Product User Manual for the Blended TROPOMI+GOSAT Product

Nicholas Balasus

20 June 2023

1) Introduction

The blended TROPOMI+GOSAT files are formed by applying the correction described in Balasus et al. (2023) to the variable *methane_mixing_ratio_bias_corrected* in the RPRO (until 17 July 2022) or OFFL (after 17 July 2022) TROPOMI files. This adds the new variable of *methane_mixing_ratio_blended*. The correction is applicable to the files with a processor version of 02.04.00 or 02.05.00 (these processors are identical for the methane files). The following sections describe downloading the data, the file names, the file contents, and how to plot the file contents using Python.

2) Downloading the data

The data is located at https://dataverse.harvard.edu/dataverse/blended-tropomi-gosat-methane and is updated monthly (and thus extends from April 2018 to present). The data is organized into folders approximately by month (such as 2018-4 and 2018-05) although acquiring a full month of data might require downloading the first file from the next month and the last file from the previous month as orbits cross months. For each month, a ZIP of the orbit files for the entire month can be downloaded (typically about 440 netCDF files and 3 GB) or a subset of individual netCDF files can be downloaded.

If you want to download all of the data, the following bash script can help you do that efficiently.

```
#!/bin/bash

# One key per month of data on Harvard Dataverse
dois=(
    OTQ78I VCGEVY DXNAEY IP3PKM 7SFWOK DNLHLR 7XHJX7 VP3UUH 35RSXD 9LOKEM IJBQCZ
    QLZIIM JEUCDK SJAN8E V5ERMW DGMWR4 CLOY0S QC1IR6 T3ZAB5 GYNVC3 JTKDG2 ZHOEST
    M0ZLJX RQLRD8 BZ4AY2 ALNDUA LN9TJT BTKMP3 FGPH1J 7SQHVA EJOEK1 ED9NMV IQPUAZ
    PNIO7E WWGJIX 36YTAV GAFID2 42SPOR PIEEBJ ZSMI0B 77DRTT 8SUSYO CAGF2L CGGKLQ
    IC2RTG S70QWJ 3WCX2J KY4YU6 G1FLCR PONQIL U99JT7 1JBLWH PBGUNV LNWRGR FV7LIK
    IDXDNF AFZ3EL 4BWBGG 0UDBBI U6M16S USE40U EO8XFE
)

for doi in "${dois[@]}"; do
    wget --tries=0 -O "${doi}.zip"
"https://dataverse.harvard.edu/api/access/dataset/:persistentId?persistentId=doi:10.7910/DVN/${doi}" && unzip "${doi}.zip" && rm "${doi}.zip"
done
```

3) File Names

An example file name is:

S5P_BLND_L2__CH4____20191215T112041_20191215T130211_11252_03_020400_20230614T125420.nc

S5P -> Sentinel-5 Precursor, the mission name
BLND -> indicates the blended TROPOMI+GOSAT product
L2 -> level 2 data (i.e., individual retrievals)
CH4 -> methane files
20191215T112041 -> start time of the granule in UTC
20191215T130211 -> end time of the granule in UTC
11252 -> the orbit number
03 -> collection number
020400 -> processor version
20230614T125420 -> file generation time in UTC

The only differences between the original operational file name (below) and the blended TROPOMI+GOSAT file name is the replacement of the processing type (RPRO/reprocessed or OFFL/offline) with BLND and the overwriting of the file generation time.

S5P_RPRO_L2__CH4____20191215T112041_20191215T130211_11252_03_020400_20221119T003414.nc

4) File Contents

The files contain the subset of soundings from the OFFL/RPRO files that had a *qa_value* == 1.0. All other observations are excluded because the blended TROPOMI+GOSAT correction is not valid to apply to them. Additionally, only a subset of variables is included in the BLND files (and they no longer reside in netCDF groups). The files have dimensions of *nobs*, *layer*, and *corner*. All variables are the same as described in the TROPOMI methane PUM except for *methane_mixing_ratio_blended* (derived in this work) and *time_utc* (expanded to all observations in these files). The following is an example for the orbit number 11252 file listed above.

<class 'netCDF4._netCDF4.Dataset'>
root group (NETCDF4 data model, file format HDF5):
Title: Blended TROPOMI+GOSAT Methane Product
Contact: Nicholas Balasus (nicholasbalasus@g.harvard.edu)
dimensions(sizes): nobs(136308), layer(12), corner(4)
variables(dimensions): uint8 qa_value(nobs), float32 latitude(nobs), float32 longitude(nobs), float32 methane_mixing_ratio(nobs), float32 methane_mixing_ratio_precision(nobs), float32 methane_mixing_ratio_bias_corrected(nobs), <class 'str'> time_utc(nobs), float32

latitude_bounds(nobs, corner), float32 longitude_bounds(nobs, corner), float32 chi_square_SWIR(nobs), float32 surface_albedo_SWIR(nobs), float32 surface_albedo_NIR(nobs), float32 surface_albedo_SWIR_precision(nobs), float32 surface_albedo_NIR_precision(nobs), float32 aerosol_size(nobs), float32 aerosol_size_precision(nobs), float32 column_averaging_kernel(nobs, layer), float32 surface_altitude(nobs), float32 surface_altitude_precision(nobs), uint8 surface_classification(nobs), float32 surface_pressure(nobs), float32 pressure_interval(nobs), float32 reflectance_cirrus_VIIRS_SWIR(nobs), float32 methane_profile_apriori(nobs, layer), float32 dry_air_subcolumns(nobs, layer), float32 methane_mixing_ratio_blended(nobs)

groups:

5) Plotting

Included here is a simple example for how to plot the data in Python. The pixels are shown as their full extents (using latitude and longitude bounds), though a scatter plot of the latitude and longitude centers can be used for a quick look.

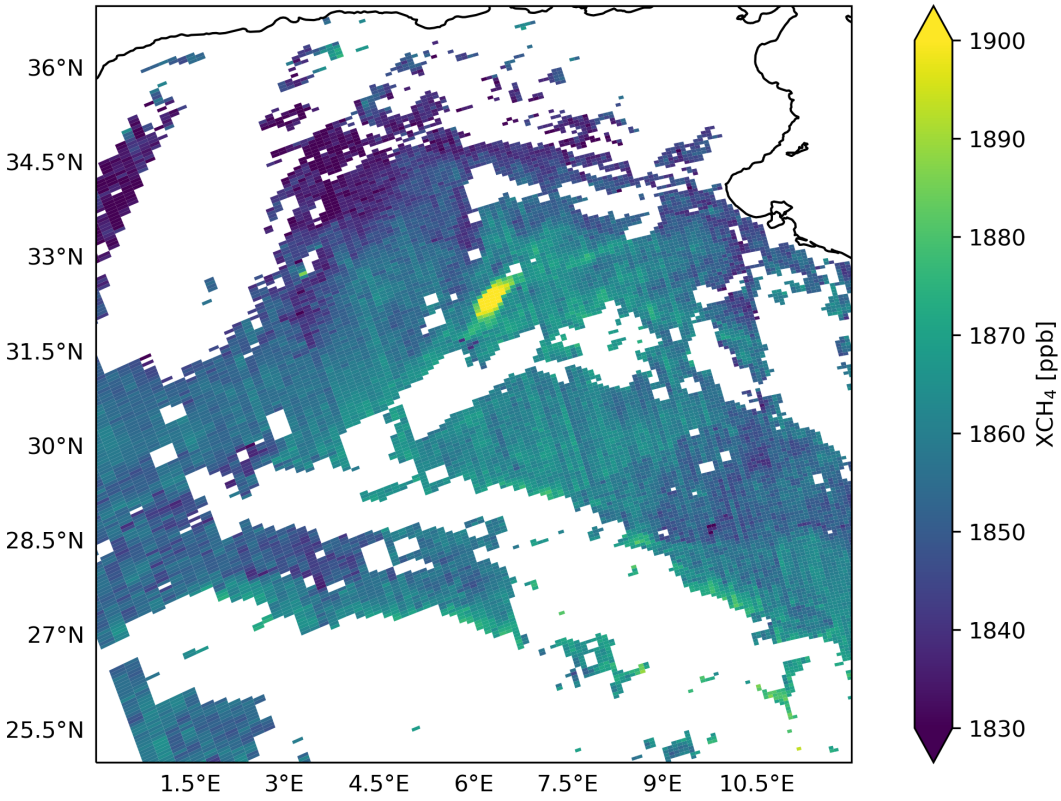
```
import matplotlib.pyplot as plt
import numpy as np
import cartopy.crs as ccrs
from matplotlib.collections import PolyCollection

fig, ax = plt.subplots(figsize=(10, 6), subplot_kw={"projection": ccrs.PlateCarree()})

dataset_file =
"S5P_BLND_L2__CH4____20191215T112041_20191215T130211_11252_03_020400_20230614T125420.nc"
with Dataset(dataset_file) as ds:
    lon_bounds = ds["longitude_bounds"][:]
    lat_bounds = ds["latitude_bounds"][:]
    xch4_blended = ds["methane_mixing_ratio_blended"][:]

verts = [np.column_stack((lon_bounds[i, :], lat_bounds[i, :])) for i in range(len(xch4_blended))]
pc = ax.add_collection(PolyCollection(verts, array=xch4_blended))
pc.set_clim(vmin=1830, vmax=1900)

ax.coastlines()
ax.set_extent([0, 12, 25, 37])
gl = ax.gridlines(draw_labels=True, zorder=-1, linewidth=0)
gl.top_labels = gl.right_labels = False
plt.colorbar(pc, ax=ax, extend="both", label=r"XCH$_4$ [ppb]")
```



6) Contact

If you have questions about the data, please contact Nicholas Balasus at nicholasbalasus@g.harvard.edu.