BErkeley High Resolution (BEHR) NO$_2$ product - User Guide

Josh Laughner

January 13, 2017

1 Reading

1. Russell et al. (2011) - description of original BEHR algorithm

2 Product overview

2.1 Product types

Currently, we have two data products available: one at the native OMI pixel resolution and one in which each swath has been gridded to a 0.05° × 0.05° fixed grid. The gridded product is ideal for users who simply wish to obtain an NO$_2$ VCD, as the latitude and longitude of each grid point will remain fixed over time, whereas the native OMI pixels do not. However, the native OMI resolution files have additional variables compared to the gridded product, such as scattering weights, averaging kernels, our NO$_2$ *a priori* profile, etc. that will be useful to users wishing to modify the product in some way.

Plans to produce a third product with a single, daily grid are underway and we hope to have this available soon.

2.2 Version numbering

The BEHR version numbering system combines the OMNO2 version number with an internal version letter. So, v2.1A represents the first BEHR product based on version 2.1 of the NASA OMNO2 product. A subsequent version of BEHR still based on version 2.1 of OMNO2 would be v2.1B. If OMNO2 were to update to version 3, the BEHR product based on that would be v3.0A.

Should a minor change be made (e.g. one that adds information to the output but does not change the core algorithm), a revision number will be appended to the version number. For example, v2.1A and v2.1Arev0 will be the same, but v2.1Arev1 would indicate this sort of minor change.

The version number may also be formatted as, e.g. v2-1A. This is used as the version string in file names to prevent any issue with file systems unable to handle a . in a filename

that is not separating the file extension. This form is completely equivalent to the one with the period and is used interchangeably.

2.3 File format

All products will be made available as HDF version 5 (`.h5`) files (*c.f.* `https://www.hdfgroup.org`). Please note when trying to open these that many programming languages and utilities have different commands and tools for opening version 4 and 5 HDF files. If you are having trouble opening these files:

1. Ensure that you are using the correct command for an HDF5 file, not an HDF4 file.

2. Try using HDFView (available from `https://www.hdfgroup.org/products/java/index.html` to browse the file. This will confirm that it downloaded properly.

3. Check if the utility or programming language you are using requires the HDF5 library (`https://www.hdfgroup.org/HDF5/`) to be installed.

For both the native and gridded products, the HDF files are organized similarly. Under the `/Data` group, each swath is contained within its own group, named as `Swath#`. There will be 3–5 swaths per day. Each swath will contain all relevant variables as datasets.

Starting with BEHR version 2.1A, fill values will be directly stored in the HDF FillValue information for each dataset, along with four additional attributes: Description, Range, Product, and Unit. *Description* is a brief, one-line description of the meaning of each variable. *Range* is the range of values that variable may correctly take on. *Product* indicates whether this dataset is copied directly from the NASA standard product (represented by **SP**) or is added by the BEHR product (unsurprisingly represented by **BEHR**). Finally, *Unit* is the physical unit assigned to each dataset.

The native product will also be provided as comma separated value files. Because part of the value of the gridded product is the 2-dimensional structure of the grid reflects the geographic distribution, and because this would be difficult to maintain in a .csv file, we will continue to provide the gridded product in HDF format only.

2.4 Tools for working with HDF files

The following programs or programming languages are known to be able to read HDF files:

- MATLAB: current versions have high-level functions such as `h5info` and `h5read` which can easily read in HDF5 files. This does not seem to rely on the external HDF library.

- Python: the `h5py` package (`http://www.h5py.org`) can read HDF5 files, however it does depend on having the HDF library installed, at least on Unix based systems.

- IDL: various users have sucessfully read HDF5 files in IDL; however since we do not use it ourselves, we cannot offer specific advice on the best way to do so.

- Igor Pro, v. > 5.04: `http://www.wavemetrics.com/products/igorpro/dataaccess/hdf5.htm`

- GNU Octave: `https://www.gnu.org/software/octave/`

This is not an exclusive list, however these are common scientific software packages that indicate they have the capability to read HDF5 files. Note that our experience is focused on MATLAB and Python, so our ability to offer specific advice for other utilities is limited.

3 Variables

Table 1 will list the attributes of all variables found in the BEHR files. Most categories are fairly self-explanatory. *Product* indicates whether the variable is directly copied from the NASA OMNO2 product (SP) or calculated from BEHR. *Gridding* indicates whether the variable will be contained in the native OMI pixel resolution files ("Native") or both the native and gridded files ("both").

For more information on the SP variables, see the links at `http://disc.sci.gsfc.nasa.gov/Aura/data-holdings/OMI/omno2_v003.shtml`, especially the Readme and OMNO2 Data Format. Here we will describe primarily the BEHR variables in detail, although we will describe the presentation of some OMNO2 variables where necessary.

- **BEHRAMFTrop**: the tropospheric AMF calculated by the BEHR algorithm using high resolution albedo, terrain pressure, and NO_2 *a priori* inputs. This AMF estimates a ghost column as well, and corresponds to the VCD in the BEHRColumnAmountNO2Trop field.

- **BEHRAMFTropVisOnly**: the tropospheric AMF calculated by the BEHR algorithm that does not attempt to estimate a ghost column. This corresponds to the VCD in BEHRColumnAmountNO2VisOnly.

- **BEHRAvgKernels**: a vector of averaging kernels that can be used when comparing model output to the BEHR product. See §4.4 for details.

- **BEHRColumnAmountNO2Trop**: the tropospheric NO_2 column calculated using the BEHR algorithm as $V_{\mathrm{BEHR}} = V_{\mathrm{NASA}} \cdot A_{\mathrm{trop, NASA}}/A_{\mathrm{trop, BEHR}}$, where V_{BEHR} and V_{NASA} are the vertical column densities for BEHR and NASA respectively, and $A_{\mathrm{trop, NASA}}$ and $A_{\mathrm{trop, BEHR}}$ are the tropospheric AMFs. This field includes estimated ghost columns that are obscured by clouds. The estimate essentially scales the above-cloud component by the ratio of the above-cloud to total column from the model *a priori* profile.

- **BEHRColumnAmountNO2TropVisOnly**: the tropospheric NO_2 column calculated using the BEHR algorithm and the same equation as for the previous variable, except that the visible-only AMF is used instead. This retrieves only the visible NO_2 column, thus this is the NO_2 column to ground for the clear fraction of the pixel, and only the above-cloud NO_2 column for the cloudy fraction of the pixel.

- **BEHRNO2Apriori**: this is the NO_2 profile used as the *a priori* for the AMF calculations. It is given as unscaled mixing ratios; so multiplying these numbers by the

number density of air will directly give the number density of NO_2. See §4.4 for more information.

- **BEHRPressureLevels**: the pressure level that correspond to the BEHRAvgKernels, BEHRNO2Apriori, and BEHRScatteringWeights vectors. See §4.4 for more information.

- **GLOBETerpress**: The average surface pressure of the pixel calculated from the GLOBE database (`http://www.ngdc.noaa.gov/mgg/topo/globe.html`). Currently it is calculated from the GLOBE terrain heights using a simple scale height calculation, with scale height set to 7.4 km.

- **Latcorn, Loncorn**: corner coordinates of the pixels calculated assuming a 2 second integration period and the sensitivity given as a flat topped Gaussian function, $f(x) = \exp[c(x - x_0)^4]$

- **MODISAlbedo**: The average albedo of the pixel calculated from Band 3 (459–479 nm) of the MCD43C3 combined MODIS albedo product. Because this product is a 16-day average produced every 8 days, the temporally nearest MCD43C3 file is chosen.

- **MODISCloud**: The cloud fraction of the pixel averaged from Aqua MODIS MYD06 cloud fraction data.

- **RelativeAzimuthAngle**: the azimuthal difference between solar and viewing angles, calculated as:

$$x = |\text{SAA} + 180 - \text{VAA}|$$
$$\text{RAA} = \begin{cases} 360 - x \text{ if } x > 180 \\ x \text{ otherwise} \end{cases}$$

- **XTrackQualityFlags**: directly taken from the OMNO2 product; however, while they should always be considered a bit array (and thus integers) in some version of BEHR these are converted to floating point numbers. Also, in the gridded product, in any case where more than one pixel contributed to a grid cell, the value given will be the result of a bitwise OR operation applied to the flags from each pixel. Thus, if a flag is set in any contributing pixel, it will be set in the grid cell.

- **vcdQualityFlags**: same notes as XTrackQualityFlags.

Variable	Gridding	Product	Range	Unit
AMFStrat	Native	SP	$[0, \infty)$	unitless
AMFTrop	Both	SP	$[0, \infty)$	unitless
BEHRAMFTrop	Both	BEHR	$[0, \infty)$	unitless
BEHRAMFTropVisOnly	Both	BEHR	$[0, \infty)$	unitless
BEHRAvgKernels	Native	BEHR	$[0, \infty)$	unitless
BEHRColumnAmountNO2Trop	Both	BEHR	$[0, \infty)$	molec. cm^{-2}
BEHRColumnAmountNO2TropVisOnly	Both	BEHR	$[0, \infty)$	molec. cm^{-2}
BEHRNO2Apriori	Native	BEHR	$(-\infty, \infty)$	unscaled mixing ratio
BEHRPressureLevels	Native	BEHR	$[0, \infty)$	hPa
BEHRScatteringWeights	Native	BEHR	$[0, \infty)$	unitless
CloudFraction	Both	SP	$[0.0, 1.0]$	unitless
CloudPressure	Native	SP	$[0, \infty)$	hPa
CloudRadianceFraction	Both	SP	$[0.0, 1.0]$	unitless
ColumnAmountNO2	Native	SP	$[0, \infty)$	molec. cm^{-2}
ColumnAmountNO2Strat	Native	SP	$[0, \infty)$	molec. cm^{-2}
ColumnAmountNO2Trop	Both	SP	$[0, \infty)$	molec. cm^{-2}
ColumAmountNO2TropStd	Native	SP	$[0, \infty)$	molec. cm^{-2}
GLOBETerpres	Both	BEHR	$[0, \infty)$	hPa
Latcorn	Native	BEHR	[-90.0, 90.0]	degrees
Latitude	Both	SP	[-90.0, 90.0]	degrees
Loncorn	Native	BEHR	[-180.0, 180.0]	degrees
Longitude	Both	SP	[-180.0, 180.0]	degrees
MODISAlbedo	Both	BEHR	[0.0, 1.0]	unitless
MODISCloud	Both	BEHR	[0.0, 1.0]	unitless
RelativeAzimuthAngle	Native	BEHR	[0.0, 180.0]	degrees
Row	Both	SP	[0.0, 59.0]	unitless
SlantColumnAmountNO2	Native	SP	$[0, \infty)$	molec. cm^{-2}
SolarAzimuthAngle	Native	SP	[-180.0, 180.0]	degrees
SolarZenithAngle	Native	SP	[0.0, 90.0]	degrees
Swath	Native	SP	$[0, \infty)$	unitless
TerrainHeight	Native	SP	$[0, \infty)$	hPa
TerrainReflectivity	Native	SP	[0.0, 1.0]	unitless
Time	Native	SP	$[0, \infty)$	s
ViewingAzimuthAngle	Native	SP	[-180.0, 180.0]	degrees
ViewingZenithAngle	Native	SP	[0.0, 90.0]	degrees
XTrackQualityFlags	Both	SP	N/A	bit array flag
vcdQualityFlags	Both	SP	N/A	bit array flag

Table 1: Variables found in the BEHR files.

4 Considerations when working with BEHR files

4.1 Pixel filtering

For users wishing to identify high quality NO_2 column information to ground, there are six key criteria to look for:

1. The vcdQualityFlags field must be an even number. This indicates that the summary bit is not set, meaning there were no significant processing issues.

2. Cloud fraction: BEHR contains three cloud fractions: OMI geometric, OMI radiance, and MODIS cloud fraction. We generally filter for OMI geometric or MODIS fraction to be < 0.2 (20%). As discussed in Russell et al. 2011, MODIS cloud fraction is often less susceptible to identifying high albedo ground surfaces as clouds.

3. The column amount must not be a fill value.

4. Filter for row anomaly, typically by requiring that the XTrackQualityFlags field $= 0$.

A previous version of this document indicated to removing negative VCDs and VCDs $> 1 \times 10^{17}$ molec. cm^{-2}; this is no longer recommended as removing negative VCDs can make the error in the stratospheric subtraction systematic instead of random (especially for small VCDs), and removing large VCDs should not be necessary as long as you check that XTrackQualityFlags $= 0$.

If you see VCDs greater than 1×10^{17} molec. cm^{-2}, check the BEHRAMFTrop field for that pixel; especially in older versions of the product, the AMF may be clamped to 1×10^{-6} as a minimum value in cases where the algorithm encounters a non-fatal error.

4.2 Weighting temporal averaging

When averaging the gridded product over time, use the *Areaweight* variable to weight each contribution. This is calculated for each grid cell as the average of the reciprocals of the areas of each pixel included in that grid cell. Thus, grid cells containing contributions from large pixels are weighted less than those from smaller, more representative pixels.

4.3 Total vs. visible only columns

The field BEHRColumnAmountNO2Trop contains the total column, which includes the estimated ghost column below the cloudy part of the pixel. The field BEHRColumnAmountNO2TropVisOnly does not include the ghost column; only the above-cloud NO_2 is included for the cloudy component of the pixel.

Most users should use the BEHRColumnAmountNO2Trop field, with appropriate filtering for cloud fraction (§4.1). Users interested in cloud slicing approaches should use the BEHRColumnAmountNO2TropVisOnly field.

The difference in the VCDs stems from the difference in the computation of the cloudy AMF. The total AMF (BEHRAMFTrop or BEHRAMFTropVisOnly) is the cloud radiance fraction (f) weighted average of clear and cloudy AMFs:

	p_0	p'_0	w
A_{clr} (total and vis only)	Terrain pressure	Terrain pressure	Clear scattering wts.
A_{cld} (total)	Cloud pressure	Terrain pressure	Cloudy scattering wts.
A_{cld} (vis. only)	Cloud pressure	Cloud pressure	Cloudy scattering wts.

Table 2: Meaning of variables in Eq. (2) for the clear-sky AMF component, the cloudy AMF component for the total column, and the cloudy AMF component for the visible-only column.

$$A = (1 - f) \cdot A_{\mathrm{clr}} + f \cdot A_{\mathrm{cld}} \tag{1}$$

The component AMFs are each calculated as:

$$A_{\mathrm{clr\ or\ cld}} = \frac{\int_{p_0}^{p_{\mathrm{tp}}} w(p)g(p)\,dp}{\int_{p'_0}^{p_{\mathrm{tp}}} g(p)\,dp} \tag{2}$$

Eq. (2) is used for the calculation of all the AMF components (clear and cloudy). The variables p_0, p'_0, and w_i change depending on whether the clear or cloudy AMF is being calculated, and whether the cloudy AMF should retrieve the total or visible only column (see Table 2). $g(p)$ is always the WRF-Chem *a priori* NO$_2$ profile.

From this, we can see how the AMF adds back in the ghost column to the total column. For the cloudy AMF, the numerator of Eq. (2) always calculates the modeled above-cloud slant column actually observed by OMI. In the visible-only product, the denominator is the model vertical column above cloud, thus the definition of AMF = SCD/VCD uses the visible column for both the SCD and VCD. In the total column product, the VCD is the total modeled column, so the visible SCD is scaled by how much NO$_2$ the model expects there to be below cloud.

4.4   Using scattering weights/averaging kernels

4.4.1   Variable layout

These are for advanced users who might wish to either use their own *a priori* NO$_2$ profile or to compare modeled NO$_2$ VCDs correctly with BEHR VCDs. Note that the averaging kernels for each pixel are simply the scattering weights divided by the BEHR AMF; they are provided separately simply for user convenience.

For new users wishing to begin using these variables, several resources will be helpful. Palmer et al. (2001) contains the original formulation of the relationship between scattering weights, *a priori* NO$_2$ profiles, and air mass factors and should definitely be studied. The first several chapters of *The Remote Sensing of Tropospheric Composition from Space* assembled by Burrows, Platt, and Borrell (Burrows et al., 2011) also describes the relationship of scattering weights, averaging kernels, and air mass factors.

What follows will be a description of why the scattering weights are presented how they are. BEHRScatteringWeights, BEHRAvgKernels, BEHRPressureLevels, and BEHRNO2apriori

are 3D variables; the first dimension is the vertical dimension, the second and third correspond to the dimensions of the normal 2D variables. To put this another way, if X were the array of values for these variables, then $X(:, 1, 1)$ would be the vector of values for the $(1, 1)$ pixel, $X(:, 1, 2)$ the vector for the $(1, 2)$ pixel and so on.

The BEHRPressureLevels variable gives the vertical coordinates for the other three. 28 of the pressure levels will be the same for every pixel; the remaining two will correspond to the terrain and cloud pressure. Should the terrain or cloud pressure match one of the standard 28 pressures, then the vector of values for this pixel will still be 30 elements long, but will end with fill values which should be removed.

4.4.2 Calculation of scattering weights

The scattering weights presented (w_{tot}) are calculated as the weighted average of a vector of clear and cloudy scattering weights obtained by interpolating the NASA OMNO2 TOMRAD look up table to the appropriate values of SZA, VZA, RAA (relative azimuth angle), albedo, and surface pressure. Mathematically:

$$w_{i,\text{tot}} = f_r w_{i,\text{cld}} + (1 - f_r) w_{i,\text{clr}} \tag{3}$$

$$w_{i,\text{clr}} = \begin{cases} 0 & \text{if } p_i > p_{\text{terr}} \\ w_{i,\text{clr}} & \text{otherwise} \end{cases} \tag{4}$$

$$w_{i,\text{cld}} = \begin{cases} 0 & \text{if } p_i > p_{\text{cld}} \\ w_{i,\text{cld}} & \text{otherwise} \end{cases} \tag{5}$$

where f_r is the radiance cloud fraction, $w_{i,\text{clr}}$ is the ith element in the clear sky scattering weight vector, $w_{i,\text{cld}}$ is the ith element in the cloudy scattering weight vector, p_i is the pressure level for the ith element in the vectors, p_{terr} is the terrain pressure for the pixel, and p_{cld} is the cloud top pressure for the pixel.

Therefore the final vector of scattering weights is the cloud radiance fraction-weighted average of the clear and cloudy scattering weight vectors after setting the clear sky vectors to 0 below the ground, and the cloudy vectors to 0 below the cloud top. This is an approximation of how the NASA scattering weights are reported. These scattering weights can be combined with an *a priori* NO$_2$ profile to get the total AMF using Eq. (2) above.

Note that, because the clear and cloudy scattering weights are combined by Eq. (3), using them in Eq. (2) will directly produce the final, total column AMF (rather than separate clear and cloudy AMFs that must be combined). You must use the terrain pressure for p_0'. Because of this, there is currently no way to produce visible-only columns using your own *a priori* profiles.

Further note that Eq. (2) integrates $g(p)$ as the in mixing ratio over pressure, see Appendix B of Ziemka et al. (2001) for information on how mixing ratio integrated over pressure should be done to be equivalent to number density integrated over altitude.

4.4.3 Justification for including terrain and cloud pressure weights

Although we publish the scattering weights in this form, we calculate our AMF as:

$$A = f_r A_{\text{cld}} + (1 - f_r) A_{\text{clr}} \tag{6}$$

where

$$A_{\mathrm{cld}} = \int_{p_{\mathrm{cld}}}^{p_{\mathrm{trop}}} g(p)w_{\mathrm{cld}}(p)\,dp \tag{7}$$

$$A_{\mathrm{clr}} = \int_{p_{\mathrm{terr}}}^{p_{\mathrm{trop}}} g(p)w_{\mathrm{clr}}(p)\,dp \tag{8}$$

Mathematically, this should be the same as:

$$A_{\mathrm{tot}} = \int_{p_{\mathrm{terr}}}^{p_{\mathrm{trop}}} g(p)w_{\mathrm{tot}}(p)\,dp \tag{9}$$

however computationally it was not. This was because in Eqns. 8-9 we interpolate $w(p)$ to the terrain or cloud top pressure *before* setting $w(p)$ to zero below those pressures. This is done because the interpolation should not modify the scattering weights—surface pressure is already accounted for in the lookup table—but rather is only meant to find the scattering weight at our lower integration limit.

However, if the same approach is carried out with w_{tot} in Eqn. 10, we are not interpolating between the same values, because now we *have* set $w_{\mathrm{clr}}(p)$ and $w_{\mathrm{cld}}(p)$ to 0 below terrain and cloud top, respectively. To avoid this problem, $w_{\mathrm{clr}}(p)$ and $w_{\mathrm{cld}}(p)$ are each interpolated to both terrain and cloud top height before being used in Eqns. 3–5. Using data from 1 Aug 2013, with the same NO_2 profile, the difference between the AMFs calculated using Eqns. 7 and Eqn. 10 decrease from a median of 12.9% without pre-interpolating to 0.299% with pre-interpolating.

References

Burrows, J., Platt, U., and Borrell, P.: The Remote Sensing of Tropospheric Composition from Space, Springer, 2011.

Palmer, P., Jacob, D., Chance, K., Martin, R., Spurr, R., Kurosu, T., Bey, I., Yantosca, R., Fiore, A., and Li, Q.: Air mass factor formulation for spectroscopic measurments from satellites: Application to formaldehyde retrievals from the Global Ozone Monitoring Experiment, J. Geophys. Res., 106, 14 539–14 550, 2001.

Russell, A., Perring, A., Valin, L., Bucsela, E., Browne, E., Min, K., Wooldridge, P., and Cohen, R.: A high spatial resolution retrieval of NO_2 column densities from OMI: method and evalutation, Atmos. Chem. Phys., 11, 8543–8554, 2011.

Ziemka, J., Chandra, S., and Bhartia, P.: "Cloud slicing": A new technique to derive upper tropospheric ozone from satellite measurements, J. Geophys. Res. Atmos., 106, 9853–9867, 2001.