Tropospheric Emissions: Monitoring of Pollution (TEMPO) – Advanced Baseline Imager (ABI) Hybrid Aerosol Detection Product (ADP) Users' Guide

Version 1, January 2025

Table of Contents

List of Figures

List of Tables

1. Purpose of this Guide

This Tropospheric Emissions: Monitoring of Pollution (TEMPO) – Advanced Baseline Imager (ABI) Hybrid Aerosol Detection Product (ADP) User's Guide is intended for users of the ADP Level 2 (L2) files generated from the synergy between the UV-visible spectrometer onboard TEMPO and ABI onboard the GOES-R satellites. It provides a general introduction to the TEMPO and ABI sensors, the Hybrid Aerosol Detection Product (including smoke/dust detection and UV Absorbing Aerosol Index), and the format and contents of the TEMPO-ABI Hybrid ADP L2 data files. This guide serves as an introduction to the more technical TEMPO-ABI Hybrid ADP Algorithm Theoretical Basis Document (ATBD), which as of early January 2025 is not yet finalized.

2. Points of Contact

For questions or comments regarding this document, please contact Shobha Kondragunta, Pubu Ciren, or Amy Huff.

3. Document Definitions

The aerosol detection product (ADP), also called the smoke/dust mask, is a qualitative indicator of the presence of smoke and dust aerosols in the atmosphere. Table 1 lists additional acronyms and abbreviations used in this document.

Table 1. List of acronyms and abbreviations used in this document.

Acronym/Abbreviation	Definition
AAI	Absorbing Aerosol Index
ABI	Advanced Baseline Imager
ADP	Aerosol Detection Product
AERONET	AErosol RObotic NETwork
ATBD	Algorithm Theoretical Basis Document
ECM	Enterprise Cloud Mask
EPS	Enterprise Processing System
FOR	Field of Regard
GOES	Geostationary Operational Environmental Satellites
IS-40e	Intelsat-40e
JPSS	Joint Polar Satellite Series
L2	Level 2
NUC	None/Unknown/Clear Sky
POCD	Probability of Correct Detection
RGB	True Color Imagery (Red-Green-Blue)
TOA	Top of the Atmosphere
TEMPO	Tropospheric Emissions: Monitoring of Pollution
UV	Ultraviolet
Vis	Visible

4. TEMPO and ABI Overview

TEMPO is the first space-based instrument to monitor air pollutants on an hourly basis across North America during the daytime. It is an ultraviolet (UV, 290-490 nm)–visible (Vis, 540-740 nm) spectrometer. TEMPO flies onboard Intelsat-40e (IS-40e), a commercial telecommunications satellite centered at 91.0° W longitude.

ABI is the primary instrument on the GOES-R Series satellites for imaging Earth's weather, oceans and environment. It is an imaging radiometer with 16 spectral bands: 2 visible bands, 4 near-infrared bands, and 10 infrared bands. The current operational GOES-R satellites are GOES-16 (GOES-East), centered at 75.2° W longitude, and GOES-18 (GOES-West), centered at 137.0° W longitude.

The spatial resolution of TEMPO is 2.0 km x 4.75 km at the center of its Field of Regard (FOR): 33.7° N latitude and 91.0° W longitude. The FOR extends from Mexico City and the Yucatan Peninsula to the Canadian oil sands in the north-south direction, and from the Atlantic Ocean to the Pacific Ocean in the east-west direction. TEMPO scans from east to west, covering its FOR in about 1 hour, with shorter scan times in the morning and evening (~40 minutes) and during special high-time-resolution operations. Data collected during each scan are distributed as granules to make individual file sizes manageable. See Section 6 for more information on the TEMPO-ABI Hybrid ADP L2 data files.

5. TEMPO-ABI Hybrid ADP Algorithm Overview

The TEMPO-ABI Hybrid ADP algorithm uses spectral and spatial threshold tests to identify pixels with smoke or dust aerosols. The Hybrid algorithm takes advantage of the synergy between TEMPO and ABI observations, using TEMPO's measurements in the UV spectrum for higher sensitivity to absorbing aerosols (Including smoke and dust) and ABI's measurements in the shortwave IR spectrum to separate smoke and dust. The algorithm treats detection differently over water and over land. Details of the Hybrid ADP algorithm are described in Appendix 2.

ADP is not retrieved at night or over regions covered by clouds, snow, or ice. ADP is retrieved over sun glint water regions, but dust ADP over sun glint should not be used due to the high number of false alarms.

The Hybrid ADP algorithm can run through three paths, depending on the spectral range of the input data: (1) IR-visible based detection, (2) UV/deep-blue based detection, and (3) combined IR-visible based and UV/deep-blue based detection. As shown in Table 2, TEMPO makes observations in the UV and deep-blue regions of the spectrum required for the UV/deep-blue path, and ABI makes observations in the visible and IR parts of the spectrum required for the IR-visible path. As a result, TEMPO-ABI Hybrid ADP is generated via option 3, using the combined IR-visible and UV/deep-blue algorithm paths. Note that ABI observations are co-registered to the TEMPO grid using a weighted average of all ABI pixels falling into a TEMPO pixel.

In addition, the Hybrid ADP algorithm inputs the upstream (external to the algorithm) Enterprise Cloud Mask (ECM) and the NOAA Global Multisensor Automated Snow/Ice Cover map to screen pixels that are contaminated by clouds and snow/ice, respectively. Details about the inputs to the TEMPO-ABI Hybrid ADP algorithm will be given in the ATBD.

The UV/deep-blue algorithm path computes the UV Absorbing Aerosol Index (AAI) and the Scaled Absorbing Aerosol Index (SAAI); SAAI conveys the relative intensity (thickness) and presence of smoke and dust aerosols. In contrast, the IR-visible path indicates only the presence (yes/no) of smoke and dust aerosols. Details about working with the TEMPO-ABI Hybrid ADP data files, including the SAAI and UV AAI variables, are given in Section 7.

The accuracy requirements of TEMPO-ABI Hybrid ADP retrievals are set at 80% probability of correct detection (POCD) for dust over water and land, 80% POCD for smoke over land, and 70% POCD for smoke over water. Verification of TEMPO-ABI Hybrid ADP POCD requirements is computed based on comparisons with AERONET Ångström-based smoke/dust classifications, as will be described in the ATBD. The Hybrid ADP algorithm performs most accurately for thick smoke and dust plumes over dark surfaces. Smoke detection over semi-arid and arid regions is less accurate due to the lower contrast with the relatively bright surface.

Table 2. TEMPO and ABI observations used as inputs to the Hybrid ADP algorithm.

Sensor	Sensor Band	Nominal Central Wavelength (µm)	Use in Hybrid ADP Algorithm				
			Aerosol Type		Algorithm Path		
			Smoke	Dust	UV	Deep-Blue	IR-Visible
TEMPO	UV detector	0.354	X	X	X		
	UV detector	0.384	X	X	X		
	UV detector	0.412	X	X		X	
	UV detector	0.445	X	X		X	
	UV detector	0.488	X	X	X	X	X
	Visible detector	0.555	X		X	X	
	Visible detector	0.64	X	X	X	X	X
ABI	3	0.865	X	X	X	X	X
	4	1.378		X			X
	5	1.61	X	X			X
	6	2.25	X		X	X	X
	7	3.9	X	X			X
	13	10.35		X			X
	14	11.2	X	X			X
	15	12.3		X			X

6. TEMPO-ABI Hybrid ADP Level 2 Data Files

Figure 1 breaks down the file naming convention for TEMPO-ABI Hybrid ADP L2 data files, which are distributed as granules in netCDF4 (.nc) format. Files are organized in terms of the TEMPO scan number (e.g., S004) and granule number (e.g., G07) for each day. Each TEMPO scan covers a time period of ~60 minutes. Individual granule files contain data for the full FOR in the North-South direction, but only a portion of the East-West direction corresponding to a short time range (~6.7 minutes in nominal operations).

TEMPO-ABI Hybrid ADP data variables have 2.0 km x 4.75 km spatial resolution at the center of TEMPO's FOR and are contained in ~123 x 2048 arrays, with dimensions corresponding to TEMPO's East-West scanning mirror steps and North-South tracks. Due to the relatively short granule length and the data resolution, users should expect a total of approximately ~150 granule files each day (daytime coverage).

The granule number (e.g., G01, G02) does not always correspond to the same geographic location for each scan, but rather to the order in which the granule was collected for the scan. This is because granules will be missing from scans made early and late in the calendar day, corresponding to locations where the sun is below the horizon (before sunrise or after sunset).

Note that the files use UTC time. Some TEMPO scans during daylight hours over North America occur during or after UTC midnight, so files from late in the calendar day will be labeled with the date of the next day.

The "product version" in the filename (e.g., "V03" in Figure 1) refers to the version of the entire TEMPO product system, which is tied to the version of TEMPO L1B (radiances) data. It does not refer to the version of the Hybrid ADP data or the Hybrid ADP algorithm; this information is given in global file metadata as the "algorithm_version" attribute.

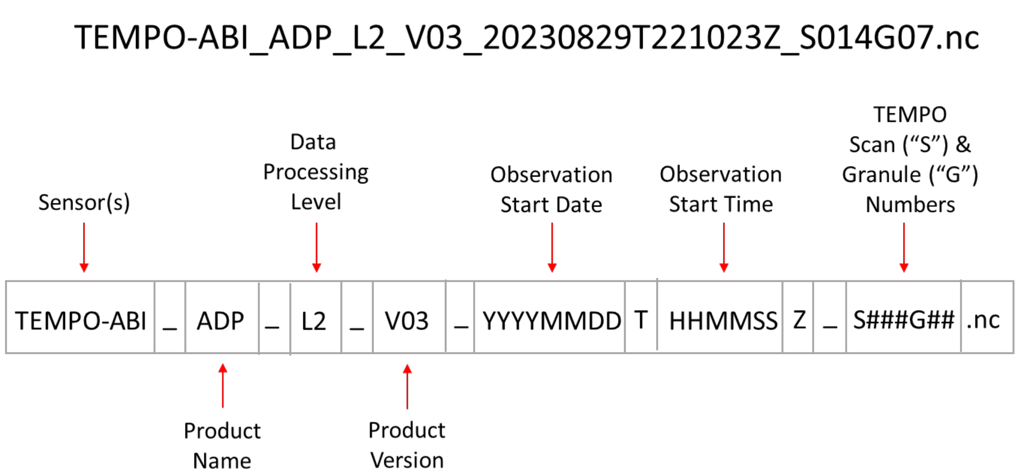

Figure 1. Example of a TEMPO-ABI Hybrid ADP Level 2 filename and breakdown of the naming convention.

7. Working with TEMPO-ABI Hybrid ADP Level 2 Data Files

Hybrid ADP L2 files contain many variables, listed in Table 3. The variables of interest to most users include (group name listed first):

- geolocation/latitude
- geolocation/longitude
- product/smoke
- product/dust
- product/saai
- product/uv_aai
- product/deepblue_aai
- quality_diagnostic_flags/qc_flag
- quality_diagnostic_flags/pqi2
- quality_diagnostic_flags/pqi4

Since the Hybrid ADP algorithm allows for two detection paths, users can work with the ADP data in two primary ways: (1) identify the presence only (yes/no) of dust and smoke aerosols or (2) identify the presence and relative intensity of dust and smoke aerosols. Figures 2 and 3 show examples of displaying ADP data for these two options, and examples of IDL and Python code for processing the Hybrid ADP data files are given in Appendix 1.

For option 1 (presence only of smoke/dust), use the following variables (**bolded**) in the Hybrid ADP data file (group name listed first):

- **geolocation/latitude**
- **geolocation/longitude**
- For smoke aerosols:
  - **product/smoke** (see Table 3)
    - Select pixels = 1 for smoke present
- For dust aerosols:
  - **product/dust** (see Table 3)
    - Select pixels = 1 for dust present
  - **quality_diagnostic_flags/pqi2**, bit 1 (see Table 6)
    - Select pixels = 0 for outside of sun glint region

For option 2 (presence and relative thickness of smoke/dust), use the following variables (bolded) in the Hybrid ADP data file (group name listed first):

- **geolocation/latitude**
- **geolocation/longitude**
- For smoke aerosols:
  - **product/smoke** (see Table 3)
    - Select pixels = 1 for smoke present
  - **quality_diagnositc_flags/pqi4**, bits 4-5 (see Table 8)
    - Select pixels = 0 for deep-blue algorithm path **and** = 48 for both algorithm paths

    - **product/saai**
        - Display SAAI in the range of [0,2] for smoke (thin to thick intensity)
- For dust aerosols:
    - **product/dust** (see Table 3)
        - Select pixels = 1 for dust present
    - **quality_diagnostic_flags/pqi2**, bit 1 (see Table 6)
        - Select pixels = 0 for outside of sun glint region
    - **quality_diagnostic_flags/pqi4**, bits 6-7 (see Table 8)
        - Select pixels = 0 for deep-blue algorithm path **and** = 192 for both algorithm paths
    - **product/saai**
        - Display SAAI in the range of [0,5] for dust (thin to thick intensity)

The Hybrid ADP data files also include a confidence flag, **quality_diagnostic_flags/qc_flag**, which allows users to select high, medium, or low-quality pixels. For ***qualitative*** applications, all qualities (high + medium + low) may be used; this is the default setting, so **quality_diagnostic_flags/qc_flag**, does not need to be set explicitly for qualitative use. For ***quantitative*** applications, the Top 2 qualities (high + medium) are recommended, as follows:

- For smoke pixels:
    - **quality_diagnostic_flags/qc_flag**, bits 2-3 (see Table 4)
        - Select pixels = 0 for high quality **and** = 4 for medium quality
- For dust pixels:
    - **quality_diagnostic_flags/qc_flag**, bits 4-5 (see Table 4)
        - Select pixels = 0 for high quality **and** = 16 for medium quality

8. Known Issues to Date

The TEMPO-ABI Aerosol team has identified the following ADP data quality problems that the users should be aware of:

- False smoke detections and extremely high AAI may occur over bright surfaces at certain times of the day.
- Sulfur dioxide (SO_2) plumes from volcanic eruptions may be identified as smoke.
- Thin dust plumes over vegetated surfaces may be mis-classified as smoke.

August 2023 is the first month with operational TEMPO L1b data, so as a result, many individual granules and full scans are missing, sometimes for entire days. These missing upstream L1b data cause corresponding gaps in ADP L2 granules and scans.

TEMPO L1b data will very likely be reprocessed by the TEMPO science team in the future to rectify calibration issues. At that time, ADP L2 data will be reprocessed accordingly.

9. Data Access

Currently, one month of TEMPO-ABI Hybrid ADP data files for August 2023 are available from NOAA.

The primary source for the TEMPO-ABI Hybrid ADP data files will be NASA Earthdata, although as of January 2025, the TEMPO aerosol products have not yet been added to Earthdata. We plan to provide 17 months of TEMPO aerosol products data (August 2023 to December 2024) through ASDC for user access. Users can register for a free account (needed to download data files) at Earthdata login. Once the TEMPO aerosol files have been uploaded to Earthdata, use the collection short name "TEMPO-ABI_ADP_L2" to search for the Hybrid ADP data files.

Table 3. Output variables from the TEMPO-ABI Hybrid ADP algorithm. Units of "1" indicate a unitless quantity.

Group	Data Variable	Type	Description	Units	Range
geolocation	longitude	Float	Pixel center longitude	° East	-180, 180
	latitude	Float	Pixel center latitude	° North	-90, 90
product	smoke	Byte	Smoke Flag: 1 = yes, 0 = no	1	0, 1
	dust	Byte	Dust Flag: 1 = yes, 0 = no	1	0, 1
	cloud	Byte	Cloud Flag: 1 = yes, 0 = no	1	0, 1
	nuc	Byte	None, Unknown, Clear Sky Flag: 1 = yes, 0 = no	1	0, 1
	snowice	Byte	Snow/Ice Flag: 1 = yes, 0 = no	1	0, 1
	saai	Float	Scaled Absorbing Aerosol Index	1	0, 30
	dsdi	Float	Dust Smoke Discrimination Index	1	-50, 50
	deepblue_aai	Float	Deep-blue Absorbing Aerosol Index	1	-30, 30
	uv_aai	Float	UV Absorbing Aerosol Index	1	-50, 50
quality_diagonostic_flags	qc_flag	Byte	Quality Flag for Smoke, Dust, and NUC (see Table 4)	1	-128, 127
	pqi1	Byte	Product Quality Information (see Table 5)	1	-128, 127
	pqi2	Byte	Product Quality Information (see Table 6)	1	-128, 127
	pqi3	Byte	Product Quality Information (see Table 7)	1	-128, 127
	pqi4	Byte	Product Quality Information (see Table 8)	1	-128, 127
	std_dev_410nm	Float	standard deviation of the reflectance at 410 nm in a 3x3 box	1	0, 10
	std_dev_865nm	Float	standard deviation of the reflectance at 865 nm in a 3x3 box	1	0, 10
	std_dev_2210nm	Float	standard deviation of the reflectance at 2210 nm in a 3x3 box	1	0, 10

Table 4. Definitions of bit-wise quality flags for the **quality_diagnostic_flags/qc_flag** variable.

Bit*	Quality Flag Name	Meaning (2-bits)			
		01	10	00	11
0-1	----	Low quality	Medium quality	High quality	Bad/ missing
2-3	QC_SMOKE_CONFIDENCE				
4-5	QC_DUST_CONFIDENCE				
6-7	QC_NUC_CONFIDENCE				

*Start from the least significant bit

Table 5. Definitions of bit-wise diagnostic flags for the **quality_diagnostic_flags/pqi1** variable.

Bit*	Diagnostic Flag Name	Meaning			
		1-bit	0 (default)	1	-----
		2-bits	00 (default)	01	11
0	QC_INPUT_LON	1-bit	Valid longitude	Invalid longitude (longitude > 180 or < -180)	-----
1	QC_INPUT_LAT	1-bit	Valid latitude	Invalid latitude (latitude > 90 or < -90)	-----
2-3	QC_INPUT_SOLZEN	2-bits	Valid solar zenith angle (SZA) (0 ≤ SZA ≤ 60)	Invalid SZA (SZA > 90 or < 0)	90 ≥ SZA > 60
4-5	QC_INPUT_SATZEN	2-bits	Valid local zenith angle (VZA) (0 ≤ VZA ≤ 60)	Invalid VZA (VZA > 90 or < 0)	90 ≥ VZA > 60
6-7	QC_INPUT_SNOW/ICE_SOURCE	2-bits	TEMPO/ABI snow/ice mask	IMS snow/ice mask	Internal snow/ice mask

*Start from the least significant bit

Table 6. Definitions of bit-wise diagnostic flags for the **quality_diagnostic_flags/pqi2** variable.

Bit*	Diagnostic Flag Name	Meaning (1-bit)	
		0 (default)	1
0	QC_INPUT_SUNGLINT_SOURCE	TEMPO/ABI sun glint mask (from Cloud Mask product)	Internal sun glint mask
1	QC_INPUT_SUNGLINT	Outside of sun glint	Within sun glint
2	QC_INPUT_LAND/WATER	Water	Land
3	QC_INPUT_DAY/NIGHT	Day	Night
4	QC_WATER_SMOKE_INPUT	Valid TEMPO/ABI inputs	Invalid TEMPO/ABI inputs
5	QC_WATER_SMOKE_CLOUD	Cloud-free	Obscured by clouds
6	QC_WATER_SMOKE_SNOW/ICE	Snow/ice free	With snow/ice
7	QC_WATER_SMOKE_TYPE (only for IR/Visible algorithm path)	Thin Smoke	Thick Smoke

*Start from the least significant bit

Table 7. Definitions of bit-wise diagnostic flags for the **quality_diagnostic_flags/pqi3** variable.

Bit*	Diagnostic Flag Name	Meaning (1-bit)	
		0 (default)	1
0	QC_WATER_DUST_INPUT	Valid TEMPO/ABI inputs	Invalid TEMPO/ABI inputs
1	QC_WATER_DUST_CLOUD	Cloud-free	Obscured by clouds
2	QC_WATER_DUST_SNOW/ICE	Snow/ice free	With snow/ice
3	QC_WATER_DUST_TYPE (only for IR/Visible algorithm path)	Thin dust	Thick dust
4	QC_LAND_SMOKE_INPUT	Invalid TEMPO/ABI inputs	Valid TEMPO/ABI inputs
5	QC_LAND_SMOKE_CLOUD	Cloud-free	Obscured by clouds
6	QC_LAND_SMOKE_SNOW/ICE	Snow/ice free	With snow/ice
7	QC_LAND_SMOKE_TYPE (only for IR/Visible algorithm path)	Fire	Thick smoke

*Start from the least significant bit

Table 8. Definitions of bit-wise diagnostic flags for the **quality_diagnostic_flags/pqi4** variable.

<table>
<tr><th rowspan="3">Bit*</th><th rowspan="3">Diagnostic Flag Name</th><th colspan="5">Meaning</th></tr>
<tr><th>1-bit</th><th>0 (default)</th><th>1</th><th>-----</th><th>-----</th></tr>
<tr><th>2-bits</th><th>00 (default)</th><th>10</th><th>01</th><th>11</th></tr>
<tr><td>0</td><td>QC_LAND_DUST_INPUT</td><td>1-bit</td><td>Valid TEMPO/ABI inputs</td><td>Invalid TEMPO/ABI inputs</td><td>-----</td><td>-----</td></tr>
<tr><td>1</td><td>QC_LAND_DUST_CLOUD</td><td>1-bit</td><td>Cloud-free</td><td>Obscured by clouds</td><td>-----</td><td>-----</td></tr>
<tr><td>2</td><td>QC_LAND_DUST_SNOW/ICE</td><td>1-bit</td><td>Snow/ice free</td><td>With snow/ice</td><td>-----</td><td>-----</td></tr>
<tr><td>3</td><td>QC_LAND_DUST_TYPE
(only for IR/Visible algorithm path)</td><td>1-bit</td><td>Thin dust</td><td>Thick dust</td><td>-----</td><td>-----</td></tr>
<tr><td>4-5</td><td>Smoke_Detection_Algorithm_Path</td><td>2-bits</td><td rowspan="2">UV/Deep-blue</td><td rowspan="2">Missing</td><td rowspan="2">IR-Visible</td><td rowspan="2">Both</td></tr>
<tr><td>6-7</td><td>Dust_Detection_Algorithm_Path</td><td>2-bits</td></tr>
</table>

*Start from the least significant bit

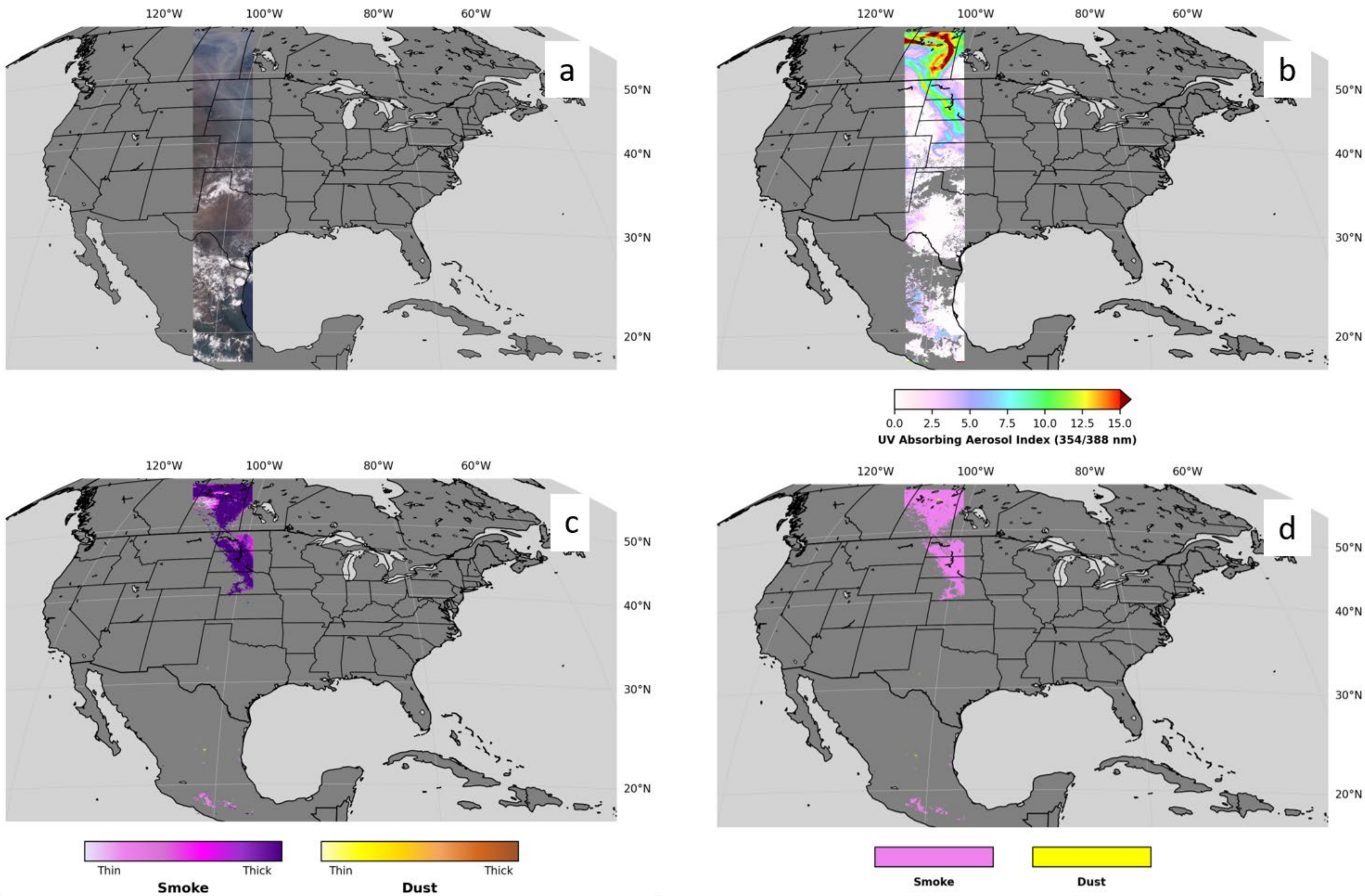

Figure 2. TEMPO-ABI Hybrid ADP for Granule 07 of Scan 014 (22:10 UTC) on August 29, 2023: (a) true color image showing wildfire smoke, (b) UV Absorbing Aerosol Index, (c) Scaled Absorbing Aerosol Index, and (d) smoke & dust detection.

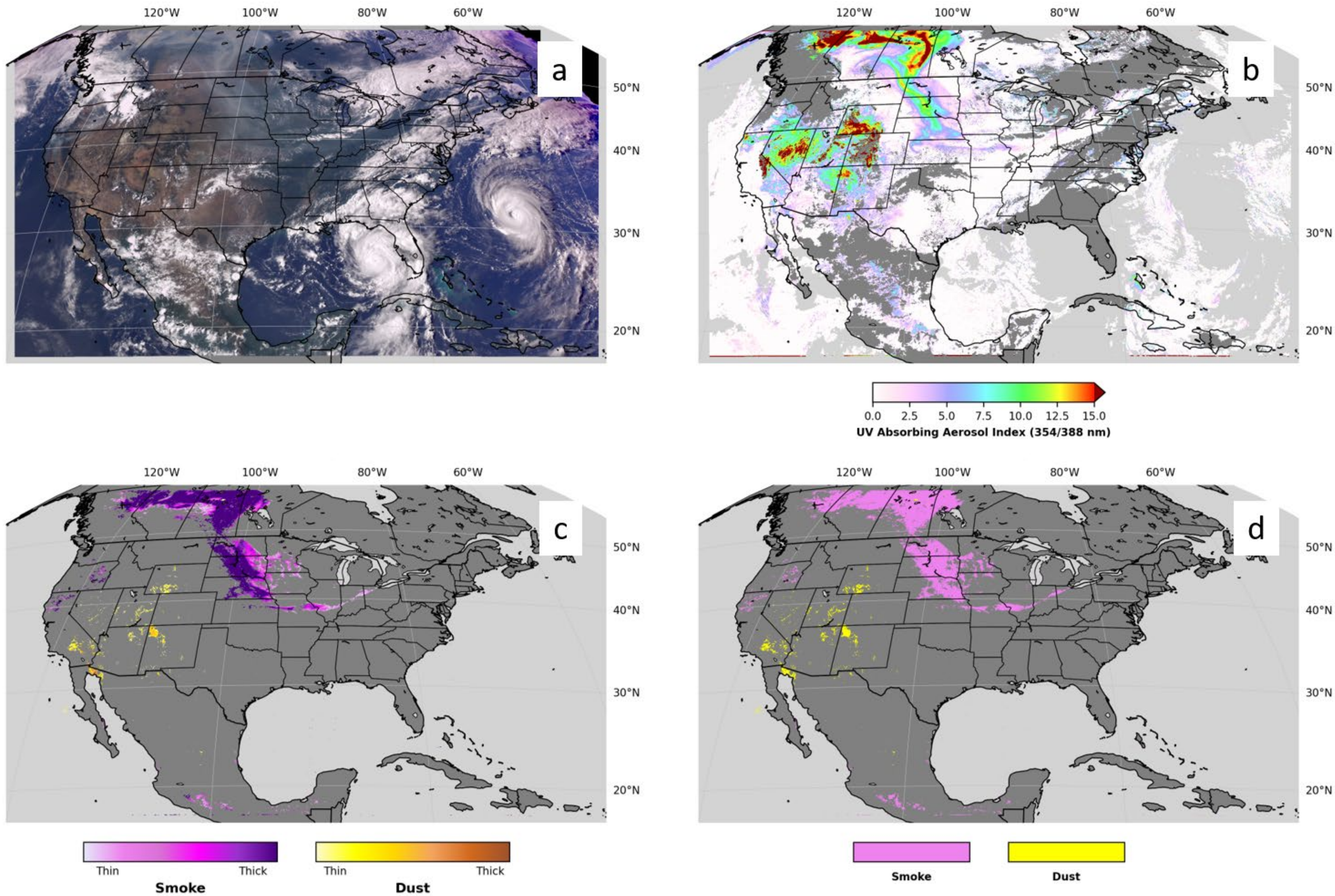

Figure 3. TEMPO-ABI Hybrid ADP for all Granules in Scan 014 (21:33-22:29 UTC) on August 29, 2023: (a) true color image showing wildfire smoke, (b) UV Absorbing Aerosol Index, (c) Scaled Absorbing Aerosol Index, and (d) smoke & dust detection.

Appendix 1. Helpful Tools for Working with TEMPO-ABI Hybrid ADP Files

A. NetCDF Data Model

For users unaccustomed to working with NetCDF4 formatted files, please visit the website https://docs.unidata.ucar.edu/netcdf-c/current/netcdf_data_model.html for information.

B. Panoply Data Viewer

The Panoply NetCDF, HDF and GRIB Data Viewer developed by NASS GISS is a convenient tool for visualization of the EPS ADP outputs. Please visit the website https://www.giss.nasa.gov/tools/panoply/ for more information about this software.

C. IDL Tools

IDL has a built-in library of commands for NetCDF4 files. Documentation can be found online at https://www.harrisgeospatial.com/docs/NCDF_Overview.html or using IDL Help.

Michael Galloy has written a particularly helpful IDL program to read HDF5 (also works for NetCDF4) arrays into IDL, available at http://docs.idldev.com/idllib/hdf5/mg_h5_getdata-code.html.

D. Example Code

a. Example of IDL Code for Processing TEMPO-ABI Hybrid ADP L2 Files

```
pro read_TEMPO-ABI_ADP
misval=-999.9
filename=' TEMPO-ABI_ADP_L2_V03_20230808T184123Z_S002G03.nc'
; get version no from the file
Version=strmid(filename,16,3)

;Read the data
lat =mg_h5_getdata(filename,'/geolocation/latitude')
lon = mg_h5_getdata(filename,'/geolocation/longitude')
    ;smoke/dust mask (1/0 - YES/NO)
    smoke=mg_h5_getdata(filename,'/product/smoke')
```

```
        dust=mg_h5_getdata(filename,'/product/dust')
        cld = mg_h5_getdata(filename,'/product/cloud')
        Nuc = mg_h5_getdata(filename,'/product/nuc')
snowice=mg_h5_getdata(filename,'/product/snowice')
         ; Scaled-Absorbing Aerosol Index
  AAI = mg_h5_getdata(filename,'/product/saai')
; Deep_blue (visible) Absorbing aerosol Index)
    Deepblue_AAI =
mg_h5_getdata(filename,'/product/deepblue_aai')
; UV Absorbing Aerosol Index
    UV_AAI = mg_h5_getdata(filename,'/product/uv_aai')


Qual_B1=mg_h5_getdata(filename,'/quality_diagnostic_flags/qc_Ffag')
Qual_B2=mg_h5_getdata(filename,'/quality_diagnostic_flags/pqi1')
Qual_B3=mg_h5_getdata(filename,'/quality_diagnostic_flags/pqi2')
Qual_B4=mg_h5_getdata(filename,'/quality_diagnostic_flags/pqi3')
Qual_B5=mg_h5_getdata(filename,'/quality_diagnostic_flags/pqi4')

     dimsize=Size(lat,/DIMENSION)
;    print, dimsize
     nlon=dimsize(0)
     nlat=dimsize(1)
;=============================================================================================
; Ancillary information: sun glint mask(sungln,  0: out of glint; 1: inside glint), land/water mask (0: water, 1: land)
;=============================================================================================
sugln=smoke
sugln(*,*)=0
lndwat=smoke
lndwat(*,*)=0
lndwat=smoke
lndwat(*,*)=0
```

```
; Byte3 bit 2 (sun glint)
tmp_mask=Qual_B3
idx = WHERE((tmp_mask AND 2) EQ 2, COMPLEMENT=cidx, nc)
IF nc GT 0 THEN sugln[idx] = 1

; Byte3 bit 3 (land/water)
tmp_mask=Qual_B3
idx = WHERE((tmp_mask AND 4) EQ 4, COMPLEMENT=cidx, nc)
IF nc GT 0 THEN lndwat[idx] = 1

;no glint on land
idx=where(lndwat eq 1 and sugln eq 1,nn)
if ( nn gt 0) then sugln(idx)=0

;==================================================================================
; ADP INFORMATION; Three different ways to extract ADP information
;===================================================================================
;1). smoke, dust type (yes/no flag: 1/0): Smoke_Type, Dust_Type
;===================================================================================
Smoke_Type=smoke
Dust_Type=dust

; note that dust over sun glint area has to be removed in order to reduce false detection, user has to mask the dust out with sun glint mask as following:
idx=where(Dust_Type eq 1 and sugln eq 1, nn)
if (nn gt 0) then begin
Dust_Type(idx)=0
endif

; User can also choose results from which the algorithm path to use here
;smoke/dust detection algorithm path (indicating the detected smoke/dust is from which algorithm path)

;Smoke_Type
;byte5, bits 4-5 (smoke detection algorithm source): 00: UV/deep-blue only; 01:ir_visible only; 11: both
```

```
tmp_mask=Qual_B5

;a. smoke detected by either deep-blue or IR_Visible Path (default)

;b. Choose smoke detected with deep-blue algorithm path
 idx=where(((ishft(tmp_mask,-4) and 3) eq 2),COMPLEMENT=cidx,n)
 if ( n gt 0) then smoke_Type(idx)=0

;c. Choose smoke detected with IR_Visible algorithm path
idx=where(((ishft(tmp_mask,-4) and 3) eq 0),COMPLEMENT=cidx,n)
 if ( n gt 0) then smoke_Type(idx)=0

;dust_type
```

;byte5, bits 6-7 (Dust detection algorithm source): 00: UV/deep-blue only; 01:ir_visible only; 11: both

```
tmp_mask=Qual_B5

;a. Dust detected by either deep-blue or IR_Visible Path (default)

;b. Choose dust detected with UV/deep-blue algorithm path

 idx=where(((ishft(tmp_mask,-6) and 3) eq 2), n)
if (n gt 0 ) then Dust_Type(idx)=0

;c. Choose Dust detected with IR_Visible algorithm path
idx=where(((ishft(tmp_mask,-6) and 3) eq 0),COMPLEMENT=cidx,n)
 if ( n gt 0 ) then Dust_Type(idx)=0

;==================================================================================================
```

;2). **smoke, dust type Quality (Confidence) level: Smoke_Qual and Dust_Qual**
; **1: Low; 2: Medium; and 3: High;**

```
;==================================================================================================
```

```
Smoke_Qual=smoke
Smoke_Qual(*,*)=0
 Dust_Qual=smoke
Dust_Qual(*,*)=0

; byte1 bits 3-4 (smoke quality: High)
tmp_mask=Qual_B1
idx = WHERE(((ishft(tmp_mask,-2) AND 3) EQ 3) and (Smoke_Type eq 1), COMPLEMENT=cidx, nc)
IF nc GT 0 THEN Smoke_Qual[idx] = 3

;byte1, bits 5-6 (dust quality: High)
tmp_mask=Qual_B1
idx = WHERE(((ishft(tmp_mask,-4) AND 3) EQ 3) and (Dust_Type eq 1), COMPLEMENT=cidx, nc)
IF nc GT 0 THEN Dust_Qual[idx] = 3


; byte1 bits 3-4 (smoke quality: Medium)
tmp_mask=Qual_B1
idx = WHERE(((ishft(tmp_mask,-2) AND 3) EQ 2) and (Smoke_Type eq 1), COMPLEMENT=cidx, nc)
IF nc GT 0 THEN Smoke_Qual[idx] = 2


;byte1, bits 5-6 (dust quality: Medium)
tmp_mask=Qual_B1
idx = WHERE(((ishft(tmp_mask,-4) AND 3) EQ 2) and (Dust_Type eq 1), COMPLEMENT=cidx, nc)
IF nc GT 0 THEN Dust_Qual[idx] = 2


; byte1 bits 3-4 (smoke quality: Low)
tmp_mask=Qual_B1
idx = WHERE(((ishft(tmp_mask,-2) AND 3) EQ 1) and (Smoke_type eq 1), COMPLEMENT=cidx, nc)
IF nc GT 0 THEN Smoke_Qual[idx] = 1
```

```
;byte1, bits 5-6 (dust quality)
tmp_mask=Qual_B1
idx = WHERE(((ishft(tmp_mask,-4) AND 3) EQ 1) and (Dust_type eq 1), COMPLEMENT=cidx, nc)
IF nc GT 0 THEN Dust_Qual[idx] = 1

;==========================================================================================
;3). smoke, dust type with AAI value in order to visualize smoke/dust intensity: Smoke_AAI and Dust_AAI
;==========================================================================================

;smoke/dust detection algorithm path (indicating the detected smoke/dust is from which algorithm path)
;AAI values is only available for pixels detected with UV/deep-blue algorithm path
;smoke
;byte5, bits 4-5 (smoke detection algorithm source): 00: UV/deep-blue 01:ir_visible 11: both

;a. Choose smoke detected with deep-blue algorithm path
tmp_mask=Qual_B5
idx=where(((ishft(tmp_mask,-4) and 3) eq 2),COMPLEMENT=cidx,n)
if ( n gt 0) then Smoke(idx)=0
;dust
;byte5, bits 6-7 (smoke detection algorithm source): 00: UV/deep-blue 01:ir_visible 11: both
;a. Choose dust detected with deep-blue algorithm path
 tmp_mask=Qual_B5
idx=where(((ishft(tmp_mask,-6) and 3) eq 2), n)
 if (n gt 0) then dust(idx)=0


;dust (remove dust detection in the sun glint region)
idx=where(dust eq 1 and sugln eq 1, nn)
if (nn gt 0) then begin dust(idx)=0
endif


; scaled Absorbing Aerosol Index for dust - show dust intensity
 Dust_AAI=AAI
```

```
Dust_AAI(*,*)=MISVAL idx=where(dust
eq 1,n)
if (n gt 0) then Dust_AAI(idx)=AAI(idx)
```

; scaled Absorbing Aerosol Index for smoke - show smoke intensity
; User can color-scaled this value from 0 to 2 for smoke and 0 to 5 for dust for best color-stretch.

```
Smoke_AAI=AAI

Smoke_AAI(*,*)=MISVAL

 idx=where(smoke eq 1,n)

if (n gt 0) then Smoke_AAI(idx)=AAI(idx)

end
```

b. Example of Python Code for Processing TEMPO-ABI Hybrid ADP L2 Files using Xarray & NumPy

```
Python configuration:
python=3.11
  - numpy=1.26.4
  - netcdf4=1.7.2
  - dask=2024.10.0
  - xarray=2024.10.0

# User: enter directory & file name of TEMPO data file
file_path = Path('D://Data/2023/20230829')  # Directory where .nc file is located
file_name = 'TEMPO-ABI_ADP_L2_V03_20230829T221023Z_S014G07.nc'

# User: enter type of ADP to process
product = 'saai'  # 'saai' (SAAI, relative thickness) or 'detection' (presence flag, yes/no)
```

```
# Module to set filesystem paths for user's operating system
from pathlib import Path

# Library to work with labeled multi-dimensional arrays
import xarray as xr

# Library to perform array operations
import numpy as np


# Process TEMPO ADP smoke/dust detection
def process_tempo_adp_detection(dt):

    # Convert xarray Data Arrays to NumPy masked arrays w/correct dtype
    # Select "smoke present" (smoke = 1) and "dust present" (dust = 1) pixels
    # Casting xarray float32 to native int8 gives error for fill value
    # Silence "invalid value encountered in cast" warning with np.errstate
    with np.errstate(invalid="ignore"):
        smoke_detection = dt['/product/smoke'].where(dt['/product/smoke'] == 1).to_masked_array().astype('int8')
        dust_detection = dt['/product/dust'].where(dt['/product/dust'] == 1).to_masked_array().astype('int8')
        pqi2 = dt['/quality_diagnostic_flags/pqi2'].to_masked_array().astype('int8')
        pqi4 = dt['/quality_diagnostic_flags/pqi4'].to_masked_array().astype('int8')

    # Select deep-blue based algorithm smoke pixels using "pqi4" bits 4-5
    # Mask missing and IR-visible path
    # deep blue (00): 0 + 0 = 0, missing (10): 16 + 0 = 16, IR-visible (01): 0 + 32 = 32, both (11): 16 + 32 = 48
    smoke_algorithm_mask = ((pqi4 & 16 == 16) & (pqi4 & 32 != 32)) | ((pqi4 & 16 != 16) & (pqi4 & 32 == 32))
    smoke_detection = np.ma.masked_where(smoke_algorithm_mask, smoke_detection)

    # Select dust pixels outside of sun-glint areas using "pqi2" bit 1
    # outside sun-glint = 0, within sun-glint = 2
    dust_detection = np.ma.masked_where(pqi2 & 2 == 2, dust_detection)
```

```
    # Select deep-blue based algorithm dust pixels using "pqi4" bits 6-7
    # Mask missing and IR-visible path
    # deep blue (00): 0 + 0 = 0, missing (10): 64 + 0 = 64, IR-visible (01): 0 + 128 = 128, both (11): 64 + 128 = 192
    dust_algorithm_mask = ((pqi4 & 64 == 64) & (pqi4 & 128 != 128)) | ((pqi4 & 64 != 64) & (pqi4 & 128 == 128))
    dust_detection = np.ma.masked_where(dust_algorithm_mask, dust_detection)

    return smoke_detection, dust_detection

# Process TEMPO ADP SAAI
def process_tempo_adp_saai(dt):

    # Convert xarray Data Arrays to NumPy masked arrays w/correct dtype
    # Select "smoke present" (smoke = 1) and "dust present" (dust = 1) pixels
    # Casting xarray float32 to native int8 gives error for fill value
    # Silence "invalid value encountered in cast" warning with np.errstate
    with np.errstate(invalid="ignore"):
        saai_smoke = dt['/product/saai'].where(dt['/product/smoke'] == 1).to_masked_array().astype('float32')
        saai_dust = dt['/product/saai'].where(dt['/product/dust'] == 1).to_masked_array().astype('float32')
        pqi2 = dt['/quality_diagnostic_flags/pqi2'].to_masked_array().astype('int8')
        pqi4 = dt['/quality_diagnostic_flags/pqi4'].to_masked_array().astype('int8')

    # Select deep-blue based algorithm smoke pixels using "pqi4" bits 4-5
    # Mask missing and IR-visible path
    # deep blue (00): 0 + 0 = 0, missing (10): 16 + 0 = 16, IR-visible (01): 0 + 32 = 32, both (11): 16 + 32 = 48
    smoke_algorithm_mask = ((pqi4 & 16 == 16) & (pqi4 & 32 != 32)) | ((pqi4 & 16 != 16) & (pqi4 & 32 == 32))
    saai_smoke = np.ma.masked_where(smoke_algorithm_mask, saai_smoke)

    # Select dust pixels outside of sun-glint areas using "pqi2" bit 1
    # outside sun-glint = 0, within sun-glint = 2
    saai_dust = np.ma.masked_where(pqi2 & 2 == 2, saai_dust)
```

```
    # Select deep-blue based algorithm dust pixels using "pqi4" bits 6-7
    # Mask missing and IR-visible path
    # deep blue (00): 0 + 0 = 0, missing (10): 64 + 0 = 64, IR-visible (01): 0 + 128 = 128, both (11): 64 + 128 = 192
    dust_algorithm_mask = ((pqi4 & 64 == 64) & (pqi4 & 128 != 128)) | ((pqi4 & 64 != 64) & (pqi4 & 128 == 128))
    saai_dust = np.ma.masked_where(dust_algorithm_mask, saai_dust)

    return saai_smoke, saai_dust

# Main function
if __name__ == "__main__":

    # Set full path for ADP data file
    file_id = file_path / file_name

    # Open file using xarray (automatically closes file when done)
        with xr.open_datatree(file_id, engine='netcdf4') as dt:
            if product == ‘detection’:
                # Process TEMPO ADP smoke/dust detection
                smoke, dust = process_tempo_adp_detection(dt)
             elif product == ‘saai’:
                # Process TEMPO ADP SAAI
                smoke, dust = process_tempo_adp_saai(dt)

        # Read in latitude and longitude
        # Fill missing lat/lon pixels so mpl.pcolormesh plots ADP data
        if np.isnan(np.sum(dt['/geolocation/latitude'].values)):
            latitude = dt['/geolocation/latitude'].fillna(-999.99)
            longitude = dt['/geolocation/longitude'].fillna(-999.99)
        else:
            latitude = dt['/geolocation/latitude']
            longitude = dt['/geolocation/longitude']
```

Appendix 2. Details of the TEMPO-ABI Hybrid ADP Algorithm

The TEMPO-ABI Hybrid ADP algorithm combines observations from TEMPO with ABI to optimize the detection of smoke and dust. Since TEMPO provides observations in the UV spectrum, where absorption by both smoke and dust is strongest, the Joint Polar Satellite System Enterprise Processing System (JPSS EPS) Visible Infrared Imaging Radiometer Suite (VIIRS) ADP algorithm has been modified to include a UV component in addition to the existing deep-blue and IR-Visible components. As nearly all IR-Visible observations come from ABI, the TEMPO-ABI Hybrid ADP algorithm focuses on the other two components, which rely on three indices: the UV Absorbing Aerosol Index (UV_{AAI}), the Visible (deep-blue) Absorbing Aerosol Index (Vis_{AAI}) and the Dust Smoke Discrimination Index (DSDI). They are defined as:

$$Vis_{AAI} = -100\left[\log_{10}\left(\frac{R_{412nm}}{R_{445nm}}\right) - \log_{10}\left(\frac{R'_{412nm}}{R'_{445nm}}\right)\right] \quad (1)$$

$$UV_{AAI} = -100\left[\log_{10}\left(\frac{R_{354nm}}{R_{388nm}}\right) - \log_{10}\left(\frac{R'_{354nm}}{R'_{388nm}}\right)\right] \quad (2)$$

$$DSDI = -10\left[\log_{10}\left(\frac{R''_{412nm}}{R''_{2250nm}}\right)\right] \quad (3)$$

where R is the TOA reflectance, R' is the reflectance from Rayleigh scattering, and R'' is the TOA reflectance corrected for Rayleigh scattering.

The JPSS EPS ADP algorithm is enhanced by the inclusion of a UV component. As indicated by the spectral variation of the imaginary part of the refractive index of atmospheric aerosols, absorption by smoke, dust, and volcanic ash increases as the wavelength decreases in the UV spectrum. This behavior creates a wavelength-dependent absorption feature that aids in the detection of smoke and dust in the atmosphere. In addition, UV AAI is less sensitive to surface type (surface reflectance is low at UV wavelengths compared to visible wavelengths), but it is enhanced when absorbing aerosol is present in clear sky conditions, above clouds, or above a snow-ice surface, as a result of the multiple scattering between clouds/snow-ice and the aerosol layer. This feature allows us to detect smoke/dust above clouds, which is not possible for the previous JPSS EPS ADP algorithm that relies just on the deep-blue AAI alone.

The first step in applying the TEMPO-ABI Hybrid ADP algorithm is generating all required inputs for each TEMPO pixel. Depending on the surface type, either land or water, the algorithm follows different paths due to varying surface contrasts at UVBand1, UVBand2, M1, M2, and M11, which necessitate different threshold values for detecting smoke, dust, and residual clouds. If a pixel is determined to have snow/ice present by the snow/ice mask, the smoke/dust detection process is bypassed. The presence of cloud in each TEMPO pixel is determined by the percentage of confident cloudy ABI pixels from the ABI L2 cloud mask remapped to the TEMPO grid. If this percentage exceeds 50%, the pixel is classified as cloudy, halting smoke/dust detection. Additionally, Rayleigh-corrected reflectance is used to screen cloudy pixels; if the Rayleigh-corrected reflectance at 412 nm exceeds 0.4 over land or 0.32 over water, the pixel is considered cloudy. Due to the frequent misclassification of dust plumes as clouds by the ABI L2 cloud mask over land, the remapped cloud mask is not applied to dust detection over land. In addition, for smoke detection over land, a smoke call-back procedure is applied for pixels identified as cloudy by the remapped ABI cloud mask, but with an extremely high UV AAI (>14), owing to the advantage of the UV spectrum, i.e., the capability to detect absorbing aerosols above clouds.

For cloud-clear and snow/ice-free pixels, smoke detection starts with a UV AAI threshold, followed by smoke and dust discrimination using the DSDI. However, for cloudy pixels, smoke is called back if UV AAI >14. The same procedure is used for dust detection, except the cloud mask is not used over land to avoid the misclassification of dust as clouds by the ABI cloud mask.

Quality control assigns one of three confidence levels (low, medium, high) to smoke/dust detections based on the proximity of the UV AAI value to its threshold.

In the smoke/dust detection process over land, the following conditions apply:

- If UV AAI ≥ 4.0 and DSDI ≤ 0.0, thin smoke exists
- If UV AAI ≥ 9.0, DSDI ≤ 1.0, and the Rayleigh-corrected reflectance at 412 nm is between 0.2 and 0.4, thick smoke exists
- If UV AAI ≥ 8.0 and DSDI ≥ 1.0, dust exists

Additionally, any detected smoke pixels undergo a uniformity test using the standard deviation of reflectance at 440 nm within a 3x3 box to screen out residual clouds, i.e., exceeding a value of 0.015 is considered contamination by clouds.

In the smoke/dust detection process over water, the following criteria apply:

- If UV AAI ≥ 5.0, DSDI ≤ -6.0, and the Rayleigh-corrected reflectance at 412 nm is < 0.17, thin smoke exists
- If UV AAI ≥ 10.0 and DSDI ≤ -3.0, thick smoke exists
- If UV AAI ≥ 6.5 and DSDI ≥ -6.0, dust exists

Residual cloud screening over water involves a uniformity test using the standard deviation of reflectance at 865 nm within a 3x3 box, i.e., exceeding a value of 0.015 is considered contamination by clouds.

For sun glint covered areas, where the sun glint angle (η) is less than 40°, AAI values from specular reflection may mimic thin dust, therefore, dust is considered not detectable. The sun glint angle is calculated as:

$$\cos(\eta) = \cos\alpha\cos\beta + \sin\alpha\sin\beta\cos(180 - \varphi) \quad (4)$$

where α is the solar zenith angle, β is the satellite zenith angle and φ is the relative azimuth angle, defined as the difference between the solar azimuth angle and the satellite azimuth angle.