RELAMPAGO LMA Level 3 File Documentation

Level 3 data were created using the lmatools Python package (https://github.com/deeplycloudy/lmatools). See references below for more information about LMA data, processing, and each variable.

Each RELAMPAGO LMA Level 3 file is a netCDF4 architecture. Note that RELAMPAGO LMA data always start (and end) 1 second after each 10-minute period (e.g., 000001, 001001, 002001, etc.). This is an artifact of how the Level 0 data were recorded.

All data were processed with a chisqr maximum of 5, and requiring at least 5 VHF sources to identify a flash. Data are gridded to 1-km resolution grids (all dimensions) and binned every minute. New files are available every 10 minutes. Two-dimensional data structures are (ntimes, lon, lat) and 3D data structures are (ntimes, lon, lat, alt). Number of bins in each dimension are 10 (ntimes) x 400 (lon) x 401 (lat) x 20 (alt)

Info about files

- *_flash_extent.nc & *_flash_extent_3d.nc: Flash extent density (km^{-2} or km^{-3}; i.e., number of flashes that pass through a given pixel/cube in the given time bin)
- *_flash_init.nc & *_flash_init_3d.nc: Number of flashes that initiated within the given pixel/cube and time bin
- *_footprint.nc & *_footprint_3d.nc: Footprints (mean area/volume in km^2/km^3) of flashes that occurred within pixel/cube and time bin
- *_specific_energy.nc & *_specific_energy _3d.nc: Specific energy (mean; J/kg) of flashes that occurred within pixel/cube and time bin
- *_total_energy.nc & *_total_energy _3d.nc: Total energy (J) of flashes that occurred within pixel/cube and time bin
- *_source.nc & *_source_3d.nc: Number of sources (km^{-2} or km^{-3}) that occurred in grid element
- *_flashsize_std.nc & *_flashsize_std_3d.nc: Standard deviation (km^2 or km^3) of flash sizes within grid element

Each file also contains dimensional variables such as time, longitude, latitude, and altitude.

In Python, you can retrieve level 3 data into numpy arrays using the following example code snippet:

```
import numpy as np
import xarray
data = xarray.open_dataset('REL_20181214_230001_600_5src_0.0105deg-
dx_flash_extent.nc')
lon = np.array(data.lon.data)
lat = np.array(data.lat.data)
bintime = np.array(data.time.data)
flash_extent = np.array(data.flash_extent.data)
data.close()
```

References

Bruning, E.C. and D.R. MacGorman, 2013: Theory and Observations of Controls on Lightning Flash Size Spectra. J. Atmos. Sci., 70, 4012-4029, https://doi.org/10.1175/JAS-D-12-0289.1

Chmielewski, V. C., and Bruning, E. C. (2016), Lightning Mapping Array flash detection performance with variable receiver thresholds, J. Geophys. Res. Atmos., 121, 8600- 8614, doi:10.1002/2016JD025159.

Fuchs, B. R., Bruning, E. C., Rutledge, S. A., Carey, L. D., Krehbiel, P. R., & Rison, W. (2016). Climatological analyses of LMA data with an open-source lightning clustering algorithm. Journal of Geophysical Research: Atmospheres, 121, 8625-8648. https://doi.org/10.1002/2015JD024663

Rison, W., R. Thomas, P. Krehbiel, T. Hamlin, and J. Harlin (1999), A GPS-based three-dimensional lightning mapping system: Initial observations in central New Mexico, Geophys. Res. Lett., 26, 3573-3576.

Thomas, R. J., Krehbiel, P. R., Rison, W., Hunyady, S. J., Winn, W. P., Hamlin, T., & Harlin, J. (2004). Accuracy of the lightning mapping array. Journal of Geophysical Research, 109, D14207. https://doi.org/10.1029/2004JD004549