CNES Falcon Navigation and State Parameters

Viltard, Nicolas* Nicolas.Viltard@latmos.ipsl.fr

Scientist, Laboratoire Atmosphères, Milieux, Observations Spatiales (LATMOS)

Kerns, Brandon W. bkerns@uw.edu, ORCID 0000-0001-5691-7895

Senior Meteorologist, University of Washington Applied Physics Laboratory
1013 NE 40 ST, BOX 355640, Seattle, Washington, 98105-6698.

*Principal Investigator
^Prepared this document

1. Data Set Overview:

“The [NOAA P-3 and French Falcon] aircraft missions aimed to address three main science objectives of DYNAMO to better understand 1) multiscale convection–environment interactions, 2) water vapor variability and three dimensional (3D) dynamical and microphysical structure in convective cloud systems, and 3) air–sea fluxes and boundary layer structure in the MJO initiation over the Indian Ocean” [Chen et al., 2016]. The Falcon mission was also the second Megha-Tropiques validation/preparation field program funded by CNES. This document describes the flight level data. For radar and microphysical data, contact CNES.

Time period: 11 missions during 22 November – 14 December 2011

Physical location: The aircraft was based out of Gan airport, Addu Atoll, Maldives. The missions were conducted within the following geographical area centered around Gan: 4.8°S – 1.7°N, 71.3°E – 76.0°E.

Data source: Data were collected with the CNES Falcon navigation and data collection system.

2. Instrument Description

Data were collected by the onboard CNES Falcon navigation and data collection system.

3. Data Collection and Processing

The data are the original files provided by CNES, without any additional quality control.

4. Data Format

The DATA are in CF compliant NetCDF format. There is one file for each day named **aircraft.gan.falcon.flightlevel.YYYYMMDD.nc, w**here **YYYY** is the 4-digit year, **MM** is the two-digit month, and **DD** is the two-digit day. Data frequency is 1 Hz.

Below is a list of the parameters with units and attributes from ncdump:

```
dimensions:
        time = UNLIMITED ; // (7585 currently)
variables:
        float time(time) ;
                time:units = "seconds since 2011-12-12 00:00:00" ;
                time:calendar = "gregorian" ;
        float event_marker(time) ;
                event_marker:long_name = "Event marker" ;
                event_marker:standard_name = "" ;
                event_marker:units = "count" ;
                event_marker:comment = "" ;
        float latitude(time) ;
                latitude:long_name = "Latitude" ;
                latitude:standard_name = "latitude" ;
                latitude:units = "degree_north" ;
                latitude:comment = "from GPS" ;
        float longitude(time) ;
                longitude:long_name = "Longitude" ;
                longitude:standard_name = "longitude" ;
                longitude:units = "degree_east" ;
                longitude:comment = "from GPS" ;
        float altitude_GPS(time) ;
                altitude_GPS:long_name = "Altitude from GPS" ;
                altitude_GPS:standard_name = "altitude" ;
                altitude_GPS:units = "m" ;
                altitude_GPS:comment = "from GPS" ;
        float altitude_INS(time) ;
                altitude_INS:long_name = "Altitude from INS" ;
                altitude_INS:standard_name = "altitude" ;
                altitude_INS:units = "m" ;
                altitude_INS:comment = "from INS" ;
        float platform_roll_angle(time) ;
                platform_roll_angle:long_name = "Platform roll angle" ;
                platform_roll_angle:standard_name = "platform_roll_angle" ;
                platform_roll_angle:units = "degree" ;
                platform_roll_angle:comment = "from INS" ;
        float platform_pitch_angle(time) ;
                platform_pitch_angle:long_name = "Platform pitch angle" ;
                platform_pitch_angle:standard_name = "platform_pitch_angle" ;
                platform_pitch_angle:units = "degree" ;
                platform_pitch_angle:comment = "from INS" ;
        float platform_orientation(time) ;
                platform_orientation:long_name = "Platform orientation" ;
                platform_orientation:standard_name = "platform_orientation" ;
                platform_orientation:units = "degree" ;
                platform_orientation:comment = "from INS" ;
        float air_pressure(time) ;
                air_pressure:long_name = "Air pressure" ;
                air_pressure:standard_name = "air_pressure" ;
                air_pressure:units = "hPa" ;
                air_pressure:comment = "From front sensor, corrected for the so-called
static defect" ;
        float air_temperature(time) ;
                air_temperature:long_name = "Air temperature" ;
                air_temperature:standard_name = "air_temperature" ;
                air_temperature:units = "degree_Celsius" ;
                air_temperature:comment = "from deiced Rosemount sensor" ;
        float dew_point_temperature(time) ;
                dew_point_temperature:long_name = "Dew point temperature" ;
                dew_point_temperature:standard_name = "dew_point_temperature" ;
                dew_point_temperature:units = "degree_Celsius" ;
                dew_point_temperature:comment = "from 1011B top dew-point hygrometer"
;
        float relative_humidity(time) ;
                relative_humidity:long_name = "Relative humidity (%)" ;
                relative_humidity:standard_name = "relative_humidity" ;
                relative_humidity:units = "1" ;
                relative_humidity:comment = "from Aerodata sensor" ;
        float humidity_mixing_ratio_GE1011B(time) ;
                humidity_mixing_ratio_GE1011B:long_name = "Humidity mixing ratio
(g/kg) from GE1011B" ;
                humidity_mixing_ratio_GE1011B:standard_name = "humidity_mixing_ratio"
;
                humidity_mixing_ratio_GE1011B:units = "1" ;
                humidity_mixing_ratio_GE1011B:comment = "from top dew-point hygrometer
(GE 1011B)" ;
        float humidity_mixing_ratio_Aerodata(time) ;
                humidity_mixing_ratio_Aerodata:long_name = "Humidity mixing ratio
(g/kg) from Aerodata" ;
                humidity_mixing_ratio_Aerodata:standard_name = "humidity_mixing_ratio"
;
                humidity_mixing_ratio_Aerodata:units = "1" ;
                humidity_mixing_ratio_Aerodata:comment = "from Aerodata sensor" ;
        float platform_speed_wrt_air(time) ;
                platform_speed_wrt_air:long_name = "Air speed" ;
                platform_speed_wrt_air:standard_name = "platform_speed_wrt_air" ;
                platform_speed_wrt_air:units = "m s-1" ;
                platform_speed_wrt_air:comment = "from pitot" ;
        float platform_acceleration_along_vertical_axis(time) ;
                platform_acceleration_along_vertical_axis:long_name = "Acceleration
along vertical axis" ;
                platform_acceleration_along_vertical_axis:standard_name = "" ;
                platform_acceleration_along_vertical_axis:units = "m s-2" ;
                platform_acceleration_along_vertical_axis:comment = "from INS" ;
        float platform_course_INS(time) ;
                platform_course_INS:long_name = "Course from INS" ;
                platform_course_INS:standard_name = "platform_course" ;
                platform_course_INS:units = "degree" ;
                platform_course_INS:comment = "from INS" ;
        float platform_speed_wrt_ground_INS(time) ;
                platform_speed_wrt_ground_INS:long_name = "Ground speed from INS" ;
                platform_speed_wrt_ground_INS:standard_name =
"platform_speed_wrt_ground" ;
                platform_speed_wrt_ground_INS:units = "kt" ;
                platform_speed_wrt_ground_INS:comment = "from INS" ;
        float platform_course_GPS(time) ;
                platform_course_GPS:long_name = "Course from GPS" ;
                platform_course_GPS:standard_name = "platform_course" ;
                platform_course_GPS:units = "degree" ;
                platform_course_GPS:comment = "from GPS" ;
        float platform_speed_wrt_ground_GPS(time) ;
                platform_speed_wrt_ground_GPS:long_name = "Ground speed from GPS" ;
                platform_speed_wrt_ground_GPS:standard_name =
"platform_speed_wrt_ground" ;
                platform_speed_wrt_ground_GPS:units = "m s-1" ;
                platform_speed_wrt_ground_GPS:comment = "from GPS" ;
        float upward_platform_speed_wrt_ground_INS(time) ;
                upward_platform_speed_wrt_ground_INS:long_name = "Upward ground speed
from INS" ;
                upward_platform_speed_wrt_ground_INS:standard_name = "" ;
                upward_platform_speed_wrt_ground_INS:units = "m s-1" ;
                upward_platform_speed_wrt_ground_INS:comment = "from GPS" ;
        float upward_platform_speed_wrt_ground_GPS(time) ;
                upward_platform_speed_wrt_ground_GPS:long_name = "Upward ground speed
from GPS" ;
                upward_platform_speed_wrt_ground_GPS:standard_name = "" ;
                upward_platform_speed_wrt_ground_GPS:units = "m s-1" ;
                upward_platform_speed_wrt_ground_GPS:comment = "from GPS" ;
        float angle_of_attack(time) ;
                angle_of_attack:long_name = "Angle of attack" ;
                angle_of_attack:standard_name = "" ;
                angle_of_attack:units = "degree" ;
                angle_of_attack:comment = "from sensor on the boom" ;
        float angle_of_sideslip(time) ;
                angle_of_sideslip:long_name = "Angle of sideslip" ;
                angle_of_sideslip:standard_name = "" ;
                angle_of_sideslip:units = "degree" ;
                angle_of_sideslip:comment = "from sensor on the boom" ;
        float eastward_wind(time) ;
                eastward_wind:long_name = "Eastward wind" ;
                eastward_wind:standard_name = "eastward_wind" ;
                eastward_wind:units = "m s-1" ;
                eastward_wind:comment = "Attitudes and speed wrt ground from INS, air
angles from radome, air speed from pitot (m/s)" ;
        float northward_wind(time) ;
                northward_wind:long_name = "Northward wind" ;
                northward_wind:standard_name = "northward_wind" ;
                northward_wind:units = "m s-1" ;
                northward_wind:comment = "Attitudes and speed wrt ground from INS, air
angles from radome, air speed from pitot (m/s)" ;
        float upward_air_velocity(time) ;
                upward_air_velocity:long_name = "Upward air velocity" ;
                upward_air_velocity:standard_name = "upward_air_velocity" ;
                upward_air_velocity:units = "m s-1" ;
                upward_air_velocity:comment = "Attitudes and speed wrt ground from
INS, air angles from radome, air speed from pitot (m/s)" ;
        float wind_from_direction(time) ;
                wind_from_direction:long_name = "Wind from direction" ;
                wind_from_direction:standard_name = "wind_from_direction" ;
                wind_from_direction:units = "degree" ;
                wind_from_direction:comment = "Attitudes and speed wrt ground from
INS, air angles from radome, air speed from pitot (m/s)" ;
        float wind_speed(time) ;
                wind_speed:long_name = "Wind speed" ;
                wind_speed:standard_name = "wind_speed" ;
                wind_speed:units = "m s-1" ;
                wind_speed:comment = "Attitudes and speed wrt ground from INS, air
angles from radome, air speed from pitot (m/s)" ;
        float icing_detector_raw_output(time) ;
                icing_detector_raw_output:long_name = "Icing detector raw output" ;
                icing_detector_raw_output:standard_name = "" ;
                icing_detector_raw_output:units = "volt" ;
                icing_detector_raw_output:comment = "from Rosemount Icing Detector,
model 871 FA" ;

// global attributes:
                :Conventions = "CF-1.6" ;
                :title = "Falcon Dynamo Legacy Data 2011-12-12 00:00:00" ;
                :institution = "SAFIRE" ;
                :description = "SAFIRE Falcon 20 Dynamo Legacy Data with CF1.6
Standard Conventions" ;
                :history = "\nJune 23, 2016: Original python script for Falcon
aircraft NetCDF file created.\nJuly 7, 2016: Script was heavily edited: Description,
source, comments, and variable attributes were changed (or added, if missing). Code
was modified to be more concise.\nJuly 14, 2016: Added CF compliant standard names,
units to dataset vars. Added more comments to address things that need to be fixed.\n"
;
                :source = "Falcon 20 aircraft" ;
                :references = "" ;
                :comment = "\nNot complete!\n\nCF-1.6 compliant NetCDF file produced
at the Rosenstiel School of Marine and Atmospheric Science, University of
Miami.\n\nAuthor notes:\n\nNeed references attribute.\n\nNot sure about dataset
institution. The aircraft is operated by SAFIRE (Service des Avions Francais
Instrumentes pour la Recherche en Environnement) but DYNAMO is a UCAR/NCAR
operation?\n\nNot all variables have standard names. The following are lacking
standard names:\n\"time\", \"event_marker\",
\"platform_acceleration_along_vertical_axis\",
\"upward_platform_speed_wrt_ground_GPS\", \"upward_platform_speed_wrt_ground_GPS\",
\"angle_of_attack\", \"angle_of_sideslip\", \"icing_detector_raw_output\".\nStandard
names (except for the variables above) were taken directly from the text file
containing the data. NOTE that some variables share standard names; I believe these
are only for the navigational data (altitude, course, etc.), which differ by
instrument (GPS/INS).\n\nContact for questions or issues with the file:
jyg3@miami.edu\n" ;
```

5. Data Remarks

The data can be accessed using the myriad of software that is able to interact with NetCDF format files, including ncdump, ncview, Matlab, Python, IDL, and NCL. Consult the user help system within each software package.

6. References

Chen, S. S., B. W. Kerns, N. Guy, D. P. Jorgensen, J. Delanoë, N. Viltard, C. Zappa, F. Judt, C.-Y. Lee, and A. Savarin, 2016: Aircraft observations of dry air, ITCZ, convective cloud systems and cold pools in MJO during DYNAMO: Bull. Amer. Meteor. Soc., 97, 405-423. doi:10.1175/BAMS-D-13-00196.1