Loon Corona Current Dataset README

Authors

Beth Reid[1], Tom Swanson[1], Ryan Endacott[1], Jared Bevis[1], Bradley Rhodes[1], Rob Carver[1], Paul Heninwolf[1], Jim Olivo[2], Cathy Kessinger[3]
[1]Loon LLC, [2]BCI, [3]NCAR

Introduction

After observing damage to flight vehicles under suspected electrical activity, Loon added a number of sensors to characterize the flight environment in order to ensure safe operation of Loon's flight systems. After initial data collection, we found that the most interesting and interpretable sensor was the corona current detector. This dataset provides data from that sensor collected over 794,000 flight hours on 252 unique flights.

Description of data

The corona current detector measured the corona current that flowed off the end of a long wire dangled from Loon's payload, which was designed to be a focal point for the formation of corona discharge when the balloon is in the presence of a large, static electric field. The circuit records a maximum current amplitude of 10 microamps. Our dataset consists of events above a threshold of 0.1 microamps. This lower threshold was chosen to be slightly larger than the typical noise and manufacturing tolerance errors that we saw from this sensor in the absence of storms or electrical activity. Positive current flows vertically up (away from Earth's surface).

We expected to see occasional readings on this sensor when we were over particularly large storms. These were expected to be small sustained average readings (several microamps for an hour or so). We also expected to see occasional large pulses due to lightning streamers attaching to the vehicle, but these were thought to be infrequent. One challenge is that we seem to experience far more corona discharge and lightning streamers than anticipated; this prevents us from estimating the average corona current (and therefore the average electric field strength); we made this inference by examining a small set of higher frequency observations not part of this data release.

This sensor became an excellent indicator that a flight vehicle was over a storm with electrical activity. Unfortunately, this sensor frequently gets maxed out due to the streamers, and it's difficult to differentiate between constant corona discharge and streamers (at least at the standard telemetry rates).

Primary and secondary telemetry concepts

When balloons were not connected to a ground station, Loon received real-time information about the balloon's location and internal state through satellite providers. To conserve bandwidth, Loon made use of “critical”, "primary", and "secondary" telemetry categories. Critical fields were sent with every telemetry message, even if unchanged. Primary fields were sent on change in every message. Secondary fields were sent on change if there was space in the message, chosen in a round-robin system. Therefore there was no guarantee on timeliness of delivery, but 30 minutes was a pretty good worst-case window under typical operation. Primary fields were typically received and logged approximately every minute, with secondary fields every 15-20 minutes. Various latencies or interruptions could temporarily increase the spacing between messages. Initially data from the corona current detector was reported as a secondary field, and the min, max, and mean current in amps was reported for the 30 minutes prior to the report. Note that when connected to a ground station (either directly or through a network of Loon balloons), Loon collected “high rate telemetry”, where we received new telemetry messages even more frequently.

Loon Navigation Algorithm’s impact on flight path sampling

For researchers undertaking statistical analysis of this dataset, we caution that Loon actively navigated away from or “over” regions of electrical activity indicated by CDO >= 2.5 and/or CloudHeights within 5000 feet, with a buffer of up to 100 km around stormy regions and “over” typically reaching at least 60,000 feet when near storms. This means that the flight’s 3d position (latitude, longitude, pressure altitude) can not be treated as independent of storm conditions (specifically, flights should remain at >= 60,000 feet within 100km of cdo >= 2.5 and at least 5000 feet above BCI CloudHeight values).

Background flight data columns

This dataset contains 9.5 million rows covering flights from Feb 2019 to Apr 2021. Several fields (cdo* and cth*) are derived from the real-time aviation layers that Loon purchased from BCI. Two fields (glm*) are derived directly from the GLM Level 2 Lightning Detection data. We derive average event density and optical flux directly from the event_lat, event_lon, event_time_offset, event_energy fields in order to retain the native spatial resolution of the dataset. Group and flash relationships were not analyzed.

There are two “missing data” indicators used for both external datasets. -9999 indicates that observations for the timestamp of interest were unavailable, usually due to short outages in the external dataset. -8888 indicates that the flight location was outside the coverage region of the external dataset. More than half of the flight data is outside the GLM coverage region surrounding North and South America. The BCI data covers latitudes from -50 to 80 degrees; occasionally Loon flights travel outside this coverage band.

- **flight_id**: unique identifier for flight; each Loon vehicle performs only one flight. This field should be matched to the flight_id column in the primary and secondary datasets.
- **date_str**: UTC ISO 8601 timestamp string, where the timestamp is the center of the evenly spaced 5 minute time window from which telemetry was selected. To align background data with primary and secondary events, merge rows on both flight_id and the floor of this field, applied on a 5-minute time interval, e.g. 23:34:56 → 23:30:00). The same flooring function should be applied to the date_str column in the primary and secondary datasets, e.g.

```
df = pd.read_csv('background_storm_trooper_dataset_v2.csv', index_col=0)
df['date'] = pd.to_datetime(df['date_str'])
df['floored_date'] = df['date'].dt.floor(pd.Timedelta(5, 'min'))
```

- **latitude**: GPS flight latitude in degrees
- **longitude**: GPS flight longitude in degrees
- **pressure altitude**: flight pressure level in Pascals
- **cdo**: BCI CDO value at the pixel nearest to the flight location. Note that the raw CDO and CloudHeight data provided by BCI are on a uniform latitude/longitude grid with spacing of 0.06 degrees in each dimension.
- **cth**: BCI CloudHeight value at the pixel nearest to the flight location. Units are in Pascals, converted from the raw BCI data (in feet) using the standard atmosphere.
- **glm_event_density**: GLM event rate density in units of events per km^2 per day, pixelized onto a 0.1 x 0.1 deg^2 grid in 5 minute intervals. Note that GLM has a resolution of 8km at NADIR and more typically ~10km, so this grid choice approximately matches the raw data resolution, but because the satellite pixels are not in a regular lat/lng grid, this means that our gridding scheme will result in some pixels with no GLM events. Averaging or accumulating over longer flight data time windows (e.g. 30 minutes when comparing with secondary telemetry data) mitigates this problem because the flight trajectory will intersect multiple spatial pixels in the time window.
- **glm_optical_flux**: GLM optical flux in Watts per m^2, pixelized onto the same grid as **glm_event_density**.
- **cdo025 (cth025)**: same as cdo (cth), but reporting the maximum (minimum) value within 25km.
- **cth025_diff_feet**: Difference between the flight height and cth025 (converted back to a height in feet), using standard atmosphere.
- **cdo_dist_km:** Minimum distance in km between nearest BCI pixel to flight and BCI pixel with CDO >= 2.5 (our navigational threshold for “electrical activity”).

Primary stormtrooper flight data columns

This dataset consists of 48582 primary telemetry messages containing the corona_above_threshold_count field. Reports suspected of out-of-order telemetry, diagnosed by a brief corona_above_threshold_count deviation from a monotonic increase, were dropped.

Primary fields were only reported after 2019-11-17, after an initial analysis of secondary telemetry allowed us to determine a valid threshold above the noise. Primary telemetry messages were typically received once per minute, but this could vary due to latencies / outages in both receiving and ingesting the messages. When interpreting features / rates from this dataset against the background dataset, the background dataset should also be restricted to timestamps after 2019-11-17.

- **flight_id**: unique identifier for flight; each Loon vehicle performs only one flight.
- **date_str:** UTC ISO 8601 timestamp string corresponding to the primary telemetry message. All messages containing a corona_above_threshold_count field from the storm_trooper corona detector are retained; they have not been downsampled like the background dataset. Note that the primary message should only contain corona_above_threshold_count if it has changed since the last report.
- **corona_above_threshold_count**: Monotonically increasing count of the number of seconds during which the absolute value of the corona current exceeded the threshold 0.1 microamps. Note that this counter does get reset upon board reboots, and so we provide more useful derived fields that account for that.
- **cnts_per_window:** Above threshold count associated with the current timestamp, computed by differencing successive **corona_above_threshold_count** messages but correcting for counter resets.
- **threshold_frac:** Given successive telemetry reports with positive cnts_per_window, this field contains the fraction of seconds covered by the message that were above the threshold; this field normalizes the impact of differing durations between messages. In an ideal world, this field would range from 0 to 1. It sometimes exceeds 1 due to mismatch between message timestamp and when the counts were read out. The value is set to NaN if the previous message was longer than 15 minutes in the past, since primary messages are guaranteed at a cadence shorter than that. Without knowing the previous timestamp, threshold_frac cannot be computed.

Secondary stormtrooper flight data columns

Secondary messages were typically received every 15-20 minutes. However, secondary fields were sent in a round-robin system and limited by space available in the message. Therefore there was no guarantee on timeliness of delivery, but 30 minutes was a pretty good worst-case window under typical operation. Therefore max/min/mean currents were computed onboard in a rolling window of size 30 minutes. We only retain reports where the absolute value of the minimum or maximum corona current was >= 0.1 microamps, the same threshold used in the primary dataset to register a change in the corona_above_threshold_count field. The sign convention for the reported current is such that positive current flows vertically up (away from Earth's surface).

- **flight_id**: unique identifier for flight; each Loon vehicle performs only one flight.

- **date_str:** UTC ISO 8601 timestamp string corresponding to the secondary telemetry message. All messages meeting the current threshold criterion have been retained; they have not been downsampled like the background dataset.
- **corona_detector_max**: Maximum corona detector current measured over the last 30 minutes in units of Amps.
- **corona_detector_min**: Minimum corona detector current measured over the last 30 minutes in units of Amps.
- **corona_detector_mean**: Average corona detector current measured over the last 30 minutes in units of Amps.