LoRaPlan Software Technical Documentation

Table of Contents

Introduction and Fundamentals

This tool is designed to easily plan out gateway locations and estimate the range for those gateways. The tool utilizes .csv files for data import and .hgt files for DEM Data. The file format will be specified in another chapter.

File Formats

Sensor Locations

For the sensor locations the tool expects a file in the following format:

Table 1: Sensor locations file format

lat	lon	sensor number

The table can contain multiple sensor types by extending the table on the right with the additional sensor types. This table has to follow the CSV file type conventions and use ";" as a separator.

Gateway Locations

For the gateways only name, lat and lon are required. The last two are used for propagation models that utilize those variables.

Table 2: Gateway locations file format

name	lat	lon	height in meters	environment

The environment variable can have the following values:
This table has to follow the CSV conventions and use ";" as a separator as well.

Table 3: Gateway location types

urban	suburban	rural

DEMs

This tool supports the SRTM dataset natively. Both the SRTM90 and the SRTM30 datasets work out of the box. Other DEMs need to be converted to the .hgt format following the exact format conventions for SRTM as well as the same file naming convention. The original download location from the USGS seems to be currently unavailable. Using NASA Earthdata all SRTM datasets are publicly available. The original source can be found at [1]. The NASA Earthdata webportal can be used for download. A Link with the relevant search term can be found at [2]. The known supported versions are: "NASA Shuttle Radar Topography Mission Global 1 arc second V003" and "NASA Shuttle Radar Topography Mission Global 3 arc second V003".

Table 4: Export Table Structure

ID	lon	lat	BestGW	RSSI	SF	NumberOfSensors	OtherGWs

Exports

The Software automatically generates two files in the Directory containing the Python files of the application. These files contain the following table in CSV and JSON format: The purpose of these files is to continue using the results from the tool in other applications. The Table above contains an array for the OtherGWs section. Therefore it has to be specially parsed. In order to avoid having to create a CSV parser the JSON file can be imported as it contains the necessary formating. The Table references the lat/lon values from the grid representation used in the tool. Because of this there is a possibility for a mismatched number of IDs in this Output file when compared to the number of sensors. The RSSI value is directly dependent on the SF value which describes the lowest SF to reach the BestGW. If the SF is fixed in the Software these two values will always refer to the selected SF.

Geographic Projection

This tool uses a grid with a spacial resolution of 100x100 Meters. For converting between geographic representations and this grid a version of the mercator projection is used. The following formula represents the conversion from Lat/Lon to x/y that the software uses.
$x = (lon - centerlon) \cdot 5000000$
$y = arctanh(sin(lat)) \cdot -5000000$
centerlon is the centerpoint in the resulting Projection. The software calculates this center as follows: $centerlon = \frac{1}{2} \cdot (maxlon - minlon)$ and is therefore the center of the provided dataset including gateways.
Stretching the coordinates was necessary to ease the assignment of Grid cells but has no effect on accuracy.
For converting from the grid to Lat/Lon the gudermann-function is used.
$Lat = arctan(sinh(y))$
$Lon = x + centerlon$

Reading DEMs

In order to use some of the features in the Software, accurate height-information is required. For this Purpose a DEM is used. It is read using either lat/lon pairs or grid-coordinates using the gudermann-function. So far the Software has only been tested using the Official versions of the 30m/1" SRTM-DEM and the 90m/3" version.
The Software also supports a 1m accurate DEM provided by the Bayerisches Vermessungsamt which has been converted to GeoTIFF Format. It has only been tested with the city of Würzburg and should not be considered as a general option. It is possible to use the GeoTIFF format but some geographic details have to be provided. It is generally not recommended to use this filetype as it is not tested for compatability with other GeoTIFF files other than the specific file the original project demanded. In addition the 100m grid almost negates the benefits of this increased spacial resolution. Due to this issue the freely available SRTM 30m/1" dataset should be sufficient and not much difference in sensor reachability was observed in the original project.

Path Loss Models and RSSI

Fundamentals

The software supports many different path loss models, but processes them the same. First of all there are two modes for all path loss models:

1. Pure range by the constraints of the LoRa-Specification
2. Range determined by geographic height and a line of sight

For the first method a maximum reachable distance purely based on the specific path loss model used and all sensors within a circle with that radius are marked as reachable. The second method calculates the same distance and also marks the same sensors. However if a sensor is determined to be unreachable using a line of sight, which will be explained in a following section, the sensor is removed from the list of the specific gateway.
In order to calculate the maximum distance, the maximum path loss for the connection is needed. Following LoRa specifications by Semtech and assuming that the receiving antenna does not improve the RSSI, the software uses this table to map from spreading factor to RSSI [3].

Table 5: RSSI values

SF	7	8	9	10	11	12
RSSI(dBm)	-123	-126	-129	-132	-133	-136

A gain of 8dBm is added to the RSSI in order to represent the total Path Loss [4]. These values are based on reference datasheets from Semtech and can vary between models and sensor nodes. Notably no assumption about antenna performance is made and thus specific antenna configurations affect the results. The 8dBm gain represents the maximum power output the SX1257 chip can produce. If more accurate values are measured, this gain should be removed from the table used in the software. If custom values are desired, the table in the software needs to be adjusted.
The basic variables DEV_{elev} and f are fixed to specific values: $DEV_{elev} = 1m$ and $f = 868.1$ due to the lack of these variables for the original project. Some path loss models rely on additional information, specifically the height of the gateway above ground and the type of environment. If any one of them is missing and a model which uses them is selected the range will fall back to default values of: $GW_{elev} = 15m$ and $GW_{type} = urban$. The resulting RSSI will also use these values, thus giving a consistent dataset and range. All formulas are dissolved to describe distance directly in order to flag the sensors. Only sensors which are considered reachable will be evaluated for the RSSI and the number of reachable gateways. So all path loss models use the dissolved formula for determining reachable Sensors and the path loss equation for assigning RSSI and signal strength.

Free Space Path Loss (FSPL)

This model is the simplest of the path loss models in the software. It does not take any environmental factors into account. The following formula describes the relationship between distance and path loss [5]:

$$PL = 20 \cdot log_{10}(f) + 20 \cdot log_{10}(d) + 32.45$$

This Formula is dissolved to describe distance as follows:

$$d = 10^{\frac{20 \cdot log_{10}(f) - PL + 32.45}{20}}$$

Manhattan

The manhattan path loss model uses a fixed radius of 1Km for the reachability calculations. It should not be used as an accurate model for evaluation and is only for testing purposes. The RSSI calculations use the FSPL model.

Hata

The Hata model is the first one in the software to use the gateway elevation which is provided in the gateway file. The RSSI is calculated as follows [6]:

$$PL = 69.55 + 26.16 \cdot log_{10}(f) - 13.82 \cdot log_{10}(GW_{elev}) - 3.2 \cdot (log_{10}(11.75 \cdot DEV_{elev}))^2 + 4.97 + [44.9 - 6.55 \cdot log_{10}(GW_{elev})] \cdot log_{10}(d)$$

dissolving this formula to describe distance:

$$d = 10^{\frac{(-(69.55+76.872985-13.82 \cdot log_{10}(GW_{elev})-3.2 \cdot (log_{10}(11.54 \cdot DEV_{elev})^2)-4.97-PL)}{(44.9-6.55 \cdot log_{10}(GW_{elev}))}}$$

Lee

The Lee model was designed to predict range at 900mhz. As the LoRa Frequency in the EU is not that far off, it was chosen as an additional PL model. It also distinguishes between urban, suburban and rural geography and uses the elevation of the gateway. It also uses two variables dependent on the geography type, described in table 6:

Table 6: Lee model parameters

Type	L$_0$	γ
rural	89	43.5
suburban	101.7	38.5
urban	110	36.8

The Lee model predicts PL as follows [7]:

$PL = L_0 + \gamma \cdot log_{10}(d) - 10 \cdot log_{10}(F_0)$
F_0 is defined by:
$F_0 = F_1 \cdot F_2 \cdot F_3 \cdot F_4 \cdot F_5$
with:
$F_1 = (\frac{GW_{elev}}{30.48})^2$

$F_2 = \frac{Gain_{GW}}{4}$

$Gain_{GW}$ in this context refers to the gain of the base station relative to a half wave dipole. The software uses a fixed value of $\frac{1}{2}$. This is a low gain so real world results could be better than this prediction. However changing this value is not difficult.

$F_3 = (\frac{DEV_{elev}}{3})^2$ if $DEV_{elev} > 3$

$F_3 = (\frac{DEV_{elev}}{3})$ if $DEV_{elev} < 3$

$F_4 = \frac{f}{900}^{-2.5}$

$F_5 = Gain_{DEV}$

$Gain_{DEV}$ is the same measurement as $Gain_{GW}$ but for all sensors. This software uses the fixed value of 2 as no data for the sensors was available.
The formula can be dissolved to describe distance:

$d = 10^{\frac{L_0 - PL - 10 \cdot log_{10}(F_0)}{-\gamma}}$

Cost231

The Cost231 model is an extension to the Hata model and also utilizes both the gateway elevation and the gateway type. For the gateway type however no distinction between rural and suburban exists so the software treats them the same.
The Cost231 model predicts PL as follows [5]:

$PL = 46.3 + 33.9 \cdot log_{10}(f) - 13.82 \cdot log_{10}(GW_{elev}) - ah_m + (44.9 - 6.55 \cdot log_{10}(GW_{elev})) \cdot log_{10}(d) + c_m$

For urban environments ah_m and c_m can be calculated as:

$ah_m = 3.2 \cdot (log_{10}(11.75 \cdot DEV_{elev}))^2 - 4.79$
$c_m = 0$

For suburban and rural environments:

$ah_m = (1.1 \cdot log_{10}(f) - 0.7 \cdot DEV_{elev} - (1.5 \cdot log_{10}(f) - 0.8))$
$c_m = 3$

The formula can be dissolved to describe distance:

$d = 10^{\frac{46.3 - 33.9 \cdot log_{10}(f) + 13.82 \cdot log_{10}(GW_{elev}) + ah_m + PL - c_m}{44.9 + 6.55 \cdot log_{10}(GW_{elev})}}$

Ericsson

The Ericsson model also utilizes both gateway type and elevation. It is also parameter based and the parameters can be changed very easily in the source code.
Path loss can be calculated with the following formula [5]:

$PL = a_0 + a_1 \cdot log_{10}(d) + a_2 \cdot log_{10}(GW_{elev}) + a_3 \cdot log_{10}(GW_{elev}) \cdot log_{10}(d)$
$-3.2(log_{10}(11.75 \cdot DEV_{elev})^2) + 44.49 \cdot log_{10}(f) - 4.78 \cdot (log_{10}(f))^2$

The parameters used in the default version of the software:

The formula can be dissolved to describe distance:

Table 7: Ericsson model parameters

Parameter	a_0	a_1	a_2	a_3
rural	45.95	100.6	12.0	10.0
suburban	43.20	68.93	12.0	0.1
urban	36.2	30.2	12.0	0.1

$$d = 10^{\frac{a_0 + a_2 \cdot log_{10}(GW_{elev}) - PL + 3.2(log_{10}(11.75 \cdot DEV_{elev})^2) + 44.49 \cdot log_{10}(f) - 4.78 \cdot (log_{10}(f))^2}{-a_3 \cdot log_{10}(GW_{elev}) - a_1}}$$

Line of Sight and Fresnel Zone

This tool supports calculation of the collision with objects in the line of sight between gateway and end device. Additionally a 2d implementation for fresnel zone interference can be utilized.

Line of Sight

In order to calculate the line of sight efficiently without additional memory for a full 3d representation two straight lines are calculated.
The first line is used to determine the x and y coordinates of the line of sight and results directly from the x and y coordinates of the currently viewed gateway and the currently viewed sensor. It represents the line of sights x and y coordinates form a "birds eye" view.
the second line is calculated for the z dimension. It is located on a plane rotated 90° from the x and y axis and its cutting straight is the first line. The two anchor points it uses is the elevation data for the gateway location and the device location from the DEM. For the gateway the information in the gateway file is added to this value.
If anywhere on this path a collision with the ground is detected, the sensor will be marked as unreachable without any consideration for damping and a possible reachability.

Fresnel Zone

In addition to the simple line of sight calculation a 2d implementation of the fresnel zone radius can be utilized. The radius of the fresnel zone can be described with the following formula [8]:

$$F_n = \sqrt{\frac{n \cdot f \cdot d_1 \cdot (d_{total} - d_1)}{d_{total}}}$$

n = the fresnel zone number. This tool uses the first fresnel zone.
d_1 the distance from the gateway. The radius F_n is for this distance.
d_{total} the total distance from gateway to end device.
This radius is then subtracted from the second line for each point that gets checked. Therefor the 2d ellipsoid is subtracted from the line of sight.

KML Vector Data

This software can utilize KML files to enhance the precision of the line of sight calculation. The files must adhere strictly to KML guidelines. They are processed as the last step in the pipeline. Because these vector data are only sensible to use when height information is present this step requires a DEM to also be specified for at all accurate results. The main check for collisions is handled by the matplotlib library. As the software processes the sensors in a 100m grid, obstacles in the KML dataset may misrepresent the reachability of specific sensor nodes, as their line of sight in the real world may not be impeded by the misrepresentation due to the 100m grid in the software. This is especially the case for smaller distances to the gateway where the fresnel zone is less pronounced and an obstacle would just lower the RSSI and not completely cut off the transmission. This implementation does not perform any assessment about structural damping of the signal and evaluates every device with no 100% clear line of sight as unreachable. If these inaccuracies are not acceptable, it is recommended to not use KML vector data.

Additional Features

The Software allows for several additional capabilities not related to the data processing methodology.

Draw DEM map

This allows for displaying the selected DEM as a background for the map. It can be slow to draw depending on the OS and Hardware. This feature has no impact on the results of the calculation and the RSSI and reachability values, as well as exported datasets will be identical.

Draw Grid lines

This Feature draws Lines between the 100m wide grid cells in order to improve the sense of scale. This feature has no impact on the results of the calculation.

Sweep over all SFs

This feature evaluates the reachability for the devices using all SF values LoRaWAN allows. The Export files and the visual information in the tool will reference the lowest SF and matching RSSI value, the device could still connect to the given Gateway. The values present in the tool represent the earliest evaluated gateway with a given SF. Therefor, if two gateways share identical connectivity attributes (same SF, same RSSI), the gateway, which is processed earlier takes precedence. This can only partly be influenced by the user without modifying the source code. If a different implementation is desired, the methods **calculatereachability_without_height** and **calculatereachability_with_height** have to be altered in order for all PL models to function consistently.

Important Variables

Sensorgrid

The sensorgrid is the main working 100m grid. It is layed out as a table for each x and y coordinate pair. The following table show the various uses of the entries.

*: Only used when the coordinate contains a gateway

Table 8: Sensorgrid layout

Index	0	1
Information	0/1-Gateway Flag	INT-Reachable sensors*
Index	2	3-x
Information	INT-Index in the gateway file*	INT-Number of sensors

Linkgrid

The linkgrid is an extension to the sensorgrid and has the same x and y dimension as the sensorgrid. It holds additional information about the best connection and came up later in development therefore it is not unified with the sensorgrid. It uses the following table:

Table 9: Linkgrid layout

Index	0	1
Information	INT-GW index with best connection	INT-Rx Power/RSSI in dBm
Index	2	
Information	INT-Number of reachable gateways	

dynamic propagation model

To indicate that a propagation model requires additional information from the gateway file this flag must be set for the propagation model. Any new implementation for additional propagation models also has to account for missing information as there is no check to ensure the required information is present. The default values for the existing propagation models have been described in section "Path Loss Models and RSSI - Fundamentals".

Important Methods

Main

generate mapframes

This method is responsible for ensuring a consistent scale factor for both the gateways and the sensors by combining them and then requesting the projection from maputils.

setup sensorgrid

sets up both the sensor and link grid according to the maximum extend of the generated mapframes.

placement action

This is the main coordinator. It is the method called when the user presses the Visualize button in the software. It clears the Sensor file contents, generates the sensor and link grid, requests the reachability calculation and draws the screen.

csv export

This method exports the results from the tool to CSV and JSON. You can alter this method to filter out or add other information to this export if necessary.

Maputils

The mercator projection, guderman, fresnel radius and findcenterlon methods are self explanatory and should not be changed in order for the projection and reachability to function as intended.

generatescreenmap

generates a grid from the provided dataframe using the mercator projection. It is not the final sensor or link grid as all sensors and gateways are combined for this dataframe. It also offsets the mercator projection (usually (0,0) is in the center) but for this application (0,0) is the top left corner for easier drawing.

getelevation

provides methods for parsing either the .tiff DEM (specifically checked for) or the SRTM DEM. This mathod has a variation for both the elevation from the map coordinates or directly from the lat lon coordinate pair.

readkml and kmlintersectswithline

Both of these methods prepare the KML files (if provided) for evaluation using matplotlib.

calculaterssi

All PL-models must be implemented in this in their original not dissolved format. This method populates the RSSI entry in the sensorgrid table for each reachable sensor. It came up later in development so it is not used for calculating the reachability.

calculatereachability without height

This method uses the dissolved PL-models to mark all sensors in a circle around them as reachable. It also fills in the linkgrid.

calculatereachability with height

Uses the line of sight check in addition to the PL-models. It fills in the same fields as the version without height and is used when at least a DEM is provided.

is in line of sight

Calculates both of the 2D representations of the line of sight explained in the section Line of Sight. If a KML Path is provided it also invokes the check method as well as the fresnel radius method if requested by the user. It iterates over a predetermined distance difference and checks for collisions.

List of Tables

References

[1] "Usgs srtm download (currently unavailable 2021-04-26)," accessed 2020-09-14. [Online]. Available: https://dds.cr.usgs.gov/srtm/version2_1/SRTM3/Eurasia/

[2] "Nasa earthdata srtm download (requires sign in)," accessed 2021-04-26. [Online]. Available: https://search.earthdata.nasa.gov/search?fi=SRTM&as[instrument][0]=SRTM

[3] "Sx1276 datasheet - semtech," accessed 2021-04-26. [Online]. Available: https://semtech.my.salesforce.com/sfc/p/#E0000000JelG/a/2R0000001Rbr/6EfVZUorrpoKFfvaF_Fkpgp5kzjiNyiAbqcpqh9qSjE

[4] "Sx1257 datasheet - semtech," accessed 2021-04-26. [Online]. Available: https://semtech.my.salesforce.com/sfc/p/#E0000000JelG/a/44000000MDmO/OfVC_rbxi4JjkT4hLzU1kq4gOXb4POLNRprWlqxRIZs

[5] Y. Alqudah, B. Sababha, A. Elnashar, and S. H. Sababha, "On the validation of path loss models based on field measurements using 800 mhz lte network," *2016 Annual IEEE Systems Conference (SysCon)*, pp. 1–5, 2016.

[6] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Transactions on Vehicular Technology*, vol. 29, pp. 317–325, 1980.

[7] D. Dobrilovic, M. Malic, D. Malić, and S. Sladojevic, "Analyses and optimization of lee propagation model for lora 868 mhz network deployments in urban areas," *Journal of Engineering Management*, vol. 7, pp. 55–62, 2017.

[8] A. Larsson, A. Piotrowski, T. Giles, and D. Smart, "Near-earth rf propagation - path loss and variation with weather," *2013 International Conference on Radar*, pp. 57–63, 2013.