UAV project python scripts information

This aim of this file is to clarify the python codes functionality and usage

=============================================================

**LTE\_logging\_test.py:** this python script used during the flight test to collect the LTE radio parameters and store them in text file

# GPS data: latitude and longitude

# Serving cell: the cell which currently provide the coverage and the service (MCC, MNC, PCI, EARFCN, Cell ID, RSRP, RSRQ, RSSI, SINR)

# NB cell: the cell is potential the user will get the service in case the service from the serving cell degrades (EARFCN, PCI, RSRQ, RSRP, RSSI)

# System info: to get date and time

==============================================================

Meaning of the above parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Stands for | More info | Unit |
| latitude | latitude | Positioning | N/A |
| longitude | longitude | Positioning | N/A |
| MCC | Mobile Country Code | A Code assigned per country to avoid confusion | N/A |
| MNC | Mobile Network Code | A Code assigned per operator by a regulatory of a country | N/A |
| PCI | Physical Cell ID | An ID assigned per cells to avoid internal interference and can be reused in a network | N/A |
| EARFCN | Evolved Absolute Radio Frequency Channel Number | a number given to channels to organize and to avoid inter-operator interference | N/A |
| Cell ID | Cell ID | An ID assigned per cell and can’t be reused in network | N/A |
| LAC | Location Area Code | A logical ID given to group of cells to speed up service delivery process | N/A |
| RSRP | Reference Signal Received Power | The power of LTE signal received by the pMLTE or mobile | dBm |
| RSRQ | Reference Signal Received Quality | The Quality of LTE signal received by the pMLTE or mobile | dB |
| RSSI | Received Signal Strength Indicator | The strength of the signal received by pMLTE or mobile for a certain channel | dB |
| SINR | Signal-to-interference + Noise Ratio | This ratio can refer to the service quality | dB |

**LTE\_map\_during\_test.py:** this python script used during the flight test to map some of the LTE radio parameters in real time using Python map libraries and auto update during the flight test and export these maps to html file, this script run when there is new data obtained from the pMLTE.

**LTE\_map\_post\_test.py:** this Python script function just like **LTE\_map\_during\_test.py** script but can be used for any old log flight test run once on whole log file the difference between this script and the previous one, the previous one check the update on the LTE log file if there is no update it will stop processing which this case not happening in the old log file

**LTE\_data\_extracting\_during\_test.py:** this python script used during the flight test to get the LTE parameters in real time (since in the log file contains many texts) and arrange them in CSV text file format to be ready for mapping and plot some statistics, this script run when there is new data obtained from the pMLTE.

**LTE\_data\_extracting\_post\_test.py:** this Python script function just like **LTE\_data\_extracting\_during\_test.py** script but can be used for any old log flight test run once on whole log file the difference between this script and the previous one, the previous one check the update on the LTE log file if there is no update it will stop processing which this case not happening in the old log file.

**NW\_delay\_logging\_Nping.py:** this python script is run during the flight test to get the network end-to-end delay time in milli seconds using Nping tool and write the results into text file.

**NW\_delay\_data\_extract\_Nping.py:** this python script is run after the flight test to get the End-to-End delay parameters in milliseconds from the log file to a comma separated value text file.

**NW\_speed\_logging\_Iperf3.py:** this python script runs during the flight test to get the network end-to-end throughput UL and DL in Mbps using the Iperf3 protocol and write the results into text files.

**NW\_speed\_data\_extract\_Iperf3.py:** this python script used after the flight test to get the End-to-End throughput in both directions Uplink and Downlink in Mbps from the log file to a comma separated value text file.

**Statistics\_of\_test\_area.py:** this python script used after the flight test to get some statistics about the area tested like CDF (Cumulative Distribution Function), PDF (Probability Distribution Function), Mean, Max, and Min of LTE connection values.

**CDF\_UL\_DL.py:** this python script plots a statistical chart of Cumulative Distribution Function CDF of Uplink and Downlink throughput to give an overview of this metric.

**3d\_trajectory\_map.py:** this Python script plots a 3d trajectory of the LTE radio metrics in a space to present the parameters with respect to the position and altitude.

**RSRP\_by\_altitude.py:** this Python script plots a statistical chart of Cumulative Distribution Function CDF of mean and Standard Deviation of the LTE radio metrics with respect to the altitude since the flight test is performed in different altitudes