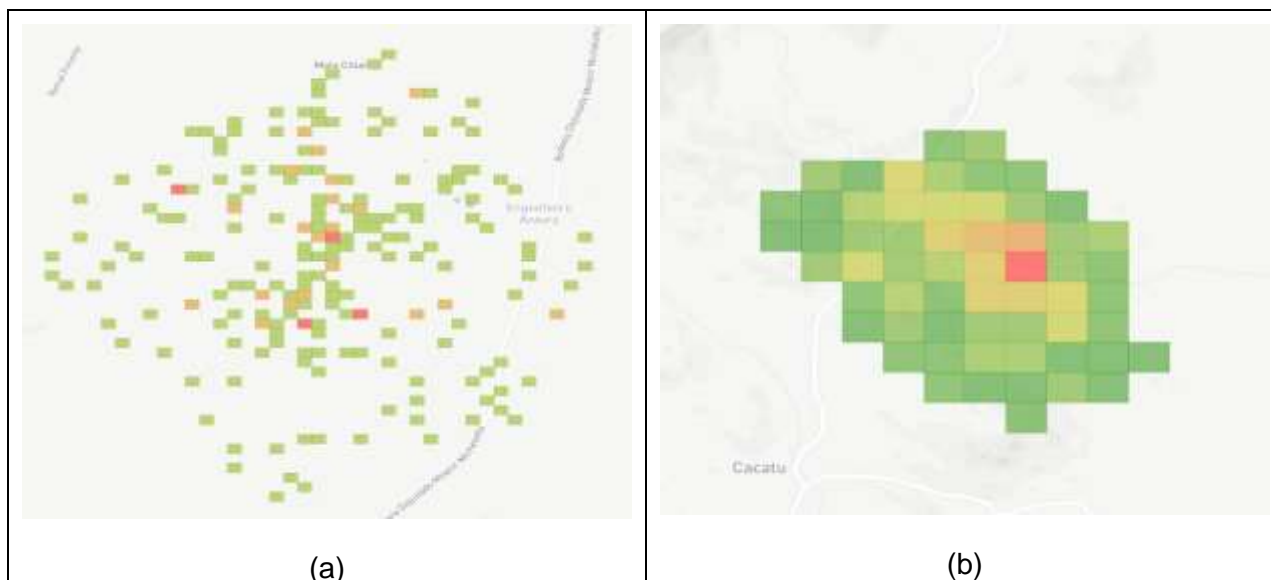
Figure 4. Density Extension of Flashes Tool in action: (a) A snapshot of the tool from a sparse, medium Mean Stroke Rate storm. It shows that the lightnings are scattered and rarely form high density cells (There are 3 high density cells in the storm area, followed by some cells of medium density and many of low density cells) (b) a snapshot of the tool from a dense, medium Mean Stroke Rate storm. It shows that the lightnings in a restricted area and form a center of dense cells, surrounded by some cells of medium density. The storm are enclosed by low density cells.

Density Extension of Flashes tool is currently classifying areas from green (low density) to red (high density). A sparse storm is shown in the tool as a scattered group of cells, rarely forming high density areas. This is what Figure 4(a) shows: a large radius storm with a medium Mean Stroke Rate, or even higher, is shown in the tool as scattered and low density because of the huge storm area, that is, although it is classified by NOAA's criteria as intense, in fact, from Density Extension of Flashes perspective it can be demonstrated that it has many cells of low density activity. On the other hand. Figure 4(b) shows a low radius, moderate Medium Stroke Rate storm in the tool. It has well defined areas of high, medium and low densities and shows 2 centers of moderate/high density cells.

Risk Level of Lightning Occurrence

Figure 5 shows Risk Level tool in two different storms, with different radios and Mean Stroke Rates. Figure 5(a) shows a sparse storm, where Risk Level is diluted among many cells and rarely reaches high risk level, while Figure 5(b) shows denser areas where Risk Level has high risk areas.

Risk Level tool classifies areas of low risk level (yellow), moderate risk level (orange). A sparse storm is shown in Risk Level as a scattered group of cells, rarely forming areas of high risk level, as in Figure 5(a). This is also caused by the big storm area like in Density Extension of Flashes tool and in this figure we see only areas of low risk level. On the other hand, Figure 5(b) shows a low radius, medium Mean Stroke Rate storm in Risk Level tool. It has well defined areas of high, moderate and low risk levels and shows 2 centers of moderate/high risk levels.
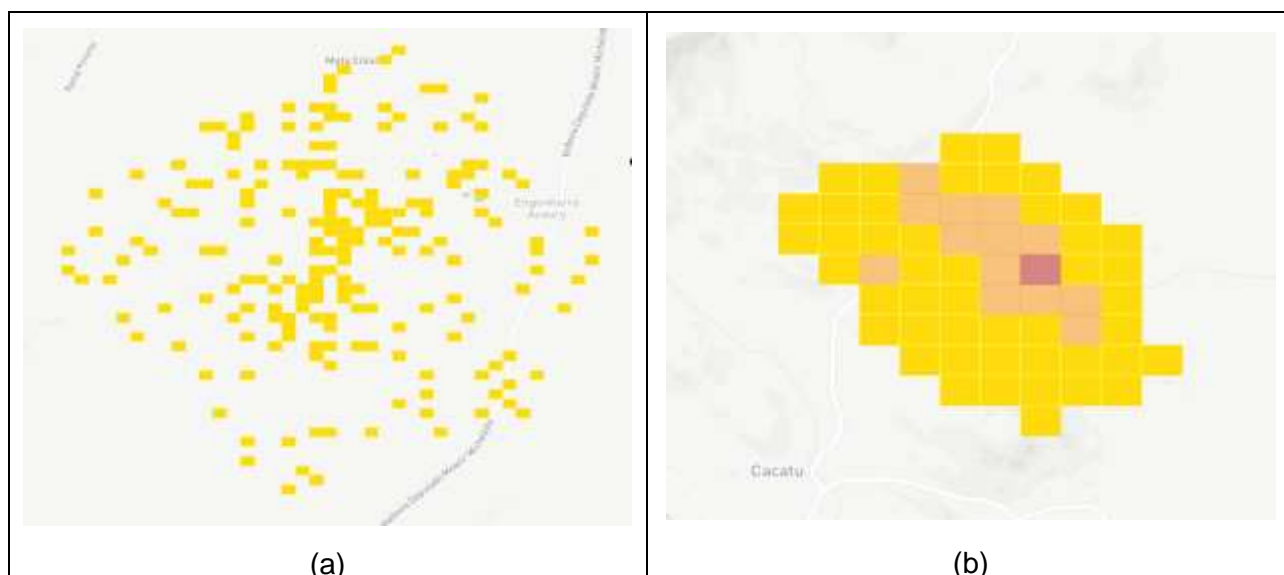
Figure 5. Risk Level Tool in action: (a) A snapshot of Risk Level tool from a sparse, low Mean Stroke Rate storm. It shows that the entire storm area is under low risk level of lightning occurrence (b) a snapshot of Risk Level tool from a dense, medium Mean Stroke Rate storm. It shows 2 cells in high risk, surrounded by cells of medium risk level. The storm area is enclosed by low risk cells.

Alert System Evaluation

Reference [23] sets four parameters to evaluate lightning risk warnings:

- Effective Alarm (EA) is the warning that was triggered before the CG lightning occurred inside the area of concern(AOC);
- Lead Time (LT) is the time interval between the time of the warning and the occurrence of a CG lightning inside the AOC;
- Failure to warning (FTW), when the CG lightning occurs in the AOC without previous warning;
- False Alarm (FA) is an alarm triggered without the occurrence of a CG inside the AOC.

AOC is defined by TrackingStorm Pipeline analysis as, when a storm is present, as the storm area or a circle of 20 km around a cell for initiating storms or short time lightning events, and the EA, FA and FTW are defined as:

- Effective Alarm is an alarm that is triggered by the storm area projection and then confirmed to be in the storm area in the future or, when we have a short time storm, a lightning inside the cell's area;
- False Alarm is an alarm triggered by the storm area projection that is not confirmed to be in the storm area in the future, or is outside the 20 km circle around each cell for short time storms;
- Failure to warning is a lightning that occur without any warning, in both storm the area projection or the 20 km circle for short time storms.

We evaluated alerts at all cells defined by the coverage area for the 40 simulated storms, computing Effective Alarm, False Alarm and Failure to warning and the results are tabulated in Table 1. It shows that are there at least 86.57% of effective alarms, computed by evaluation criteria, failing to warn about 55%. In other words, when TrackingStorm Pipeline issued an alert for a given cell, there are 86.57% chances that it would be effective and are ~55% chances that system will fail at the first warning, since currently the system does not predict when the first lightning strikes.

Table 1. EA, FA and FTW obtained by evaluation of TSP Alert system

Parameter	Alert Cells	% of Data[1]
EA	1663	86.57
FA	258	13.43
FTW	1046	54.45

[1]Percentages are referred to the total of alerts issued by TrackingStorm Pipeline (EA + FA).

DISCUSSION

The TrackingStorm Pipeline and Interface test made by simulation showed what we should expect from the system. We tested the entire process, from lightning detection, lightning location made by Storm Detector Network, storm identification and computing parameters, and TrackingStorm Interface is used to show the results. The tools discussed in this paper are based only on lightning locations over time, and are a foundation to a short range lighning location network made by SPS/FEA/UFPR.

The results of Storm Identification and Storm Area determination are within expected limits. There are 5% of misplaced lightning locations from Storm Detector Network, which is coherent with the total lightning rejected by Storm Identification, which by the way, reduced the average Storm Area by 2%.

The Tracking/Storm Dynamics and Wide Area Probability results are, however, surprising. Despite the fact that dynamics is inherently delayed by the lightning snapshot window (15 minutes), for long time storms it establishes a reliable way to predict which areas will be alerted in the next few minutes, avoiding false alarms in areas behind the storm direction, delimiting the alert prediction only to the storm area projection and its vicinity in the next few minutes. Considering the last 6 past Wide Area Probability data as an area prediction, it reaches at least 75% of efficiency in area on lightning risk, reaching 95% when used the last 20 past Wide Area Probability data as an area prediction. It could even reaches 98.86% when more than 30 past Wide Area Probability data is available.

The results of the alert system employed by TrackingStorm Pipeline is also surprising. It reaches 86.57 % of mean Effective Alarms in the 40 storms simulated for this paper, and has a low False Alarm ratio. Indeed, it has 54.45% of Fail to Warning, but this behavior is expected, as the network could not predict when the first lightning strikes for each of 40 simulated storms. The use of storm dynamics to predict alert areas is useful, even for short time storms or lightning events, since the alert system will not extend the storm area for long time until it reaches at least 15 minutes of lifespan, minimizing marking alert cells that would eventually contribute to False Alarms.

However, there are some improvements to be made. For example, tracking tool is based only on raw location values of the storm area centroids and it is noisy, as we can clearly see in Figure 1(a). A smoothing strategy for data collected by tracking tool or even the use of an extended Kalman filter for storm dynamics would, in theory, improves the velocity/heading parameters of storm dynamics. A lightning grid counting like we used for Density Extension of the Flashes/Risk Level before storm identification would agglomerate more lightnings to storms, due to the fact that it would mitigate location errors and ceil the positions to the area centroids, easing the work for DBSCAN algorithm.

The simulation uses a random strategy to emulate real situations that lightning nature is able to produce and to simulate sensor behavior, except for the storm area (currently a circle) and Mean Stroke Rate (considered constant, although time effects from simulation produce noisy data). Storm movements are governed mainly by wind direction in a real environment and is quite granted that, for short periods of time, it will remain constant on a straight line. However, the fact that we can track storms in any direction, as well with different Mean Stroke Rates, tells us that the system operation in a real environment is likely to happen consistently with simulation efficiency parameters.

Conflicts of Interest: The authors declare no conflict of interest.

REFERENCES

1. Heilmann A, Da Silva E, Filho H, Schuhmann J, Adams A, Dartora C, De Lima D, Burkarter E. "Detection Efficiency Analysis of Atmospheric Discharges using AS3935 Sensor : Data Correlation of LINET Network".   International Symposium on Lightning Protection (XV SIPDA). 2019; XV(972): 1–7.
2. Jordahl K. Geopandas: Python Tools for Geographic Data [Internet]. 2014. Available from: https://github.com/geopandas/geopandas.
3. Ester M, Kriegel H, Sander J, Xu X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, ser. KDD'96. AAAI Press. 1996; 226–31
4. Microsystems A. AS3935 Franklin Lightning Sensor IC [Internet]. 2012. Available from: https://ams.com/documents/20143/36005/AS3935_DS000385_1-00.pdf/6de19558-a8e3-11f2-a770-cbfcfa2ab32e
5. ESPRESSIF. ESP32 technical reference manual. [Internet]. 2018. Available from: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

6. Masdani M, Darlis D. A Comprehensive Study on MQTT as a Low Power Protocol for Internet of Things Application. IOP Conference Series: Materials Science and Engineering. Dec 2018; 434(012274): 1. doi.org/10.1088/1757-899x/434/1/012274

7. Pédeboy S. An Introduction to the IEC 62858: Lightning Density based on Lightning Locating Systems. International Lightning Protection Synposium in Shenzen China. Oct 2018; 2018.

8. Mialdea-Flor J, Segura-Garcia S, Felici-Castell M, Garcia-Pineda J, Alcaraz-Calero M, Navarro-Camba E. Development of a Low-cost IoT System for Lightning Strike Detection and Location. Electronics. Dec 2019; 8(12): 1512. dx.doi.org/10.3390/electronics8121512.

9. Fang T. Trilateration and extension to Global Positioning System navigation. Journal of Guidance, Control, and Dynamics. 1986; 9(6): 715–717. doi.org/10.2514/3.20169.

10. Geyer M. Earth-referenced Aircraft Navigation and Surveillance Analysis [Internet]. Jun 2016. Available from: https://rosap.ntl.bts.gov/view/dot/12301.

11. Heilmann A, Morales C. Theoritical Analyses of the Location Errors retrieved in Brazil for the Zeus VLF System. VII International Symposium on Lightning Protection. São Paulo: IEE/USP. 2005; 8: 387–391.

12. Heilmann A, Morales C. Location Error Evaluation for the Zeus Long Range Lightning Monitoring Network in Brazil. Journal of Lightning Research. 2008; 3:10–9.

13. Levenberg. A Method for the Solution of Certain Non-Linear Problems in Least Squares. Quarterly of Applied Mathematics. 1944; 2(2): 164–8. jstor.org/stable/43633451.

14. Marquardt W. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. Journal of the Society for Industrial and Applied Mathematics. 1963; 11(2):431–41. doi.org/10.1137/0111030.

15. Nielsen H. Damping Parameter in Marquardt's Method. Technical University of Denmark, Department of Mathematical Modelling. 1994. books.google.com.br/books?id=kqqjygAACAAJ.

16. Čapkun S, Hubaux J. Secure Positioning of Wireless Devices with Application to Sensor Networks. 2011.

17. Van Rossum G, Drake F. Python 3 Reference Manual. Scotts Valley, CA: CreateSpace. 2009.

18. Lam S, Pitrou A, Seibert S. Numba: A LLVM-based Python JIT Compiler. Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC, ser. LLVM '15. New York, NY, USA: Association for Computing Machinery. 2015. doi.org/10.1145/2833157.2833162.

19. Grinberg M. Flask Web Development: developing Web Applications with Python. O'Reilly Media, Inc. 2018.

20. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V et al. Scikit-Learn: Machine Learning in Python. Journal of machine learning research. 2011; 12(Oct): 2825–30.

21. Nagpal P, Mann P, Comparative study of Density Based Clustering Algorithms. Int. J. Comput. Appl. 2011; 27(11): 421–35.

22. NOAA. Lightning Activity Level (LAL) [Internet]. 2014. Available from: https://graphical.weather.gov/definitions/defineLAL.html.

23. Ferro M, Yamasaki J, Pimentel D, Naccarato P, Saba M. Lightning Risk Warnings based on Atmospheric Electric Field measurements in Brazil. Aerospacial Technology Management. 2011; 3(3): 301-10.