



System for Analysis of Wind Collocations (SAWC): A Novel Archive and Collocation Application for the Intercomparison of Winds from Multiple Observing Platforms

User Manual

Version 1.1.0

July 2023

Prepared for:

U.S. Department of Commerce
National Oceanic and Atmospheric Administration (NOAA)
National Environmental Satellite, Data, and Information Service (NESDIS)

Prepared by:

Katherine E. Lukens, *U. Maryland/ESSIC/CISS, NOAA/NESDIS/STAR*
Kevin Garrett, *NOAA/NWS/OSTI*
Kayo Ide, *U. Maryland*
David Santek, *U. Wisconsin-Madison/CIMSS*
Brett Hoover, *NOAA/NWS/NCEP/EMC, Lynker Technologies*
David Huber, *NOAA/OAR/WPO/EPIC, Redline Performance Solutions, LLC*
Ross N. Hoffman, *U. Maryland/ESSIC/CISS, NOAA/NESDIS/STAR*
Hui Liu, *U. Maryland/ESSIC/CISS, NOAA/NESDIS/STAR*

Version Description Record

DOCUMENT TITLE: <i>System for Analysis of Wind Collocations (SAWC): A Novel Archive and Collocation Application for the Intercomparison of Winds from Multiple Observing Platforms User Manual</i>		
Revision	Date	Description of Change
1.0.0	April 15, 2023	Initial Release of User Manual
1.0.1	April 21, 2023	<i>Bug Fixes:</i> Corrected Plotting Program's use of input path for archived index files. Corrected AMV type maps to show observations at all longitudes.
1.1.0	July 15, 2023	<i>Updates:</i> Updated application installation section. Converted HLOS_error to m/s in Aeolus Rayleigh-clear NetCDF files (<i>original</i> (non-reprocessed) only: Sep 2018 - April 2020). Added missing observations to Aircraft NetCDF data files. Optimized Collocation and Plotting Programs to improve performance. Added a Height/Pressure vs Longitude plot type to the Plotting Program. <i>Bug Fixes:</i> Various bug fixes to the Plotting Program.

Preface

This document comprises the National Oceanic and Atmospheric Administration (NOAA) National Environmental Satellite, Data, and Information Service (NESDIS) publication of the System for Analysis of Wind Collocations (SAWC) *User Manual (UM)*.

This User Manual provides important information for wind observation users. It gives an overview of the System for Analysis of Wind Collocations (SAWC), including the wind observation archive, its datasets, and variables available. It also provides step-by-step instructions on the installation and operation of the collocation application developed for use in conjunction with the archived wind datasets.

The document will be updated as required.

Table of Contents

How to Use this Manual	1
1. Introduction	2
2. The Archive	3
2.1 Archive Contents.....	3
2.1.1 Source Datasets.....	3
2.1.2 Collocation Index Files and Application.....	4
2.2 Directory Paths.....	4
2.2.1 Paths to Aeolus Source Data.....	6
2.2.2 Paths to Non-Aeolus Data.....	6
2.3 File Name Formats.....	7
2.3.1 Aeolus File Names.....	7
2.3.2 Non-Aeolus File Names.....	8
2.3.3 Collocation Index File Names.....	9
2.4 Supplemental Information.....	10
3. Dataset Descriptions	11
3.1 Aeolus.....	11
3.2 Aircraft.....	14
3.3 AMVs.....	15
3.4 Loon.....	16
3.5 Sonde.....	17
4. Collocation Application	18
4.1 Overview.....	18
4.1.1 Data Acquisition.....	19
4.1.2 Collocation of Winds.....	19
4.1.3 Analysis and Visualization.....	21
4.2 Installation.....	33
4.3 Running the Application.....	36
4.3.1 Collocation Program.....	36
4.3.1.1 Order of Processing Steps.....	36
4.3.1.2 Steps for Running the Collocation Program.....	37
4.3.1.3 Advanced Modifications: Step-by-step Examples.....	38
4.3.2 Plotting Program.....	41
4.3.2.1 Order of Processing Steps.....	41
4.3.2.2 Steps for Running the Plotting Program.....	42
4.3.2.3 Advanced Modifications: Step-by-step Examples.....	42
Appendices	45
A1. Acronyms and Abbreviations.....	45
A2. Tips, Tricks, and Other Useful Information.....	46
Publications List	47
References	48

List of Figures

Figure 1: Spatial coverage of Aeolus observations for a 24-h period (2019-09-01). Colored dots denote the observation locations.....	11
Figure 2: Spatial coverage of aircraft observations for a 24-h period (2019-09-01). Colored dots denote the observation locations.....	15
Figure 3: AMV spatial coverage for a 24-h period (2019-09-01). Colored dots denote individual satellite-derived AMV observations with quality indicator (QI) > 80%. Each item in the legend lists the satellite name (top), the number of AMVs from the satellite (bottom left), and the percentage of all AMVs the satellite's count represents (bottom right in parentheses).....	15
Figure 4: Spatial coverage of Loon stratospheric balloon observations for Sept-Dec 2019. Colored dots denote the observation locations.....	16
Figure 5: Spatial coverage of sonde observations for a 24-h period (2019-09-01). Colored dots denote the observation locations.....	17
Figure 6: Flow chart illustrating the end-to-end process for using the archive and collocation application. Note that the user interface begins at Step 2 "Collocation of Winds".....	18
Figure 7: Flow chart illustrating the order of processing steps for the collocation program, beginning with the Main Bash Script, which contains the main parameters the user can/should modify.....	36
Figure 8: Flow chart illustrating the order of processing steps for the plotting program, beginning with the Main Bash Script, which contains most parameters the user can modify.....	41

List of Tables

Table 1: General archive directory abbreviations, paths, and descriptions.....	4
Table 2: Paths to all SAWC components (source datasets, index files, and collocation application). Additionally, key variables that are available per dataset are listed.....	5
Table 3: Index file name abbreviations, available options per abbreviation, and corresponding descriptions.....	9
Table 4: List of L2B processor versions as indicated by Baseline number. The operational start date of each Baseline and relevant details are also provided.....	12
Table 5: List of parameters the user can modify in the main bash script for the collocation program (MAINSCRIPT_for__Collocation.bash).....	19
Table 6: List of parameters the user can modify in the main bash script for the plotting program (MAINSCRIPT_for__PlotMain.bash).....	22
Table 7: List of types of figures generated by the plotting program.....	24
Table 8: List of file name abbreviations for figure files generated by the plotting program.....	25
Table 9: General collocation application directory abbreviations, full paths, and descriptions....	35

Acknowledgments

The authors thank ESA, ECMWF, the NCEP/EMC Obsproc team, and Loon, LLC for providing the data; Peter Marinescu for his contributions towards converting Aeolus BUFR data into 6-hour NCEP prepBUFR for the purpose of data assimilation in NOAA NWP; Chia Moeller for her contribution towards the initial development of tools for analysis and visualization; Iliana Genkova, Jim Jung, Jaime Daniels, Cathy Thomas, and Emily Liu for their expertise on NCEP prepBUFR and winds in NOAA NWP; the Aeolus Cal/Val teams, Mike Hardesty, Erin Jones, Chris Barnet, and Sebastian Bley for their scientific and technical expertise; the SSEC at the University of Wisconsin (UW)-Madison for their support in creating and managing the archive; and Sid Boukabara for his leadership and support of this project.

The UW-Madison S4 supercomputing system (Boukabara et al., [2016](#)) was used in this work.

The authors acknowledge support from the NOAA/NESDIS Office of Projects, Planning, and Acquisition (OPPA) Technology Maturation Program (TMP) through CICS and the Cooperative Institute for Satellite Earth System Studies (CISESS) at the University of Maryland (UMD)/Earth System Science Interdisciplinary Center (ESSIC) (grant nos. NA14NES4320003 and NA19NES4320002) and the Cooperative Institute for Meteorological Satellite Studies (CIMSS) at UW-Madison (grant no. NA20NES4320003).

How to Use this Manual

Inside Each Section

Section 1 provides an introduction to the System for Analysis of Wind Collocations (SAWC).

Section 2 describes the components of SAWC, including the structure of the wind observation archive, its contents, file name formats, and how to access it.

Section 3 describes the wind datasets available in the archive.

Section 4 provides an overview of the collocation application, including installation and how to run the collocation and plotting programs.

How to Get Help

To get help regarding any aspect of SAWC, send an email to the following person:

- Katherine E. Lukens (katherine.lukens@noaa.gov)

1. Introduction

The 2017-2027 decadal survey by the National Academies Press designated the observation requirement for highly accurate atmospheric wind observations as a top priority in the science community (National Academies of Sciences, Engineering, and Medicine, [2018](#)). Numerous studies have been conducted to address this need over the last few decades (e.g., Velden and Holmlund, [1998](#); Velden and Bedka, [2009](#); Bormann et al., [2003](#); Genkova et al., [2008](#), [2010](#); Santek et al., [2014](#), [2019](#), [2021](#), [2022](#); Rani et al., [2022](#); Borde et al., [2016](#), [2019](#); Martin et al., [2021](#); Hoffman et al., [2021](#); Lukens et al., [2022](#)). Many involve AMV comparisons, with each study using widely varying collocation criteria. Further, it is usually up to the researchers to acquire the data and develop their own analysis tools. These steps can be time consuming and can hinder progress, particularly for projects assigned limited periods of performance.

To quantify the satisfaction of the wind requirement and to facilitate the intercomparison of winds from multiple platforms in support of the evaluation of wind dataset characteristics, a collaboration began between NOAA/NESDIS/STAR, UMD/ESSIC/CISESS, and UW-Madison/CIMSS to establish the System for Analysis of Wind Collocations (SAWC). SAWC presents a novel centralized public archive of global 3D wind observations from multiple sources, identified pairings between the wind datasets, and a collocation application developed for their intercomparison. The potential capability of SAWC is wide-ranging, from product validation and observation error characterization to advancements in data assimilation and the global Earth observing architecture.

This document summarizes the components of SAWC: (1) the wind observation archive that contains individual wind datasets and files containing matches (collocations) between them, and (2) the collocation application developed for their intercomparison, including its installation, operation, and examples of its utility.

2. The Archive

The archive

./wind-datasets

contains all SAWC components, including the wind observation datasets (hereafter, source datasets), index files, and the collocation application. All source datasets span the entire lifetime of Aeolus (September 2018 through Present Day) with the exception of Loon which covers 2011 to January 2021 when the Loon project ended.

The archive is maintained on the S4 supercomputer at UW-Madison/SSEC and is managed by NOAA/NESDIS/STAR. Its public server is updated every 3 months (January 15, April 15, July 15, and October 15), and its contents are publicly available for download via FTP and online (URL):

FTP: <ftp.ssec.wisc.edu/wind-datasets>

URL: <https://bin.ssec.wisc.edu/wind-datasets>

Note that SAWC will move to a new home later in 2023. As such, the access links above will change accordingly. An announcement about this is forthcoming.

2.1 Archive Contents

2.1.1 Source Datasets

The archive is home to several source datasets each containing wind observations. These datasets include:

- [Aeolus](#) Level-2B (L2B) winds (de Kloe et al., [2020](#))
- [Aircraft](#) observations (e.g., Moninger et al., [2003](#))
- [Atmospheric motion vectors \(AMVs\)](#) (e.g., Velden et al., [1997](#), [2005](#); Cotton et al., [2020](#)) derived from geostationary and polar-orbiting satellite imagery
- Observations from [Loon](#), a stratospheric balloon network that is no longer operational (Rhodes and Candido, [2021](#))
- [Sonde](#) observations (e.g., Durre et al., [2006](#))

Aeolus data are provided by the European Space Agency (ESA). Aircraft, AMV, and sonde observations are provided by NCEP and are processed by the NCEP/EMC Obsproc team prior to the conversion to NetCDF. Note that all aircraft, AMV, and sonde datasets contain only unrestricted observations. Loon stratospheric balloon observations are provided by Loon, LLC. [Section 2.2](#) lists the paths to each dataset, and [Section 3](#) describes the datasets.

2.1.2 Collocation Index Files and Application

Unique collocation techniques were developed to match the winds based on specified criteria and generate collocation index files containing array indices of the wind matches as well as the collocation differences pertaining to each pair. Index files for select pairings are available in the archive.

Collocation index files are NetCDF files output from the collocation program. One file is created per Driver-Dependent dataset pairing, where the “Driver” is the reference dataset and is defined as the dataset with which all other “Dependent” datasets are collocated. Each index file contains array indices of collocated winds, i.e., array locations of matched winds, between the Driver and Dependent source data files used for collocation. There are two index arrays per file, one pertaining to the Driver and the other a Dependent dataset; these arrays are the same length, although the lengths may vary between files. In addition, each index file contains the collocation distances and differences in time and height/pressure for each pair of collocated winds and are the same length as the corresponding index arrays. Finally, each index file contains the dataset names and paths as well as the collocation criteria (maximum allowable difference in time, height/pressure, and distance) used to find matches and create the index file. Note that by saving indices and not a copy of the collocated data itself, disk space is saved.

In addition, a plotting program consisting of several visualization and statistical analysis tools has been developed to be used in conjunction with the collocation program. The collocation application, which includes both collocation and plotting programs, is available for download.

[Section 2.2](#) lists the locations of the index files and collocation application within the archive, and [Section 4](#) provides more details about the collocation process.

2.2 Directory Paths

Table 1 lists the archive’s main directories and directory abbreviations used in this document. Table 2 lists the paths to all available datasets within the archive, including key variables and available file formats.

The source datasets and the index files are stored by date within their respective directories. Their paths are listed in Sections [2.2.1](#) and [2.2.2](#).

Table 1: General archive directory abbreviations, paths, and descriptions.

Table 1 Path Abbreviation	Full Path (if applicable)	Description
`\${HOME}	<code>./wind-datasets</code>	Home directory
`\${MAIN_AEOLUS}	`\${HOME}/aeolus-dataset	Main Aeolus dataset directory

Table 1		
Path Abbreviation	Full Path (if applicable)	Description
`\${MAIN_ATMOS}`	`\${HOME}/atmos-nc-dataset`	Main conventional dataset directory containing non-Aeolus wind datasets, index files, and the collocation application
`\${YYYY}`		Year, 4 digits (e.g., 2022)
`\${MM}`		Month, 2 digits (e.g., 01)
`\${DD}`		Day, 2 digits (e.g., 01)
`\${hh}`		Hour, 2 digits (possible range: 00-23)
`\${mm}`		Minute, 2 digits (possible range: 00-59)
`\${ss}`		Second, 2 digits (possible range: 00-59)

Table 2: Paths to all SAWC components (source datasets, index files, and collocation application). Additionally, key variables that are available per dataset are listed.

Table 2			
Dataset	Path to Dataset	File Format(s)	Key Variables Available
Aeolus	`\${MAIN_AEOLUS}`	NetCDF, BUFR, and Earth Explorer (EE) format	Year, Month, Day, Hour, Latitude, Longitude, Height, Pressure, HLOS Wind Velocity, Azimuth Angle
Aircraft	`\${MAIN_ATMOS}/aircraft`	NetCDF converted from NCEP prepBUFR	Year, Month, Day, Hour, Latitude, Longitude, Height, Wind Speed, Wind Direction
AMV	`\${MAIN_ATMOS}/AMV`	NetCDF converted from NCEP prepBUFR	Year, Month, Day, Hour, Latitude, Longitude, Pressure, Wind Speed, Wind Direction
Loon	`\${MAIN_ATMOS}/Loon`	NetCDF converted from NCEP prepBUFR	Year, Month, Day, Hour, Latitude, Longitude, Height, Pressure, u-/v-components of Wind, Wind Direction
Sonde	`\${MAIN_ATMOS}/sonde`	NetCDF converted from NCEP prepBUFR	Year, Month, Day, Hour, Latitude, Longitude, Height, Pressure, Wind Speed, Wind Direction
Index Files	`\${MAIN_ATMOS}/collocation/index_files`	NetCDF	Year, Month, Day, Hour, Indices of Collocated Winds
Collocation Application	`\${MAIN_ATMOS}/collocation/application`	One tar file for every version of the application	N/A

2.2.1 Paths to Aeolus Source Data

Aeolus winds are provided in daily files (spanning ~24 hours) and are stored by month. Their general location is:

`${MAIN_AEOLUS}/${DataFormat}/${DataType}/${YYYY}/${MM}`

There are three options for **`${DataFormat}`**:

1. **bufr**. This format is the Binary Universal Form for the Representation of meteorological data (BUFR).
2. **EE-format**. This format is ESA's Earth Explorer (EE) binary format.
3. **netcdf**. This format is NetCDF.

There are two options for **`${DataType}`**:

1. **original**. "Original" refers to the L2B winds processed using the original operational L2B processor version at the time of retrieval. Within this directory the data are stored by date, with the full path being:

`${MAIN_AEOLUS}/${DataFormat}/original/${YYYY}/${MM}/${DD}`

2. **reprocessed**. "Reprocessed" refers to the L2B winds that have been reprocessed using a different L2B processor version than what was originally used at the time of retrieval. Within this directory the data are first stored by the L2B processor version used to reprocess the data, and then by date, with the full path being:

`${MAIN_AEOLUS}/${DataFormat}/reprocessed/${Processor}/${YYYY}/${MM}/${DD}`

At the time of writing, there are three options for **`${Processor}`**: **2B10**, **2B11**, and **2B14**. "2B" refers to Aeolus Level-2B data. The numbers after "2B" indicate the version or "Baseline" of the L2B processor used to reprocess the data. More information on Baselines can be found in [Section 3.1](#).

Within the **bufr** directory, there is a third option for **`${DataType}`**:

3. **trim3hr**. This directory contains Aeolus BUFR data that has been reformatted to NCEP prepBUFR, with each file containing data centered around one of the four synoptic times (0000, 0600, 1200, 1800 UTC). There are two subsets of Aeolus data available in this format:

- a. Original data for the year 2020, with the full path being:

`${MAIN_AEOLUS}/bufr/trim3hr/original/${YYYY}/${MM}/${DD}`

- b. Reprocessed data using Baseline 10 (2B10) spanning June-December 2019, with the full path being:

`${MAIN_AEOLUS}/bufr/trim3hr/reprocessed/2B10/${YYYY}/${MM}/${DD}`

2.2.2 Paths to Non-Aeolus Data

Non-Aeolus winds as well as collocation index files are provided in 6-hourly NetCDF files (centered around the NWP analysis hours 00, 06, 12, and 18 UTC) and are stored by day. Their general location is:

`${MAIN_ATMOS}/${Dataset}/${YYYY}/${MM}/${DD}`

There are several options for **Dataset**:

- **Aircraft.** This directory contains in situ atmospheric observations reported at aircraft flight level, with the full path being:
\${MAIN_ATMOS}/aircraft/\${YYYY}/\${MM}/\${DD}
- **AMV.** This directory contains atmospheric motion vectors derived from both geostationary and polar Earth-orbiting satellite imagery, with the full path being:
\${MAIN_ATMOS}/AMV/\${YYYY}/\${MM}/\${DD}
- **Loon.** This directory contains in situ atmospheric observations reported at balloon level in the upper troposphere/lower stratosphere, with the full path being:
\${MAIN_ATMOS}/Loon/\${YYYY}/\${MM}/\${DD}
- **sonde.** This directory contains in situ atmospheric observations reported in profile by sonde instrumentation, with the full path being:
\${MAIN_ATMOS}/sonde/\${YYYY}/\${MM}/\${DD}
- **collocation.** This directory contains:
 - Indices of identified pairings of winds between Driver and Dependent datasets, with the full path being:
\${MAIN_ATMOS}/collocation/index_files/\${YYYY}/\${MM}/\${DD}
 - The collocation application developed to collocate and statistically compare the winds, with the full path being:
\${MAIN_ATMOS}/collocation/application

2.3 File Name Formats

This section provides structures and examples of file names available within the archive.

2.3.1 Aeolus File Names

Within the **bufr** directory, the file names are listed as:

AE_OPER_ALD_B_N_2B_\${AEOLUS_DATETIME}_0001.BUFR

The path abbreviation **AEOLUS_DATETIME** indicates the date/time range of observations per BUFR file and refers to:

\${YYYY}\${MM}\${DD}T\${hh}\${mm}\${ss}_\${YYYY}\${MM}\${DD}T\${hh}\${mm}\${ss}

Example file name:

AE_OPER_ALD_B_N_2B_20211231T233917_20220101T011004_0001.BUFR

Within the **bufr/trim3hr** directory, the file name format is:

Aeolus_ESA_\${YYYY}\${MM}\${DD}\${hh}.ncep

Example file name:

Aeolus_ESA_2020010100.ncep

Within the **EE-format** directory, there are two files for each date/time range, a header file in xml (*.HDR) and a binary data file (*.DBL). The file names are listed as:

AE_OPER_ALD_U_N_2B_ \${AEOLUS_DATETIME}_0001.DBL

AE_OPER_ALD_U_N_2B_ \${AEOLUS_DATETIME}_0001.HDR

Example file names:

AE_OPER_ALD_U_N_2B_20220101T010941_20220101T032904_0001.DBL

AE_OPER_ALD_U_N_2B_20220101T010941_20220101T032904_0001.HDR

Within the **netcdf** directory, the file names are listed as:

Aeolus.L2B. \${AEOLUS_WIND_TYPE}.1day. \${YYYY}\${MM}\${DD}.nc

There are two options for **\${AEOLUS_WIND_TYPE}**:

1. **MieCloud**. This refers to winds derived from Mie-cloudy (aerosol) backscatter retrievals.
2. **RayClear**. This refers to winds derived from Rayleigh-clear (molecular) backscatter retrievals.

Example file names:

Aeolus.L2B.MieCloud.1day.20220101.nc

Aeolus.L2B.RayClear.1day.20220101.nc

2.3.2 Non-Aeolus File Names

Within the **aircraft** directory, the file names are listed as:

gdas. \${YYYY}\${MM}\${DD}\${hh}.aircft.tm00.bufr_d.nc4

Example file name:

gdas.2022010100.aircft.tm00.bufr_d.nc4

Within the **AMV** directory, the file names are listed as:

gdas. \${YYYY}\${MM}\${DD}\${hh}.satwnd.tm00.bufr_d.nc4

Example file name:

gdas.2022010100.satwnd.tm00.bufr_d.nc4

Within the **Loon** directory, the file names are listed as:

Loon_ \${YYYY}\${MM}\${DD}\${hh}.nc4

Example file name:

Loon_2020010100.nc4

Within the **sonde** directory, the file names are listed as:

gdas. \${YYYY}\${MM}\${DD}\${hh}.adpupa.tm00.bufr_d.nc4

Example file name:

gdas.2022010100.adpupa.tm00.bufr_d.nc4

2.3.3 Collocation Index File Names

Within the `$(MAIN_ATMOS)/collocation/index_files` directory, the file names are listed as:

`index.{$YYYY}{$MM}{$DD}{$hh}.drv_{$DRV_DSET}_{$iQC}.dset1_{$DEP_DSET}_{$iQC}.nc4`

Example file name:

`index.2019090200.drv_Aeolus_MieCloud_ReprocB11__QC.dset1_Aircraft_NoQC.nc4`

This example index file contains array indices (i.e., array locations in the source data files) of collocated winds. Winds from the Dependent (dset1) dataset (Aircraft) are collocated with the Driver (drv) dataset (Aeolus Mie-cloudy reprocessed using the Baseline 11 version of the L2B processor). Note that:

- The Driver winds are QC'd prior to collocation.
- The Dependent winds are not QC'd prior to collocation.

Table 3 lists and defines the options for `$(DRV_DSET)`, `$(DEP_DSET)`, and `$(iQC)` in the index file names.

Table 3: Index file name abbreviations, available options per abbreviation, and corresponding descriptions.

Table 3 Abbreviation	Available Options	Description
<code>\$(DRV_DSET)</code>	<code>Aeolus_{\$AEOLUS_WIND_TYPE}_{\$COL_AEOLUS_TYPE}</code>	<i>Driver dataset:</i> Obs locations to which obs from the Dependent dataset are collocated
<code>\$(DEP_DSET)</code>	Aircraft AMV_NCEP Loon Sonde	<i>Dependent dataset:</i> Obs that are collocated to the Driver obs locations
<code>\$(COL_AEOLUS_TYPE)</code>	Orig	Aeolus winds processed with L2B processor version at time of retrieval
	Reproc <code>\$(BASELINE)</code>	Aeolus winds reprocessed with different (typically updated) L2B processor (indicated by Baseline number <code>\$(BASELINE)</code>) than that at time of retrieval
<code>\$(BASELINE)</code>	<code>B##</code>	L2B processor Baseline number, where <code>##</code> indicates the 2-digit Baseline number
<code>\$(iQC)</code>	NoQC	Prior to collocation, no QC is applied to the dataset (name that precedes '___' in file name)
	QC	Prior to collocation, QC is applied to the dataset (name that precedes '___' in file name)

2.4 Supplemental Information

README files and other relevant information accompany the source data and collocation index files contain additional information about the datasets, including data exclusion periods for the user to consider. Users are encouraged to familiarize themselves with these files, which can be found in:

- **`${MAIN_AEOLUS}/docs`**
- **`${MAIN_ATMOS}/docs`**

All available variables for the NetCDF source datasets as well as the index files are defined in a separate document called [SAWC Variable Definitions](#).

- Note that for the datasets acquired from NCEP (aircraft, AMV, sonde), each NetCDF variable is associated with a BUFR Mnemonic Code, and the values of several of these variables correspond to integers with specific meanings listed in corresponding BUFR Code/Flag Tables. To look up this information for a NetCDF variable, simply find the associated BUFR mnemonic code (fifth column in the Variable Definitions document), look up that mnemonic code at the *BUFR Table B* website https://www.nco.ncep.noaa.gov/sib/jeff/bufrtab_tableb.html and click on the *Code Table* or *Flag Table* link under the *Units* column.

3. Dataset Descriptions

This section describes the source datasets available within the wind archive.

3.1 *Aeolus*

Aeolus L2B winds are derived from backscattering signals retrieved by the Aladin Flight Module A (FM-A) (or the backup FM-B) Doppler wind lidar instrument on a dawn/dusk orbit. The satellite completes one orbit approximately every 92 min, with global coverage achieved every 7 days. Winds are observed in profile along the laser's Horizontal Line-of-Sight (HLOS). See the Aeolus Operations website for more information (ESA [2020](#)).

Two main wind regimes are derived using ESA's L2B processor: (1) Rayleigh-clear (molecular backscattering) representing winds in clear scenes, and (2) Mie-cloudy (aerosol backscattering) representing winds in cloudy scenes. Aeolus spatial coverage for one day is shown in Fig. 1.

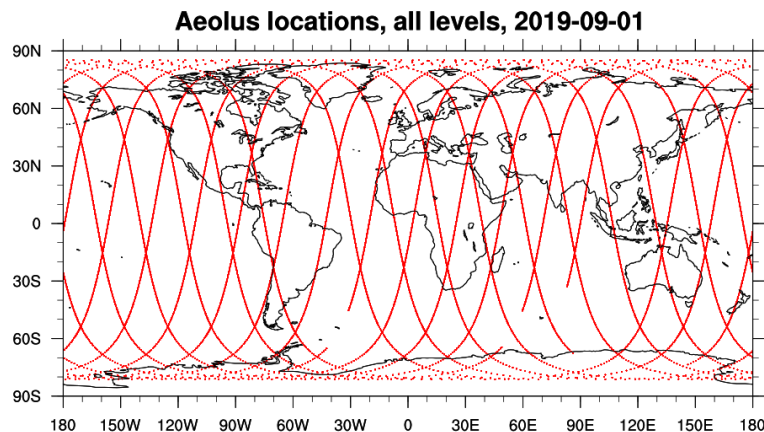


Figure 1: Spatial coverage of Aeolus observations for a 24-h period (2019-09-01). Colored dots denote the observation locations.

The L2B processor has been updated several times over the course of Aeolus's lifetime (September 2018 - April 2023); these updates are designated as Baselines and are implemented operationally for certain periods of time. Table 4 lists the different Baselines available in the archived Aeolus dataset and the dates on which they became operational. Note that because different processors have been implemented over the course of Aeolus' lifetime, the data quality and metadata available may change over time.

Table 4: List of L2B processor versions as indicated by Baseline number. The operational start date of each Baseline and relevant details are also provided.

Table 4 Baseline	Operational Start Date	Details
B01	2018-09-03	<ul style="list-style-type: none"> • FM-A instrument used for retrieval • Valid for 2018-09-03 through 2018-10-16
B02	2018-09-03	<ul style="list-style-type: none"> • FM-A instrument used for retrieval • Valid for 2018-09-03 through 2019-05-16
B03	2019-05-16	<ul style="list-style-type: none"> • FM-A instrument used for retrieval • Valid for 2019-05-16 through 2019-06-14
B04	2019-06-14	<ul style="list-style-type: none"> • FM-A instrument used for retrieval • Valid for 2019-06-14 through 2019-06-16
B05	2019-06-28	<ul style="list-style-type: none"> • Switch from FM-A to FM-B initiated on 2019-06-16 due to FM-A power loss • FM-B instrument used for retrieval • Valid for 2019-06-28 through 2019-09-05
B06	2019-06-28	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2019-06-28 through 2019-10-31
B07	2019-10-31	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2019-10-31 through 2020-04-02
B08	2020-04-02	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2020-04-02 through 2020-04-20
B09	2020-04-20	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2020-04-20 through 2020-07-09
B10	2020-04-20	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2020-04-20 through 2020-10-08 • Usage of M1 mirror temperature correction in L2B processor enabled • Regeneration of B09 into B10 completed (<i>B09 is equivalent to B10</i>) • Used for first reprocessing campaign: reprocess winds from 2019-06-28 through 2019-12-31
B11	2020-10-08	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2020-10-08 through 2021-05-26 • Used for second reprocessing campaign: reprocess winds from 2019-06-28 through 2020-10-06
B12	2021-05-26	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2021-05-26 through 2021-12-06
B13	2021-12-06	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2021-12-06 through 2022-03-29
B14	2022-03-29	<ul style="list-style-type: none"> • FM-B instrument used for retrieval • Valid for 2022-03-29 through 2022-09-13 • Used for third reprocessing campaign: reprocess winds from 2018-08-31 through 2019-06-16, when FM-A instrument was operational

Table 4 Baseline	Operational Start Date	Details
B15	2022-09-13	<ul style="list-style-type: none"> Completed switch from FM-B back to FM-A on 2022-11-29 FM-B instrument used for retrieval for 2022-09-13 through 2022-10-04 FM-A instrument used for retrieval for 2022-11-27 onward
B16	2023-04-18	<ul style="list-style-type: none"> FM-A instrument used for retrieval Valid for 2023-04-18 through 2023-04-30 Mie-cloudy winds have been significantly improved with this baseline

Additional notes on Aeolus:

- EE-format data are from ESA's public Aeolus Online Dissemination System: <https://aeolus-ds.eo.esa.int/oads/access/> (ESA, 2020)
 - The EE files can be converted into ASCII using ESA's [CODA](#) tool.
- *NetCDF files:*
 - NetCDF files are converted from EE format with metadata retained. Because only EE data flagged as "valid" are saved as NetCDF, there may be some dates without NetCDF files but with EE files.
 - Aeolus Rayleigh wind observations come with accompanying reference pressures and temperatures derived from ECMWF model forecasts (e.g., Dabas et al., 2008; Šavli et al., 2021), whereas Mie winds do not.
 - During the NetCDF conversion, each Mie wind observation is assigned a reference pressure and temperature from the nearest Rayleigh observation. (First, the closest Rayleigh profile to the Mie observation in space and time is identified. Then, the Rayleigh observation within that profile with the smallest vertical distance from the Mie observation is found and the pressure and temperature from that Rayleigh observation is assigned to the Mie observation; note that this only happens if the minimum vertical distance between the observations is ≤ 1.0 km. If there is more than one Rayleigh observation with the same minimum vertical distance to the Mie observation in question, then the pressure/temperature of the Rayleigh observation with the most similar wind speed to the Mie observation is chosen.)
 - Off-orbit profiles are removed prior to conversion to NetCDF. See the document file `_${MAIN_AEOLUS}/docs/README.Offorbit_Profiles` for more information.
 - Duplicate profiles are removed prior to conversion to NetCDF. See the document file `_${MAIN_AEOLUS}/docs/README.Duplicate_Profiles` for more information
 - NetCDF files are daily files covering ~24 hour periods (00:00-23:59 UTC) and are stratified by Aeolus wind regime listed in each file name (e.g., `Aeolus.L2B._${AEOLUS_WIND_TYPE}.1day.20190801.nc`)
 - Aeolus QC flags are added to the NetCDF files based on ESA- and ECMWF-recommended QC decisions (Rennie and Isaksen, 2020). See the document file `_${MAIN_AEOLUS}/docs/README.Quality_Controls` for more information.

- Please be aware that starting with Baseline B10, the units for the estimated HLOS wind error (name key in CODA: "hlos_error_estimate") are now **cm s⁻¹** for both L2B Rayleigh and Mie winds in the EE-format files; prior to Baseline B10, the units were **m s⁻¹**.
 - Note that during the NetCDF conversion process, the estimated HLOS wind error is converted back to **m s⁻¹**. See the document file [\\${MAIN_AEOLUS}/docs/README.Obs_Errors](#) for more information.
- *Data Exclusion Dates:*
 - Certain time periods are known to have degraded data content and are NOT recommended for use in research or operations. See the document file [\\${MAIN_AEOLUS}/docs/README.Data_Exclusion_List](#) for more information.
- *Hot Pixels:*
 - It should be noted that a patch was implemented in the operational processor which corrects pixel intensities for elevated dark signal levels (defined as "hot pixels") on Friday June 14, 2019 at 08:30 UTC. The L2B processing flags winds as invalid if a strong bias from uncorrected hot pixels occurs (ESA, [2021](#)). For more information on hot pixels and their correction, see Weiler et al. ([2021a](#)).
- *Mirror Temperature Corrections:*
 - Small fluctuations in the temperatures across the telescope's primary mirror onboard Aeolus was linked to wind biases as high as 8 m s⁻¹. This systematic error has since been corrected, reducing much of the wind bias. The mirror temperature correction has been implemented in operations since April 2020. For more information, see Weiler et al. ([2021b](#)).
- For additional information on Aeolus, visit:
https://www.esa.int/Applications/Observing_the_Earth/FutureEO/Aeolus
and
<https://space.oscar.wmo.int/satellites/view/aeolus>

3.2 Aircraft

Aircraft observations are reported at flight level in the upper troposphere/lower stratosphere (cruise altitude) and in the ascending and descending legs of each flight. The spatial coverage is regional, mostly in the NH as seen in Fig. 2. Temporal coverage spans from September 2018 - Present Day.

The data are converted to NetCDF from NCEP prepBUFR files ingested by NOAA operations for assimilation. The data are stratified by aircraft type (which are listed in the prepBUFR table containing additional metadata information: [\\${MAIN_ATMOS}/docs/prepbufr_table.aircraft](#)) and include winds and other observations.

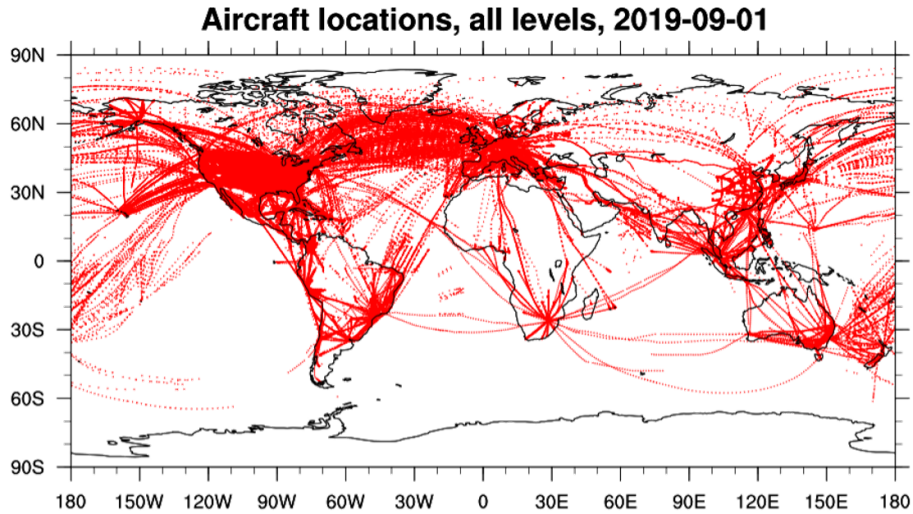


Figure 2: Spatial coverage of aircraft observations for a 24-h period (2019-09-01). Colored dots denote the observation locations.

3.3 AMVs

AMVs are derived from tracking clouds and water vapor features in satellite imagery through time, and the archived AMVs include derived winds from both geostationary (GEO) and polar Earth-orbiting (LEO) satellites. Spatial coverage is near-global at various vertical levels, as shown in Fig. 3. Temporal coverage spans from September 2018 - Present Day.

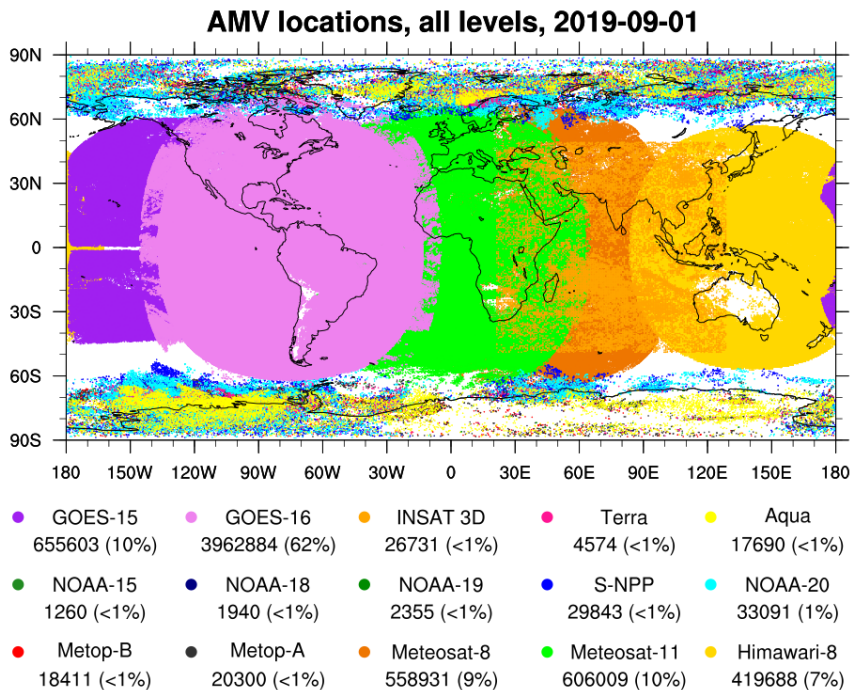


Figure 3: AMV spatial coverage for a 24-h period (2019-09-01). Colored dots denote individual satellite-derived AMV observations with quality indicator (QI) > 80%. Each item in the legend lists the satellite name (top), the number of AMVs from the satellite (bottom left), and the percentage of all AMVs the satellite's count represents (bottom right in parentheses).

AMVs are converted to NetCDF from NCEP prepBUFR files ingested by NOAA operations for assimilation. They can be stratified by the satellite from which they are derived; the full list of possible satellite sources (**README.AMV_Satellite_Identifier_List**) as well as the prepBUFR tables (**prepbufr_table.AMV***) that define certain metadata items can be found in the **/\${MAIN_ATMOS}/docs** directory.

3.4 Loon

Loon was a network of stratospheric superpressure balloons deployed by Loon, LLC, a subsidiary of Google's parent company Alphabet, to provide internet connectivity to regions with limited access. The balloon configurations consisted of an instrument payload platform tethered to a tennis court-sized balloon specifically designed to withstand harsh stratospheric conditions. The balloons remained aloft for months at a time, and a ground team was able to control their movements by remotely adjusting the balloon pressure in order to move the configurations into different airstreams. The spatial coverage is regional (Fig. 4) and depended upon the Loon mission at the time of deployment. In situ atmospheric observations were reported at balloon level (~50-100 hPa or 18-20 km) at a frequency of ~1-20 minutes. Temporal coverage spans from 2011-2021.

A full description of the Loon dataset is available here:

<https://doi.org/10.5281/zenodo.3755988> (Rhodes and Candido, 2021)

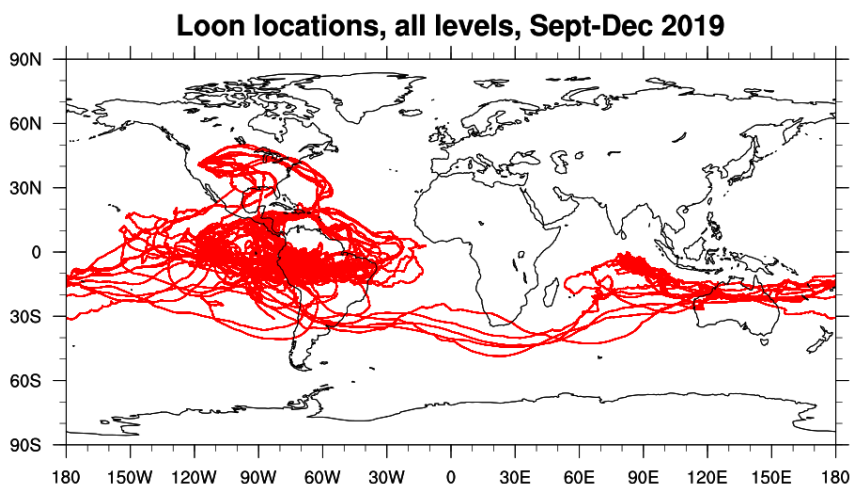


Figure 4: Spatial coverage of Loon stratospheric balloon observations for Sept-Dec 2019. Colored dots denote the observation locations.

A note on Loon:

- Before downloading or using the Loon data, please read the following files located in **/\${MAIN_ATMOS}/docs**:
 - **README.Loan_Flight_Data.pdf**
 - **README.Loan_License_Agreement-Noncommercial_Use.pdf**

3.5 Sonde

Sondes (typically radio-, drop-, or rawinsondes) report in situ observations in profile at balloon level throughout the troposphere and into the lower stratosphere. They are deployed worldwide twice per day: 00 UTC and 12 UTC. The spatial coverage is regional, mostly over land (Fig. 5). Temporal coverage spans from September 2018 - Present Day.

The data are converted to NetCDF from NCEP prepBUFR files ingested by NOAA operations for assimilation and are stratified by sonde type (which are listed in the prepBUFR table `$(MAIN_ATMOS)/docs/prepbuftr_table.sonde`). They include winds and other observations.

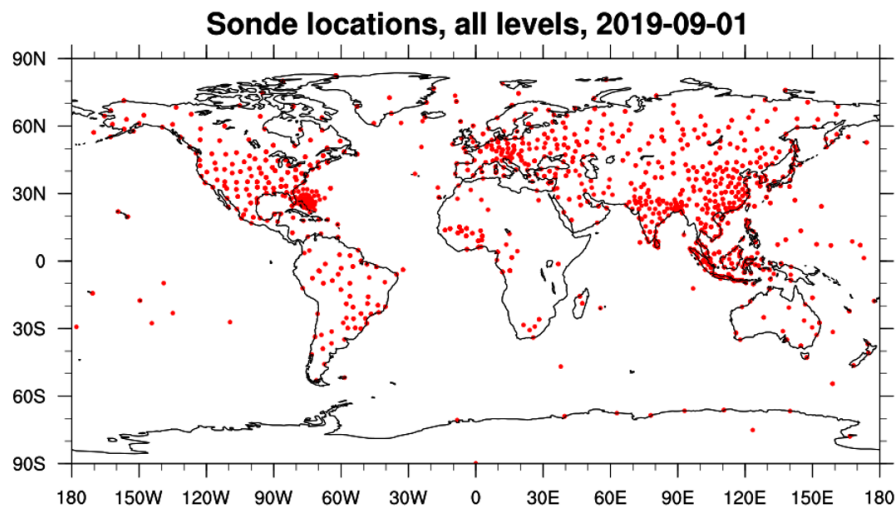


Figure 5: Spatial coverage of sonde observations for a 24-h period (2019-09-01). Colored dots denote the observation locations.

4. Collocation Application

4.1 Overview

The collocation application was developed as part of a collaboration between NOAA/NESDIS/STAR, CISS at UMD, and CIMSS at UW-Madison to address the National Academies requirement for highly accurate winds (National Academies of Sciences, Engineering, and Medicine, [2018](#)). The application is written in Python and is controlled via Bash scripting. A basic understanding of each language is recommended, although not required.

The application has two main components:

1. *Collocation Program*. A program developed to collocate any number of wind datasets using user-specified collocation criteria. Output index files are generated and contain the locations of matched winds in the source data files.
2. *Plotting Program*. A program that uses the index files to extract and statistically and visually compare the collocated winds by calling a suite of plotting functions.

The collocation and plotting programs are designed to be flexible and handle additional datasets not yet available in the archive. Later in this section, details are provided on the installation and utility of the application, including examples of its utility, such as adjusting certain parameters and adding new datasets for the user's own comparisons. If you, the user, wish to officially add a new dataset to the archive, contact the person listed in the section [How to Use this Manual](#).

The approach is composed of three main steps: Data Acquisition, Collocation of Winds, and Analysis and Visualization, as illustrated in Fig. 6.

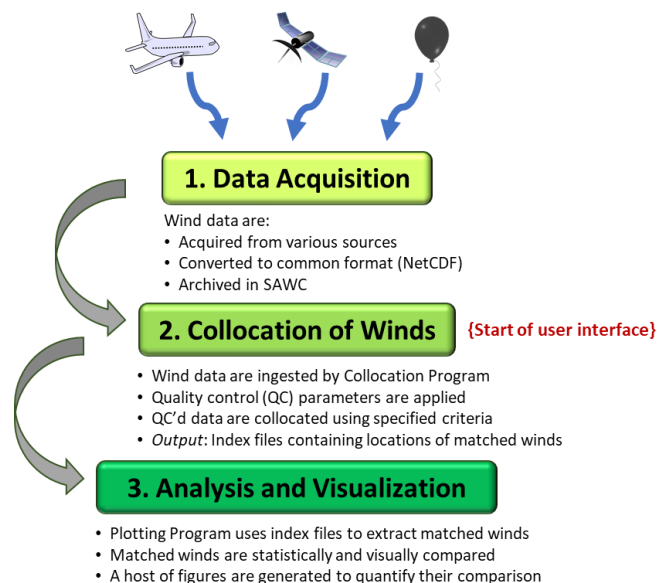


Figure 6: Flow chart illustrating the end-to-end process for using the archive and collocation application. Note that the user interface begins at Step 2 “Collocation of Winds”.

4.1.1 Data Acquisition

Data is acquired from various sources (including NOAA/NCEP, ESA, and Loon) and converted to a common format (NetCDF) before archival on the S4 supercomputer. All data files have corresponding metadata embedded to provide additional information about the observations and the data source. To view a NetCDF file `${FILENAME}`, open a terminal window and enter either of the following commands on the command line:

```
ncl_filedump ${FILENAME} ... to view metadata information
ncview ${FILENAME} ... to view variables graphically
```

Note that the file extension (either `.nc` or `.nc4`, whichever corresponds to the file in question) must be included to execute the above commands.

4.1.2 Collocation of Winds

The wind collocation program collocates wind observations between different datasets. It reads NetCDF files from a minimum of two datasets: one is the Driver (or reference) dataset, and the other is the Dependent dataset. The Driver is the dataset with which all Dependent datasets are collocated. The program allows for one or more Dependent datasets to be collocated against a single Driver dataset.

The user can modify numerous parameters in the main bash script

MAINSRIPT_for__Collocation.bash

that initiates the collocation program, including the date range, datasets to collocate, collocation criteria, and whether quality controls (QC) should be applied. A full list of parameters is provided in Table 5.

Table 5: List of parameters the user can modify in the main bash script for the collocation program (MAINSRIPT_for__Collocation.bash).

Table 5 Parameter	Type	Description
yyyy	Integer	Year. Must be one 4-digit number.
mm	Integer	Month array. Must contain at least one array item.
ddarr	Integer	Day array. Must contain at least one array item.
dir_home	String	Full path to location of MAINSRIPT_for__Collocation.bash
dir_out	String	Full path to location where collocation index files are to be saved.
archive_parent	String	Full path to archive home directory (parent directory of /atmos-nc-dataset and /aeolus-dataset directories).

Table 5 Parameter	Type	Description
dataset_names	String	Names of datasets to be collocated. The first dataset is set as the Driver; all others are set as Dependents. Must contain at least 2 dataset names.
dst_max	Float	Maximum collocation distance. Must have one array item for every Dependent dataset listed in dataset_names . Units = km
prs_max	Float	Maximum log ₁₀ (pressure) difference. Must have one array item for every Dependent dataset listed in dataset_names . Units = log ₁₀ (hPa)
tim_max	Float	Maximum time difference. Must have one array item for every Dependent dataset listed in dataset_names . Units = minutes
hgt_max	Float	Maximum height difference. Must have one array item for every Dependent dataset listed in dataset_names . Units = km
qc_flags	Integer	Quality control (QC) flags for each dataset listed in dataset_names . <i>Options available:</i> 0 (no QC applied), or 1 (QC applied)
amv_qi_flags	Integer	Quality indicator (QI) value for AMV observations. This parameter must be set even if AMVs are not listed in dataset_names . Units = % (percent)
qi_choices	String	QI choice for AMVs from NCEP ("AMV_NCEP"). <i>Options available:</i> NO_FC (default; use QI variable without forecast), or YES_FC (use QI variable with forecast)
n_max	Integer	Number of collocations allowed from a Dependent dataset for every Driver observation. Default = 50
nproc	Integer	Number of processors to use during parallelization. Default = 50
account	String	HPC account. This is the account you the user are allowed to use.
qos	String	HPC QoS. This is for setting SLURM jobs.
partition	String	HPC partition. This is for setting SLURM jobs.
timelimit	String	HPC runtime limit for SLURM jobs.
bline	String	Abbreviation for Aeolus L2B dataset type. <i>Options available:</i> orig (choose dataset processed with original L2B processor at time of retrieval), or B## (choose dataset re-processed with different L2B processor than that used at time of retrieval. ## is a 2-digit number indicating the Baseline number, e.g., B10 = Baseline 10)
dtype	String	Subdirectory based on bline choice. <i>Options available:</i> original (if bline = orig), or reprocessed/2BB (if bline is <u>not</u> orig . BB is the parameter bline)

Table 5 Parameter	Type	Description
{load modules}	–	A series of module load commands needed to run the collocation program. Please check that the following modules are loaded properly as per your HPC system's protocols: Hdf, hdf5, netcdf4, udunits2, ncvview, nco

The collocation approach is based on that developed at CIMSS (e.g., Santek et al., [2021](#)): Observations from one or more Dependent datasets are collocated to the 4D locations (latitude, longitude, height/pressure, and time) of each observation from the Driver dataset. The default collocation criteria are:

- 100 km maximum great circle distance (150 km for sondes)
- 60 minute maximum absolute time difference (90 min for sondes)
- 0.04 maximum absolute $\log_{10}(\text{hPa})$ pressure difference
- 1 km maximum absolute height difference

A default maximum number of 50 Dependent observations can be collocated with a single Driver observation. This means that multiple Dependent observations may be collocated to one Driver observation. The option to “super-ob” these observations is available in the plotting program and is described in [Section 4.1.3](#).

Once the data are read in, QC is applied if the user chooses this option, and then the program compares the QC'd input data from the Driver and Dependent datasets and finds the array locations of the collocated winds within those files. All Dependent observations are compared to each Driver observation, and if a Dependent observation meets all selected collocation criteria, its array location (index) is saved to the corresponding collocation index file. This means that one Dependent observation may be collocated with multiple Driver observations.

At present, the application can handle only point observation datasets. Gridded datasets (e.g., NWP reanalyses) are being considered for inclusion in future iterations of SAWC.

4.1.3 Analysis and Visualization

The plotting program visualizes and statistically analyzes comparisons between collocated winds. For a user-specified range of dates, the program reads in the array locations of collocated winds from the index files and extracts the corresponding wind observations from the source data files.

Once extracted but before the intercomparisons are performed, the winds undergo a mathematical conversion if Aeolus is listed as either a Driver or Dependent dataset. Specifically, non-Aeolus winds are projected onto the Aeolus HLOS direction prior to any analysis. Eqs. (1-4) are used to project non-Aeolus winds y onto the Aeolus HLOS direction x_{dir}^{HLOS} :

$$u_y = -y_{spd} \sin(y_{dir}) \quad (1)$$

$$v_y = -y_{spd} \cos(y_{dir}) \quad (2)$$

$$y_{spd}^{HLOS} = (-u_y \sin(x_{dir}^{HLOS})) + (-v_y \cos(x_{dir}^{HLOS})) \quad (3)$$

$$y_{dir}^{HLOS} = x_{dir}^{HLOS} \quad (4)$$

where y_{spd} and y_{dir} are the non-Aeolus wind speed and direction, respectively, u_y and v_y are the u- and v-components of the non-Aeolus wind, y_{spd}^{HLOS} is the non-Aeolus wind velocity projected onto the HLOS direction, and y_{dir}^{HLOS} is the new direction of the non-Aeolus wind and is equal to the HLOS direction x_{dir}^{HLOS} . Due to the unique geometry of Aeolus, HLOS winds can have both positive and negative values (unlike non-HLOS wind speeds that can only be positive), and as such should be referred to as velocities, not speeds.

After the HLOS projection (if applicable), the Dependent winds are compared with the Driver winds through a series of statistical computations via plotting functions. Numerous figures are generated that quantify the wind intercomparisons, which can be stratified by region, pressure/height level, and AMV wind type (if applicable).

The user can modify numerous parameters in the main bash script that initiates the plotting program

MAINSRIPT_for__Plotting.bash

A full list of parameters that the user can change in this script is provided in Table 6.

Table 6: List of parameters the user can modify in the main bash script for the plotting program (MAINSRIPT_for__Plotting.bash).

Table 6 Parameter	Type	Description
yyyyS	String	Start date year. Must be 4 digits.
mmS	String	Start date month. Must be 2 digits.
ddS	String	Start date day. Must be 2 digits.
hhS	String	Start date hour. Must be 2 digits.
yyyyE	String	End date year. Must be 4 digits.
mmE	String	End date month. Must be 2 digits.
ddE	String	End date day. Must be 2 digits.
hhE	String	End date hour. Must be 2 digits.

Table 6 Parameter	Type	Description
archive_parent	String	Full path to archive home directory (parent directory of /atmos-nc-dataset and /aeolus-dataset directories).
dir_home	String	Full path to location of MAINSRIPT_for__Plotting.bash
dir_in	String	Full path to location where collocation index files are located. <i>Note: Users should not modify this unless the index files they wish to use are located somewhere other than the archive's directory structure.</i>
dep_dset	String	Array of strings listing Dependent dataset names
dep_dset_qc	String	Array of strings listing whether QC was applied to Dependent datasets prior to collocation. Must be the same length as dep_dset . <i>Options available: NoQC (QC was not applied to dataset prior to collocation), and QC (QC was applied).</i>
drv_dset	String	String listing Driver dataset.
drv_dset_qc	String	String indicating whether QC was applied to Driver dataset prior to collocation. Must be the same length as drv_dset . <i>Options available: NoQC (QC was not applied to dataset prior to collocation), and QC (QC was applied).</i>
filename_segment	String	Segment of the collocation index filename listing the required Driver dataset information. USER SHOULD NOT MODIFY THIS PARAMETER.
dir_out	String	Full path to location where all figures are to be saved.
superob_choice	Integer	Choice to super-ob (average) multiple collocations per Driver observation, or use all collocations for statistical analysis. <i>Options available: -1 (use all collocations), or 0 (super-ob)</i>
account	String	HPC account. This is the account you the user are allowed to use.
qos	String	HPC QoS. This is for setting SLURM jobs.
partition	String	HPC partition. This is for setting SLURM jobs.
timelimit	String	HPC runtime limit for SLURM jobs.
{load modules}	–	A series of module load commands needed to run the collocation program. Please check that the following modules are loaded properly as per your HPC system's protocols: Hdf, hdf5, netcdf4, udunits2, ncview, nco, ncl

Notes on super-obbing (*superob_choice=0*):

- The term “super-obbing” in relation to this application refers to the averaging of all matched winds from a Dependent dataset with the same Driver observation, resulting in one Dependent observation (representing the mean of the multiple collocations) per Driver observation. For example, say you are collocating Aeolus winds (Driver) and AMVs (Dependent), and 25 AMVs pass all collocation criteria and are subsequently

collocated with the same Driver observation: If super-obbing is chosen, the 25 AMVs are all averaged together prior to any analysis or visualization taking place.

- Super-obbing the observations reduces the sizes of the matched Driver and Dependent arrays, and may in turn decrease the processing time for analysis and plotting.

All figures generated by the plotting program can be found in the **dir_out** directory. A list of the types of figures generated by the plotting program and their descriptions are provided in Table 7.

Table 7: List of types of figures generated by the plotting program.

Table 7 Start of Figure Name	Description
DENSITY_SCATTER	Density scatterplots comparing one collocated variable from one Dependent dataset to the Driver. <i>Note:</i> If the differences are statistically significant (at 95% level), a line of text will appear on the figure stating as much.
HIST	Histograms displaying the collocation differences (time differences, pressure/height difference, or collocation distance) for all Dependent datasets vs the Driver.
MAP_2Dcontour	2D cylindrical equidistant map displaying a variable summed (observation number density) or averaged (wind speed, difference) per cell on a lat/lon grid within a specified vertical layer, for one Dependent dataset compared to the Driver.
MAP_2Dpoints	2D cylindrical equidistant map displaying a variable at all observed lat/lon locations (not transformed to a lat/lon grid) within a specified vertical layer, for one Dependent dataset compared to the Driver.
MAP_3D	3D projection of a variable at all observed 3D locations, for one Dependent dataset compared to the Driver.
MAP_CE.Match_Locations	2D cylindrical equidistant map of observation locations, for all datasets.
MAP_Ortho.Match_Locations	Same as MAP_CE but on a polar projection.
MAP_Ortho_NorthPole	Same as MAP_2Dcontour but on a polar projection over the North Pole.
MAP_Ortho_SouthPole	Same as MAP_2Dcontour but on a polar projection over the South Pole.
MAP_Rotate	Rotating globe displaying some variable (e.g., number density) for one dataset.
MAP_Rotate.Match_Locations	Rotating globe displaying observation locations, for all datasets.
SCATTER	Scatterplot comparing one collocated variable from all Dependent datasets to the Driver.
STATSvsWSPD	Wind speed difference, standard deviation (SD) of the difference, and SD of each dataset vs Driver wind speed, with corresponding observation count, per Driver wind speed bin for all Dependent datasets. <i>Note:</i> If differences between the Driver and a Dependent dataset are statistically significant (at 95% level) for a certain Driver wind speed bin, a solid circle will appear at that speed along the Dependent difference line.

Table 7 Start of Figure Name	Description
TIME_SERIES	Time series displaying the mean wind differences (Dependent - Driver), correlation coefficients, SDs of the differences, root mean square differences (RMSDs), and observation counts per day (24-h period) for all Dependent datasets.
VERT_DISTR	Mean vertical distribution of wind speed differences (Dependent - Driver), SDs of the differences, and the corresponding observation counts, per vertical bin for all Dependent datasets. <i>Note:</i> If the differences between the Driver and a Dependent dataset are statistically significant (at 95% level) at a certain vertical level, a solid circle will appear at that level along the corresponding Dependent dataset profile line.
VERTvsLAT_LonRange	Vertical cross section (pressure/height vs latitude) displaying a variable summed (e.g., observation number density) or averaged (e.g., wind speed, difference) per grid cell within a specified longitude range, for one Dependent dataset compared to the Driver.
VERTvsLAT_ZonalMean	Same as VERTvsLAT_LonRange but averaged for all longitudes (i.e., zonal mean).
VERTvsLON_MeridMean	Vertical cross section (pressure/height vs latitude) displaying a variable summed (e.g., observation number density) or averaged (e.g., wind speed, difference) per grid cell averaged over all latitudes (i.e., meridional mean), for one Dependent dataset compared to the Driver.
VERTvsTIME	Pressure/height vs time figure displaying a variable summed (e.g., observation number density) or averaged (e.g., wind speed, difference) per grid cell for each day (24-h period), for one Dependent dataset compared to the Driver

Table 8 lists all possible abbreviations that are used to make up a figure's file name.

Table 8: List of file name abbreviations for figure files generated by the plotting program.

Table 8 Abbreviation	Description
#{All_DEPs}	A list of all Dependent datasets delimited by an underscore “_” <i>Example:</i> Aircraft_Loon_Sonde
#{DateRange}	Period of time the figure represents. There is one option: #{StartDate}-#{EndDate}
#{DSETabbrev}	Indicates the dataset that is plotted. There are several options: <ul style="list-style-type: none"> ○ DRV: The Driver listed as #{DRV_DSET} in the file name is plotted ○ DEP: The Dependent listed as #{DEP_DSET} in the file name is plotted ○ DEP-DRV: The difference (#{DEP_DSET} minus #{DRV_DSET}) is plotted
#{LatPOV}	Latitude on which a rotating globe's point-of-view is centered.
#{Level}	Represents the mid-level of the vertical layer in question. <i>Examples:</i> 7.0km, 70.0hPa

Table 8 Abbreviation	Description
#{LonRange}	Range of longitudes over which variables are summed or averaged in vertical cross section figures. <i>Example: 30.0-60.0</i>
#{Pole}	Refers to the polar perspective of the projection. There are two options: NorthPole , SouthPole
#{Region}	Indicates region. There are four options: Global , NH , TR , SH , where NH stands for Northern Hemisphere, TR for Tropics, and SH for Southern Hemisphere.
#{ScatterPlots}	The plotting program generates two types of scatter plots: DENSITY_SCATTER , SCATTER
#{StartDate} , #{EndDate}	#{StartDate} and #{EndDate} have the same form #{YYYY}#{MM}#{DD}#{hh} and correspond to the start date and end date, respectively, that the user specifies in the main plotting bash script. <i>Example: 2019090100</i>
#{SuperobOption}	#{SuperobOption} indicates whether the observations were super-obbed prior to statistical analysis and plotting. There are two options: <ul style="list-style-type: none"> ○ SuperOb: The observations were super-obbed ○ AllMatches: The observations were not super-obbed; rather, all matches were included in the analysis
#{VertLatType}	Refers to the type of pressure/height vs latitude plot. There are two options: LonRange#{LonRange} , ZonalMean

File name formats for each type of figure are defined below. Note that the abbreviation **#{Variable}** is unique for each figure type.

- There are two file name formats for the figures **DENSITY_SCATTER** or **SCATTER**:
 1. **#{ScatterPlots}.#{Variable}.#{Region}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperobOption}.png**
 2. **#{ScatterPlots}.#{Variable}.AMVtype.#{Region}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperobOption}.png**
 - The first displays the comparison of **#{Variable}** between the Driver and one Dependent dataset.
 - The second displays the comparison of **#{Variable}** corresponding to an AMV type between the Driver and one Dependent dataset if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - There are several options for **#{Variable}**: **HLOS_Velocity**, **WindSpeed**, **Height**, **Pressure**.
 - All other abbreviations have been previously defined.

- There are two file name formats for **HIST** figures:
 1. **HIST.Match_** $\{\text{Variable}\}$ **._** $\{\text{Region}\}$ **._** $\{\text{DateRange}\}$ **._x_** $\{\text{DRV_DSET}\}$ **._y_** $\{\text{DEP_DSET}\}$ **.png**
 2. **HIST.Match_** $\{\text{Variable}\}$ **.AMVtype.** $\{\text{Region}\}$ **._** $\{\text{DateRange}\}$ **._x_** $\{\text{DRV_DSET}\}$ **._y_** $\{\text{DEP_DSET}\}$ **.png**
 - The first displays $\{\text{Variable}\}$ for all Dependent datasets specified in the main plotting bash script.
 - The second displays $\{\text{Variable}\}$ for all AMV types, and is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - Here, $\{\text{Variable}\}$ corresponds to the collocation difference variables. There are several options:
 - **Distance**: Collocation distance between Dependent and Driver
 - **HeightDiff**: Height difference (Dependent - Driver), if the vertical variable is *height*
 - **PressureDiff**: Pressure difference (Dependent - Driver), if the vertical variable is *pressure*
 - **TimeDiff**: Time difference (Dependent - Driver)
 - All other abbreviations have been previously defined.

- There are four file name formats for **MAP_2Dcontour** figures:
 1. **MAP_2Dcontour.Nobs_Density.** $\{\text{DSETabbrev}\}$ **._** $\{\text{DateRange}\}$ **._x_** $\{\text{DRV_DSET}\}$ **._y_** $\{\text{DEP_DSET}\}$ **._** $\{\text{SuperobOption}\}$ **.png**
 2. **MAP_2Dcontour.** $\{\text{Variable}\}$ **._** $\{\text{DSETabbrev}\}$ **._** $\{\text{Level}\}$ **._** $\{\text{DateRange}\}$ **._x_** $\{\text{DRV_DSET}\}$ **._y_** $\{\text{DEP_DSET}\}$ **._** $\{\text{SuperobOption}\}$ **.png**
 3. **MAP_2Dcontour.Nobs_Density.** $\{\text{DSETabbrev}\}$ **.AMVtype.** $\{\text{DateRange}\}$ **._x_** $\{\text{DRV_DSET}\}$ **._y_** $\{\text{DEP_DSET}\}$ **._** $\{\text{SuperobOption}\}$ **.png**
 4. **MAP_2Dcontour.** $\{\text{Variable}\}$ **._** $\{\text{DSETabbrev}\}$ **.AMVtype.** $\{\text{Level}\}$ **._** $\{\text{DateRange}\}$ **._x_** $\{\text{DRV_DSET}\}$ **._y_** $\{\text{DEP_DSET}\}$ **._** $\{\text{SuperobOption}\}$ **.png**
 - The first displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a lat/lon grid.
 - The second displays a collocated variable (e.g., wind speed, wind difference) averaged per cell on a lat/lon grid within a specified vertical layer. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The third displays the number density of collocations for an AMV type per cell on a lat/lon grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The fourth displays the wind speed or wind difference for an AMV type averaged per cell on a lat/lon grid within a specified vertical layer. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - There are several options for $\{\text{Variable}\}$: **Wind** (i.e., speed or velocity), **Wind_Diff**, **Wind_SDdiff**
 - All other abbreviations have been previously defined.

- There are two file name formats for **MAP_2Dpoints** figures:
 1. **MAP_2Dpoints.{\$Variable}.{\$DSETabbrev}.{\$Level}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperobOption}.png**
 2. **MAP_2Dpoints.{\$Variable}.{\$DSETabbrev}.AMVtype.{\$Level}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperobOption}.png**
 - The first displays a collocated variable (e.g., wind speed, wind difference) corresponding to all observations within a specified vertical layer.
 - The second displays all collocated AMVs of one type within a specified vertical layer.
 - There are two options for **{\$Variable}**: **Wind** (i.e., speed or velocity), **Wind_Diff**
 - All other abbreviations have been previously defined.

- There are two file name formats for **MAP_3D** figures:
 1. **MAP_3D.{\$Variable}.{\$DSETabbrev}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperobOption}.png**
 2. **MAP_3D.AMVtype.{\$Variable}.{\$DSETabbrev}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperobOption}.png**
 - The first displays a collocated variable (e.g., wind speed, wind difference) corresponding to all observations at their 3-dimensional locations.
 - The second displays all collocated AMVs of one type at their 3-dimensional locations.
 - There are several options for **{\$Variable}**: **Wind** (i.e., speed or velocity), **Wind_Diff**
 - All other abbreviations have been previously defined.

- There are two file name formats for **MAP_CE.Match_Locations** figures:
 1. **MAP_CE.Match_Locations.{\$DateRange}.x_{\$DRV_DSET}.y_{\$All_DEPs}.{\$SuperobOption}.png**
 2. **MAP_CE.Match_Locations.AMVtype.{\$DateRange}.x_{\$DRV_DSET}.y_{\$All_DEPs}.{\$SuperobOption}.png**
 - The first displays the lat/lon locations of collocated observations from all Dependent datasets.
 - The second displays the lat/lon locations of all available types of collocated AMVs.
 - All other abbreviations have been previously defined.

- There are two file name formats for **MAP_Ortho.Match_Locations** figures:
 1. **MAP_Ortho.Match_Locations_{\$Pole}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$All_DEPs}.{\$SuperobOption}.png**
 2. **MAP_Ortho.Match_Locations_{\$Pole}.AMVtype.{\$DateRange}.x_{\$DRV_DSET}.y_{\$All_DEPs}.{\$SuperobOption}.png**
 - The first displays the lat/lon locations of collocated observations from all Dependent datasets on the corresponding polar projection **{\$Pole}**.

- The second displays the lat/lon locations of all available types of collocated AMVs on the corresponding polar projection **#{Pole}**.
 - All other abbreviations have been previously defined.
- There are four file name formats for **MAP_Ortho_#{Pole}**:
 1. **MAP_Ortho_#{Pole}.Nobs_Density.#{DSETabbrev}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperrobOption}.png**
 2. **MAP_Ortho_#{Pole}.#{Variable}.#{DSETabbrev}.#{Level}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperrobOption}.png**
 3. **MAP_Ortho_#{Pole}.Nobs_Density.#{DSETabbrev}.AMVtype.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperrobOption}.png**
 4. **MAP_Ortho_#{Pole}.#{Variable}.#{DSETabbrev}.AMVtype.#{Level}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperrobOption}.png**
 - The first displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a lat/lon grid.
 - The second displays a collocated variable (e.g., wind speed, wind difference) averaged per cell on a lat/lon grid within a specified vertical layer. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The third displays the number density of collocations for an AMV type per cell on a lat/lon grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The fourth displays the wind speed or wind difference for an AMV type averaged per cell on a lat/lon grid within a specified vertical layer. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - There are several options for **#{Variable}**: **Wind** (i.e., speed or velocity), **Wind_Diff**, **Wind_SDdiff**
 - All other abbreviations have been previously defined.
 - There are two file name formats for **MAP_Rotate**:
 1. **MAP_Rotate.Nobs_Density.#{DSETabbrev}.centerlat.#{LatPOV}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperrobOption}.gif**
 2. **MAP_Rotate.Nobs_Density.#{DSETabbrev}.AMVtype.centerlat.#{LatPOV}.#{DateRange}.x_#{DRV_DSET}.y_#{DEP_DSET}.#{SuperrobOption}.gif**
 - The first displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a lat/lon grid on a rotating globe centered on the latitude **#{LatPOV}**.
 - The second displays the number density of collocations for an AMV type per cell on a lat/lon grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset on a rotating globe centered on the latitude **#{LatPOV}**.
 - All other abbreviations have been previously defined.

- There are two file name formats for **MAP_Rotate.Match_Locations**:
 1. **MAP_Rotate.Match_Locations.centerlat.{\$LatPOV}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$All_DEPs}.{\$SuperrobOption}.gif**
 2. **MAP_Rotate.Match_Locations.AMVtype.centerlat.{\$LatPOV}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$All_DEPs}.{\$SuperrobOption}.gif**
 - The first displays the lat/lon locations of collocated observations from all Dependent datasets on a rotating globe centered on the latitude **{\$LatPOV}**.
 - The second displays the lat/lon locations of all available types of collocated AMVs on a rotating globe centered on the latitude **{\$LatPOV}**.
 - All other abbreviations have been previously defined.

- There are two file name formats for the figures **STATSvsWSPD**:
 1. **STATSvsWSPD.{\$Variable}.DEP-DRV.{\$Region}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperrobOption}.png**
 2. **STATSvsWSPD.{\$Variable}.DEP-DRV.AMVtype.{\$Region}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperrobOption}.png**
 - The first displays the difference of **{\$Variable}** between the Driver and one Dependent dataset, the SD of that difference, and collocated observation count per wind speed/velocity bin.
 - The second displays the difference of **{\$Variable}** corresponding to an AMV type between the Driver and one Dependent dataset, the SD of that difference, and collocated observation count per wind speed/velocity bin. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - There are two options for **{\$Variable}**:
 - **HLOS_Velocity_Diff**: This is only created if the Aeolus dataset is selected as either the Driver or a Dependent dataset.
 - **WindSpeed_Diff**: This is only created if the Aeolus dataset is neither the Driver nor a Dependent dataset.
 - All other abbreviations have been previously defined.

- There are two file name formats for the figures **TIME_SERIES**:
 1. **TIME_SERIES.{\$Variable}.DEP-DRV.{\$Region}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperrobOption}.png**
 2. **TIME_SERIES.{\$Variable}.DEP-DRV.AMVtype.{\$Region}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperrobOption}.png**
 - The first displays the difference of **{\$Variable}** between the Driver and all Dependent datasets, the SD of that difference, correlation coefficients between the Driver and Dependent datasets, RMSD, and collocated observation count per wind speed/velocity bin.
 - The second displays the difference of **{\$Variable}** corresponding to all AMV types between the Driver and one Dependent dataset, the SD of that difference, correlation coefficients between the Driver and Dependent datasets, RMSD, and collocated observation count per wind speed/velocity bin. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.

- There are two options for **`\${Variable}`**:
 - **HLOS_Velocity_Diff**: This is only created if the Aeolus dataset is selected as either the Driver or a Dependent dataset.
 - **WindSpeed_Diff**: This is only created if the Aeolus dataset is neither the Driver nor a Dependent dataset.
 - **Pressure_Diff**: This is created using all Dependent datasets with *pressure* as the vertical variable.
 - **Height_Diff**: This is created using all Dependent datasets with *height* as the vertical variable.
 - All other abbreviations have been previously defined.
- There are four file name formats for **VERT_DISTR**:
 1. **VERT_DISTR.HLOS_Velocity_Diff.DEP-DRV.`\${Region}`.`\${DateRange}`.x_`\${DRV_DSET}`.y_`\${All_DEPs}`.`\${SuperrobOption}`.png**
 2. **VERT_DISTR.WindSpeed_Diff.DEP-DRV.`\${Region}`.`\${DateRange}`.x_`\${DRV_DSET}`.y_`\${All_DEPs}`.`\${SuperrobOption}`.png**
 3. **VERT_DISTR.HLOS_Velocity_Diff.DEP-DRV.AMVtype.`\${Region}`.`\${DateRange}`.x_`\${DRV_DSET}`.y_`\${All_DEPs}`.`\${SuperrobOption}`.png**
 4. **VERT_DISTR.WindSpeed_Diff.DEP-DRV.AMVtype.`\${Region}`.`\${DateRange}`.x_`\${DRV_DSET}`.y_`\${All_DEPs}`.`\${SuperrobOption}`.png**
 - The first displays the average HLOS wind velocity difference, SD of the difference, and observation number density per pressure/height bin (prescribed in **MAIN.plotting_code.py** and **MAIN.plotting_code.AMVtypes.py**). This is only created if the Aeolus dataset is selected as either the Driver or a Dependent dataset.
 - The second displays the average wind speed difference, SD of the difference, and observation number density per pressure/height bin. This is only created if the Aeolus dataset is neither the Driver nor a Dependent dataset.
 - The third displays the average HLOS wind velocity difference, SD of the difference, and observation number density per pressure/height bin corresponding to each AMV type. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset, and the Aeolus dataset is selected as either the Driver or a Dependent dataset.
 - The fourth displays the average HLOS wind velocity difference, SD of the difference, and observation number density per pressure/height bin corresponding to each AMV type. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset, and the Aeolus dataset is neither the Driver nor a Dependent dataset.
 - All other abbreviations have been previously defined.

- There are four file name formats for **VERTvsLAT_LonRange** and **VERTvsLAT_ZonalMean**:
 1. **VERTvsLAT_** $\{\text{VertLatType}\}$ **.Nobs_Density.** $\{\text{DSETabbrev}\}$ **.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 2. **VERTvsLAT_** $\{\text{VertLatType}\}$ **.** $\{\text{DSETabbrev}\}$ **.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 3. **VERTvsLAT_** $\{\text{VertLatType}\}$ **.Nobs_Density.** $\{\text{DSETabbrev}\}$ **.AMVtype.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 4. **VERTvsLAT_** $\{\text{VertLatType}\}$ **.** $\{\text{DSETabbrev}\}$ **.AMVtype.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 - The first displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a pressure/height vs latitude grid.
 - The second displays a collocated variable (e.g., wind speed, wind difference) averaged per cell on a pressure/height vs latitude grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The third displays the number density of collocations for one AMV type per cell on a pressure/height vs latitude grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The fourth displays the wind speed or wind difference for one AMV type averaged per cell on a pressure/height vs latitude grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - There are two options for $\{\text{VertLatType}\}$:
 - **LonRange** $\{\text{LonRange}\}$: Average over the longitudes in range $\{\text{LonRange}\}$
 - **ZonalMean**: Average over all longitudes
 - There are several options for $\{\text{Variable}\}$: **Wind** (i.e., speed or velocity), **Wind_Diff**
 - All other abbreviations have been previously defined.
- There are four file name formats for **VERTvsLON_MeridMean**:
 5. **VERTvsLON_MeridMean.Nobs_Density.** $\{\text{DSETabbrev}\}$ **.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 6. **VERTvsLON_MeridMean.** $\{\text{DSETabbrev}\}$ **.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 7. **VERTvsLON_MeridMean.Nobs_Density.** $\{\text{DSETabbrev}\}$ **.AMVtype.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 8. **VERTvsLON_MeridMean.** $\{\text{DSETabbrev}\}$ **.AMVtype.** $\{\text{DateRange}\}$ **.x_** $\{\text{DRV_DSET}\}$ **.y_** $\{\text{DEP_DSET}\}$ **.** $\{\text{SuperobOption}\}$ **.png**
 - The first displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a pressure/height vs longitude grid.
 - The second displays a collocated variable (e.g., wind speed, wind difference) averaged per cell on a pressure/height vs longitude grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.

- The third displays the number density of collocations for one AMV type per cell on a pressure/height vs longitude grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - The fourth displays the wind speed or wind difference for one AMV type averaged per cell on a pressure/height vs longitude grid. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - There are several options for **Variable**: **Wind** (i.e., speed or velocity), **Wind_Diff**
 - All other abbreviations have been previously defined.
- There are two file name formats for **VERTvsTIME**:
 1. **VERTvsTIME.Nobs_Density.{\$DSETabbrev}.{\$Region}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperobOption}.png**
 2. **VERTvsTIME.Nobs_Density.{\$DSETabbrev}.AMVtype.{\$Region}.{\$DateRange}.x_{\$DRV_DSET}.y_{\$DEP_DSET}.{\$SuperobOption}.png**
 - The first displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a pressure/height vs time grid.
 - The second displays the number of density of collocations for either the Driver or a Dependent dataset per cell on a pressure/height vs time grid, for one AMV type. This is only created if an AMV dataset is selected as either the Driver or a Dependent dataset.
 - Here, there are only two options for **{\$DSETabbrev}**: **DRV**, **DEP**

4.2 Installation

The collocation application should be installed on an HPC compute node and not a gateway node. For example, on S4 install the application within your home directory on s4-submit.

It is recommended that you download a copy of the data you will need to your local machine prior to running the collocation application. The data should be located in a directory that has the same home directories (**MAIN_ATMOS** and **MAIN_AEOLUS**) and subdirectory structure as the archive.

Follow these steps to install the collocation application:

1. Download the ***.tar.gz** file from the archive in **MAIN_ATMOS/collocation/application** and place it in the directory where you plan to install and use the collocation application.
2. Unzip the file with the following command:


```
gunzip *.tar.gz
```
3. Untar the file with the following command:


```
tar -xf *.tar
```
4. You should now be able to see and access the **SAWC_Collocation_Application_v###** directory (where **###** refers to the version number; it is recommended that the latest

version be used). The main directories within the collocation application are listed in Table 9.

5. Now it is time to set up your conda environment. You should only have to do this once on a single HPC system; even if you download additional versions of SAWC in the future, you should not need to do this again, unless of course you intend to use SAWC on a different HPC system.
 - a. In your **.bashrc** file in your home directory (or **.cshrc** or **.kshrc**), initialize your conda environment (below is bash syntax, and the paths apply to the S4 system. You will need to change the **/opt** paths to fit what is available on your system):

```
__conda_setup="$('/opt/miniconda/3.8-s4/bin/conda' 'shell.bash' 'hook' 2> /dev/null)"
if [ $? -eq 0 ]; then
  eval "$__conda_setup"
else
  if [ -f "/opt/miniconda/3.8-s4/etc/profile.d/conda.sh" ]; then
    . "/opt/miniconda/3.8-s4/etc/profile.d/conda.sh"
  else
    export PATH="/opt/miniconda/3.8-s4/bin:$PATH"
  fi
fi
unset __conda_setup
```

- b. Save your **.bashrc** file and source it to enact your new changes:


```
source .bashrc
```
 - c. Load the miniconda 2.7 module (below is the syntax used on S4):


```
module load miniconda/2.7-base
```
 - d. Build upon your initialized conda environment using the file **SAWC_conda_env.yml** located in **`\${MAIN_COL}`/src**:


```
conda env create -f `${MAIN_COL}`/src/SAWC_conda_env.yml
```
 - e. Activate this environment:


```
conda activate SAWC_conda_env
```

 Note: If the above command doesn't work, try:


```
source activate SAWC_conda_env
```
 - f. Check that the **SAWC_conda_env** environment is active (note the double dash -- in the command below):


```
conda info --envs
```

 The environment **SAWC_conda_env** should be listed on your screen.

You are now ready to run the application! (See [Section 4.3](#))

Table 9: General collocation application directory abbreviations, full paths, and descriptions.

Table 9 Path Abbreviation	Full Path (if applicable)	Description
<code>\${MAIN_COL}</code>	<code>/SAWC_Collocation_Application_v###</code>	Collocation application main directory. Contains main bash scripts and SLURM job scripts. Note that <code>v###</code> indicates the version number of the collocation application
<code>\${SRC}</code>	<code>\${MAIN_COL}/src</code>	Directory containing all source code

4.3 Running the Application

The collocation application should be run on an HPC compute node and not a gateway node. For example, on S4 run the application within your home directory on s4-submit.

4.3.1 Collocation Program

4.3.1.1 Order of Processing Steps

The collocation program's order of processing steps is illustrated in Fig. 7.

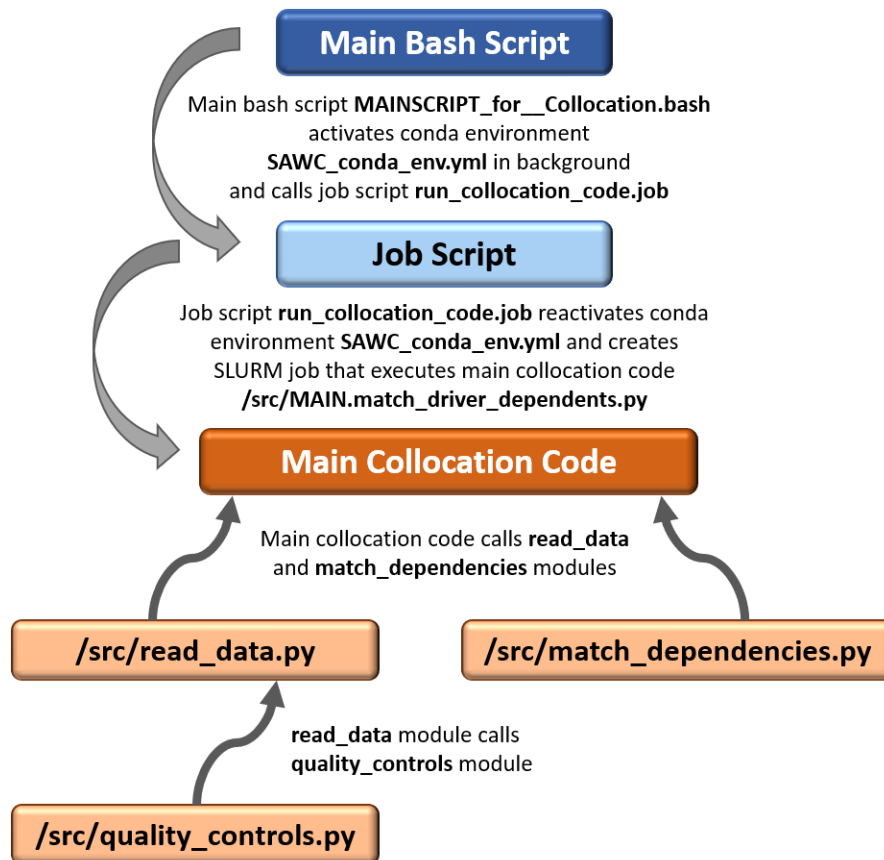


Figure 7: Flow chart illustrating the order of processing steps for the collocation program, beginning with the Main Bash Script, which contains the main parameters the user can/should modify.

The script `MAINSCRIPT_for__Collocation.bash` is the main script that performs the collocation between wind datasets, and is located in `/${MAIN_COL}`. This script activates in the background the conda environment (`SAWC_conda_env`, as per the `*.yml` file) used to run the collocation program, and calls the script `run_collocation_code.job` that initiates all jobs using SLURM, with each job name beginning with `COL*`. Each job calls the main collocation program `MAIN.match_driver_dependents.py` located in the `/src` directory. The main collocation

program reads in the wind data to be collocated, matches each Dependent dataset with the Driver, and writes the matched indices to one collocation index file in NetCDF per Driver-Dependent combination for every date `${YYYY}${MM}${DD}${hh}` in the user-specified time period. One job is initiated for each NWP analysis time (00, 06, 12, or 18 UTC), so you will have at minimum 4 jobs in your queue for a single day (8 jobs for a single day if Aeolus is either the Driver or a Dependent dataset: 4 for Rayleigh-clear comparisons, and 4 for Mie-cloudy comparisons). The corresponding log file names begin with **LOG_COL_***.

The **read_data.py** module contains all functions that read in each of the wind datasets.

The **match_dependencies.py** module contains all dependent functions for the collocation program.

The **quality_controls.py** module contains all functions that perform quality controls on each of the wind datasets prior to collocation.

4.3.1.2 Steps for Running the Collocation Program

Follow these steps to run the collocation program for matching n Dependent datasets against one Driver dataset:

1. Open **MAINSRIPT_for__Collocation.bash** and set all parameters listed in Table 5.
2. Run the program by entering on the command line

./MAINSRIPT_for__Collocation.bash

All jobs run in the background. You can check the status of the jobs by typing into the command line

```
squeue -u ${username} -o "%.18i %.9P %.30j %.8u %.8T %.10M %.9I %.6D %R"
```

where **\${username}** is your username on the HPC account you are using.

Tip: Create an alias for the **squeue** command in your **.rc** file in your home directory so you don't have to type it out every time! For example, in a **.bashrc** file, add the following:

```
alias myq='squeue -u YourUserName -o "%.18i %.9P %.30j %.8u %.8T %.10M %.9I %.6D %R"'
```

3. Once a job completes, you will find the corresponding index file in the output directory **dir_out** that you previously prescribed in **MAINSRIPT_for__Collocation.bash**.

4.3.1.3 Advanced Modifications: Step-by-step Examples

Below are some step-by-step examples of offline modifications the user can make to the collocation program source code:

1. *Including a wind dataset that is not available in the archive*
 - a. Make sure the new dataset files are all in NetCDF, preferably 6-hr format centered around the NWP analysis times 00, 06, 12, and 18 UTC.
 - b. In **read_data.py**, copy and paste one of the functions to the end of the script.
Suggestion: If the new data in each NetCDF file are organized into groups (as is common for HDF or NetCDF-4), copy the **read_aircraft** function. If not, copy the **read_loon** function.
 - i. Rename the new function as **read_\${NewDataset}**, where **\${NewDataset}** is the name the user chooses to represent the new dataset.
 - ii. Change the ***_path**, ***_path_B4**, and ***_path_A** variables to the paths to your new dataset for the current date, date before, and date after, respectively. Follow the syntax of these paths in the original **read_*** function.
Note: If all dates are located in the same directory, then ***_path**, ***_path_B4**, and ***_path_A** are the same.
 - iii. Change/add the variable names ***_var** as they are listed in the new dataset NetCDF files.
 - iv. Below the line **# Load dataset** are lines of code corresponding to each variable name, and these lines are grouped. Add/remove lines of code from each of these groups following the variable names listed in *Step iii*.
Note: Check that all new temporary variables are deleted (**del**) where appropriate.
 - c. In **MAIN.match_driver_dependents.py**:
 - i. Add the following line to the *Import python modules* section at the top of the script:


```
from read_data import read_${NewDataset}
```
 - ii. In the if-else block below the line **# Define and load DRIVER dataset**, add another **elif** condition corresponding to the new dataset just above the **else** condition. Follow the syntax of the other **elif** conditions.
Notes:
 1. Check that the output variables within the new **elif** condition (left side of the equal sign =) match those listed in the last line of the new function **read_\${NewDataset}** (on **return** line) in **read_data.py**.
 2. Check that the input variables within the new **elif** condition (within the parentheses after the function name on the right side of the equal sign) match those listed in parentheses in the first line of the new function **read_\${NewDataset}** in **read_data.py** (to the right of the function name).

- iii. Repeat *Step ii* but this time in the if-else block below the line **# Define dependent datasets...** Follow the syntax of the other **elif** conditions in this if-else block.

2. Adding/modifying QC functions

- a. In **quality_controls.py**, copy and paste the function **qc_amv** to the end of the script.
 - i. Rename the new function as **qc_\${NewDataset}**, where **\${NewDataset}** is the name the user chooses to represent the new dataset.
 - ii. The **idx** variable contains indices of the data where the input conditions within **np.where()** are **True**; these input conditions are the QC criteria. Change the QC criteria here to correspond to the new dataset.

Notes:

1. Write the QC criteria so that you KEEP the data that pass QC. For example, if your max time difference criterion is 60 minutes, the criterion should be: **np.where(time_max < 60)**
2. Each condition should be separated by a set of parentheses
3. A star * between each condition is equivalent to “and”
4. A plus sign + between each condition is equivalent to “or”

Example:

```
idx = np.where( ((prs < pmax)*(prs > pmin))+ (wspd > wspdmin) )
```

does the following:

It finds all instances **idx** where ((pressure **prs** is less than maximum pressure **pmax**) AND (**prs** is greater than minimum pressure **pmin**)) OR (wind speed **wspd** is greater than minimum wind speed **wspdmin**)).

- iii. **qc_list** is a string that lists in words the QC criteria you added in *Step ii*. Change this string to match the QC criteria you added in *Step ii*.
- b. In the function **read_\${Dataset}** in **read_data.py** that corresponds to the QC function you just made,
 - i. Add the following line to the *Import python modules* section at the top of the script:


```
from quality_controls import qc_${Dataset}
```
 - ii. In the **if runtime=="match"** block toward the end of the function, add under **if bool_qc** a call to the new function **qc_\${Dataset}**, following the syntax below:

```
tindexes, qc_list = qc_${Dataset}(...input variables for new QC function...)
```

```
sindexes = np.asarray(tindexes)
```

```
indexes = sindexes.flatten()
```

```
d_lat = tdset_lat[indexes]
```

Be sure to add a line like **d_lat=...** for each variable in the dataset.

Notes:

1. If you are adding a QC function for a new dataset that is not currently available in the archive, make sure you first complete Item #1 in this section ([Section 4.3.1.3](#)), then come back to this step.
2. The output variables for the new QC function are **tindexes** and **qc_list**.
3. Check that the input variables (within the parentheses after the function name on the right side of the equal sign in the above example) match those listed in parentheses in the first line of the new function **qc_\${NewDataset}** in **quality_controls.py** (to the right of the function name).

4.3.2 Plotting Program

4.3.2.1 Order of Processing Steps

A flow chart illustrating the plotting program's order of processing steps is shown in Fig. 8.

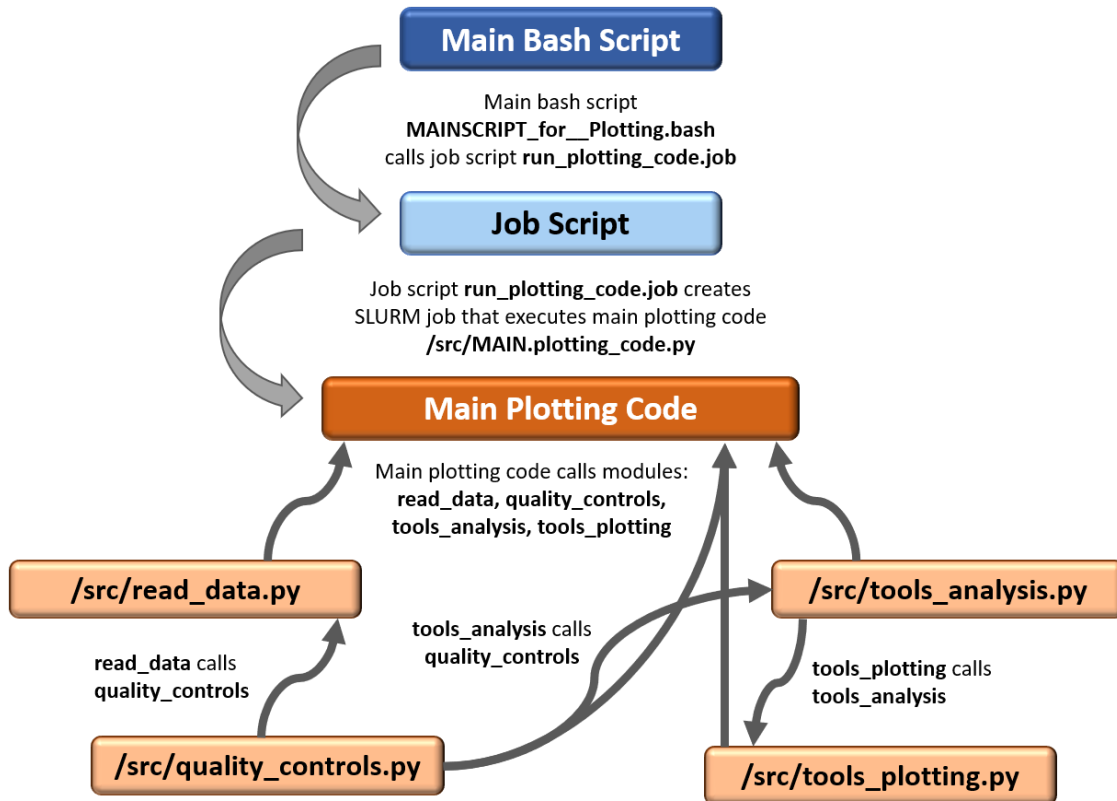


Figure 8: Flow chart illustrating the order of processing steps for the plotting program, beginning with the Main Bash Script, which contains most parameters the user can modify.

The script **MAINSRIPT_for__Plotting.bash** is the main script that statistically and visually compares the Driver and Dependent datasets, and is located in `#{MAIN_COL}`. This script calls the script **run_plotting_code.job** that initiates one job using SLURM, with the job name beginning with **PLOT***. The job calls the main plotting program **MAIN.plotting_code.py** (or **MAIN.plotting_code.AMVtypes.py** which is identical to **MAIN.plotting_code.py** except that it analyzes and plots the different AMV types instead of each dataset) located in the `/src` directory. The main plotting program reads in the matched indices from the collocation index files within the user-specified time period, and uses them to extract the collocated winds from the source data files, then compares the collocated Driver and Dependent winds through visualization and statistical analyses, and generates a suite of figures that quantify the comparisons. The corresponding log file name begins with **LOG_PLOT_***.

The **tools_analysis.py** module contains all functions that perform statistical, mathematical, and other analyses to compare the wind datasets.

The **tools_plotting.py** module contains all functions that plot the comparisons as output figures.

4.3.2.2 Steps for Running the Plotting Program

Follow these steps to run the plotting program for comparing n Dependent datasets against one Driver dataset:

1. Open **MAINSRIPT_for__Plotting.bash** and set all parameters listed in Table 6 (exception: **filename_segment**).
2. Run the program by entering on the command line

```
./MAINSRIPT_for__Plotting.bash
```

The job runs in the background. You can check the status of the job by typing into the command line

```
squeue -u ${username} -o "%.18i %.9P %.30j %.8u %.8T %.10M %.9I %.6D %R"
```

where **\${username}** is your username on the HPC account you are using.

Tip: Create an alias for the **squeue** command in your **.*rc** file in your home directory so you don't have to type it out every time! For example, in a **.bashrc** file, add the following:

```
alias myq='squeue -u YourUserName -o "%.18i %.9P %.30j %.8u %.8T %.10M %.9I %.6D %R"
```

3. You will find the figures as they are being generated in the output directory **dir_out** that you prescribed in **MAINSRIPT_for__Plotting.bash**.

4.3.2.3 Advanced Modifications: Step-by-step Examples

Below are a few examples of more advanced offline modifications the user can make to the plotting program.

1. *Enabling the plotting program to include in the intercomparisons a new wind dataset not available in the archive*
 - a. If you haven't already done so, make sure you have already added this new wind dataset to the **read_data.py** module and to the main collocation program **MAINSRIPT_for__Collocation.bash**. See [Section 4.3.1.3](#) Item #1 for details.
 - b. In **MAIN.plotting_code.py** (and in **MAIN.plotting_code.AMVtypes.py**):
 - i. Add the following line to the *Import python modules* section at the top of the script:


```
from read_data import read_${NewDataset}
```

 where **read_\${NewDataset}** is defined in [Section 4.3.1.3](#) Item #1.
 - ii. In the if-else block below the line **# Define and load DRIVER dataset**, add another **elif** condition corresponding to the new dataset just above

the **else** condition. Follow the syntax of the other **elif** conditions.

Notes:

1. Check that the output variables within the new **elif** condition (left side of the equal sign =) match those listed in the last line of the new function **read_\${NewDataset}** (on **return** line) in **read_data.py**.
 2. Check that the input variables within the new **elif** condition (within the parentheses after the function name on the right side of the equal sign) match those listed in parentheses in the first line of the new function **read_\${NewDataset}** in **read_data.py** (to the right of the function name).
 - iii. Repeat *Step ii* but this time in the if-else block within the section **# Define dependent datasets** below the line **# read dependent dataset**. Follow the syntax of the other **elif** conditions in this if-else block.
2. *Change cross section longitudes for VERTvsLAT_LonRange plots*
 - a. The arrays **vertcrossLONSmin** and **vertcrossLONSmax** govern the cross section longitude boundary minimums and maximums, respectively. They can be found in **MAIN.plotting_code.py** (and in **MAIN.plotting_code.AMVtypes.py**) under the section **# More plotting prep**. Change their values if desired.
Note: **vertcrossLONSmin** and **vertcrossLONSmax** must be the same size, and the value of each min must be less than the corresponding value of each max.
 3. *Change grid resolutions for 2D maps*
 - a. In **MAIN.plotting_code.py** (and in **MAIN.plotting_code.AMVtypes.py**) under the section **# Set global parameters**:
 - i. Modify **latlonstride** (i.e., the step for the horizontal resolution arrays **flats** (latitudes) and **flocs** (longitudes)). This can be found below the line **# set horizontal resolution**.
 - ii. Modify **plevstride** (i.e., the step for the vertical *pressure* resolution array **fpres**). Also modify **hlevstride** (i.e., the step for the vertical *height* resolution array **fhgts**). These can be found below the line **# set vertical resolution**.
 4. *Change horizontal extent of 3D maps*
 - a. In **tools_plotting.py** within the function **map_3d_profile** under **# Define horizontal extent**, modify the values of the array **extent**. **extent** must always have 4 values, which together represent the lower left and upper right corners of the horizontal space to be plotted.
Example: If **extent = [-180, 180, -90, 90]**, the lower left corner is (-180,-90) and the upper right corner is (180, 90), where corner values are defined as (longitude, latitude).

5. *Change 3D map perspective*

- a. In **tools_plotting.py** within the function **map_3d_profile** under **# 3D viewpoint**:
 - i. Modify **azim_pt**
 1. **azim_pt** moves the plot's center point left or right. When **extent = [-180, 180, -90, 90]**, **azim_pt=230** indicates the lower-left corner (-180,-90) is in the center of the plot. If **azim_pt<230**, the center of the plot moves to the right of this point; if **azim_pt>230**, the center of the plot moves to the left of this point.
 - ii. Modify **elev_pt**
 1. **elev_pt** moves the plot's center point up or down.

6. *Change plotted vertical levels for 2D maps*

- a. In **MAIN.plotting_code.py** (and in **MAIN.plotting_code.AMVtypes.py**) under the section **# Set global parameters** below the line **# Vertical levels for plotting**:
 - i. Modify **levsP**. Make sure there is at least one pressure level (units = hPa). This variable is type float.
 - ii. Modify **levsZ**. Make sure there is at least one height level (units = km). This variable is type float.

Appendices

A1. Acronyms and Abbreviations

3D	3-dimensional
2D	2-dimensional
AMV	Atmospheric motion vector
ASCII	American Standard Code for Information Interchange
BUFR, bufr	Binary Universal Form for the Representation of meteorological data
CIMSS	UW-Madison Cooperative Institute for Meteorological Satellite Studies
CISESS	UMD/ESSIC Cooperative Institute for Satellite Earth System Studies
CODA	ESA Common Data Access
ECMWF	European Centre for Medium-Range Weather Forecasts
EE, EE-format	Earth Explorer format
EMC	Environmental Modeling Center
ESA	European Space Agency
ESSIC	Earth System Science Interdisciplinary Center
FM-A	Aeolus laser nominal flight model A
FM-B	Aeolus laser nominal flight model B (backup)
FTP	File Transfer Protocol
GEO	Geostationary satellite
HLOS	Horizontal Line-of-Sight
L2B	Level-2B
LEO	Low Earth-orbiting satellite
LOS	Line-of-Sight
NCEP	National Centers for Environmental Protection
NESDIS	NOAA National Environmental Satellite, Data, and Information Service
NetCDF, netcdf	Network Common Data Form
NOAA	National Oceanic and Atmospheric Administration
NWP	Numerical weather prediction
NWS	NOAA National Weather Service
OSTI	NOAA/NWS Office of Science and Technology Integration
prepBUFR	Preprocessed BUFR
QC	Quality control
QI	Quality indicator
SSEC	Space Science and Engineering Center
STAR	NOAA/NESDIS Center for Satellite Applications and Research
UMD	University of Maryland
URL	Uniform Resource Locator (online website)
UW-Madison	University of Wisconsin-Madison
WMO	World Meteorological Organization
xml	Extensible Markup Language

A2. Tips, Tricks, and Other Useful Information

1. To view a NetCDF file `${FILENAME}`, open a terminal window and enter either of the following commands on the command line:

```
ncl_filedump ${FILENAME} ... to view metadata information
ncview ${FILENAME}      ... to view variables graphically
```

Note that the file extension (either `.nc` or `.nc4`, whichever corresponds to the file in question) must be included to execute the above commands.

2. Use the following `squeue` command to check the status of your SLURM jobs.

```
squeue -u ${username} -o "%.18i %.9P %.30j %.8u %.8T %.10M %.9I %.6D %R"
```

where `${username}` is your username on the HPC account you are using.

Tip: Create an alias for the `squeue` command in your `.rc` file in your home directory so you don't have to type it out every time! For example, in a `.bashrc` file, add the following:

```
alias myq='squeue -u YourUserName -o "%.18i %.9P %.30j %.8u %.8T %.10M %.9I %.6D %R"'
```

3. At present, QC is available for Aeolus and AMVs, but has not yet been added for Aircraft, Sonde, or Loon. To add QC for the latter datasets, or a different dataset altogether, follow the steps in [Section 4.3.1.3](#) Item #2.
4. If you see the following warning in a `LOG*` file, do not be alarmed.

WARNING: missing_value not used since it cannot be safely cast to variable data type

It is a standard Python warning letting the user know that the `missing_value` attribute in a NetCDF file is outside the “valid range” and hence it ignores the “valid range.” This has no effect on the application, data, or output.

5. To use the collocation application on a machine other than S4, make sure you modify the bash variables `archive_parent`, `account`, `qos`, `partition`, and `timelimit`, as well as all `{loaded modules}` according to your machine's protocols, in both the main collocation and main plotting scripts:

```
MAINSRIPT_for__Collocation.bash
MAINSRIPT_for__Plotting.bash
```

Publications List

List of publications that make use of SAWC:

Hoffman, R. N., Lukens, K. E., Ide, K., and Garrett, K., 2021: A collocation study of atmospheric motion vectors (AMVs) compared to Aeolus wind profiles with a feature track correction (FTC) observation operator. *Q. J. R. Meteorol. Soc.*, 148(742):321–337, Jan. (Part A), <https://doi.org/10.1002/qj.4207>.

Lukens, K. E., Garrett, K., Ide, K., Santek, D., Hoover, B., Hoffman, R. N., and Liu, H., 2023: System for Analysis of Wind Collocations (SAWC): A Novel Archive for the Intercomparison of Winds from Multiple Observing Platforms Provides for Aeolus Evaluation. In preparation.

Lukens, K. E., Ide, K., Garrett, K., Liu, H., Santek, D., Hoover, B., and Hoffman, R. N., 2022: Exploiting Aeolus level-2b winds to better characterize atmospheric motion vector bias and uncertainty, *Atmos. Meas. Tech.*, 15, 2719–2743, <https://doi.org/10.5194/amt-15-2719-2022>.

Santek, D., Dworak, R., Wanzong, S., Rink, T., Lukens, K., Reiner, S., and García-Pereda, J., 2022: NWC SAF Winds Intercomparison Study Report: 2021. International Winds Working Group, available at http://cimss.ssec.wisc.edu/iwgg/Docs/CIMSS_AMV_Comparison_2021_Report_02Nov2022.pdf.

References

- Borde, R., Carranza, M., Hautecoeur, O. and Barbieux, K., 2019: Winds of change for future operational AMV at EUMETSAT. *Remote Sensing*, 11(18), 2111. <https://doi.org/10.3390/rs11182111>.
- Borde, R., Hautecoeur, O. and Carranza, M., 2016: EUMETSAT global AVHRR wind product. *Journal of Atmospheric and Oceanic Technology*, 33, 429–438. <https://doi.org/10.1175/JTECH-D-15-0155.1>.
- Bormann, N., Saarinen, S., Kelly, G., and Thepaut, J.-N., 2003: The Spatial Structure of Observation Errors in Atmospheric Motion Vectors from Geostationary Satellite Data, *Mon. Weather Rev.*, 131, 706–718, [https://doi.org/10.1175/1520-0493\(2003\)131<0706:TSSOOE>2.0.CO;2](https://doi.org/10.1175/1520-0493(2003)131<0706:TSSOOE>2.0.CO;2).
- Boukabara, S. A., Zhu, T., Tolman, H. L., Lord, S., Goodman, S., Atlas, R., et al., 2016: S4: An O2R/R2O infrastructure for optimizing satellite data utilization in NOAA numerical modeling systems. a step toward bridging the gap between research and operations. *Bulletin of the American Meteorological Society*, 97, 2359–2378. <https://doi.org/10.1175/bams-d-14-00188.1>
- Cotton, J., Doherty, A., Lean, K., Forsythe, M., and Cress, A., 2020: NWP SAF AMV monitoring: the 9th Analysis Report (AR9), Tech. rep., NWP SAF, Version 1.0, REF: NWPSAFMO-TR-039, <https://nwp-saf.eumetsat.int/site/monitoring/winds-quality-evaluation/amv/amv-analysis-reports/> (last access: 9 May 2021).
- Dabas, A., Denneulin, M., Flamant, P., Loth, C., Garnier, A. and Dolfi-Bouteyre, A., 2008: Correcting winds measured with a rayleigh doppler lidar from pressure and temperature effects. *Tellus A: Dynamic Meteorology and Oceanography*, 60, 206–215. <https://doi.org/10.1111/j.1600-0870.2007.00284.x>.
- de Kloe, J., Stoffelen, A., Tan, D., Andersson, E., Rennie, M., Dabas, A., Poli, P., and Huber, D., 2020: Aeolus Data Innovation Science Cluster DISC ADM-Aeolus Level-2B/2C Processor Input/Output Data Definitions Interface Control Document. Tech. rep., KMNI, Aeolus, DISC, REF: AED-SDECMWF-L2B-037, <https://earth.esa.int/eogateway/documents/20142/37627/Aeolus-L2B-2C-Input-Output-D-D-ICD.pdf> (last access: 17 April 2021).
- Durre, I., Vose, R. S., and Wuertz, D. B., 2006: Overview of the Integrated Global Radiosonde Archive. *J. Climate*, 19, 53-68, doi: <https://doi.org/10.1175/JCLI3594.1>.
- European Space Agency (ESA), 2020: Aeolus L2B Earth Explorer dataset, ESA [data set], <https://aeolus-ds.eo.esa.int/oads/access/> (last access: 25 February 2021).
- European Space Agency (ESA), 2021: Summary of Quality of Aeolus Data Products from 2nd Reprocessing Campaign covering June 2019 to October 2020. <https://earth.esa.int/eogateway/documents/20142/0/Aeolus-Summary-Reprocessing-2-DI-SC.pdf> (last access: 13 April 2023).

- Genkova, I., Borde, R., Schmetz, J., Daniels, J., Velden, C., Holmlund, K: Global atmospheric motion vector intercomparison study. In Proceedings of the 9th International Winds Workshop, Annapolis, MD, USA, 14–18 April 2008. https://www.researchgate.net/profile/Johannes-Schmetz/publication/237834561_GLOBAL_ATMOSPHERIC_MOTION_VECTOR_INTERCOMPARISON_STUDY/links/0c96052825b3d80e7f000000/GLOBAL-ATMOSPHERIC-MOTION-VECTOR-INTERCOMPARISON-STUDY.pdf (last access: 13 April 2023).
- Genkova, I.; Borde, R.; Schmetz, J.; Velden, C.; Holmlund, K.; Bormann, N.; Bauer, P. Global atmospheric motion vector intercomparison study. In Proceedings of the 10th International Winds Workshop, Tokyo, Japan, 22–26 February 2010. <http://cimss.ssec.wisc.edu/iwvg/iww10/talks/genkova2.pdf> (last access: 13 April 2023).
- Hoffman, R. N., Lukens, K. E., Ide, K., and Garrett, K., 2021: A collocation study of atmospheric motion vectors (AMVs) compared to Aeolus wind profiles with a feature track correction (FTC) observation operator. *Q. J. R. Meteorol. Soc.*, 148(742):321–337, Jan. (Part A), <https://doi.org/10.1002/qj.4207>.
- Lukens, K. E., Ide, K., Garrett, K., Liu, H., Santek, D., Hoover, B., and Hoffman, R. N., 2022: Exploiting Aeolus level-2b winds to better characterize atmospheric motion vector bias and uncertainty, *Atmos. Meas. Tech.*, 15, 2719–2743, <https://doi.org/10.5194/amt-15-2719-2022>.
- Martin, A., Weissmann, M., Reitebuch, O., Rennie, M., Geiß, A., and Cress, A., 2021: Validation of Aeolus winds using radiosonde observations and numerical weather prediction model equivalents, *Atmos. Meas. Tech.*, 14, 2167–2183, <https://doi.org/10.5194/amt-14-2167-2021>.
- Moninger, W. R., Mamrosh, R. D., and Pauley, P. M., 2003: Automated Meteorological Reports from Commercial Aircraft. *BAMS*, 84(2), 203–216, <https://doi.org/10.1175/BAMS-84-2-203>.
- National Academies of Sciences, Engineering, and Medicine: Thriving on Our Changing Planet, 2018: A Decadal Strategy for Earth Observation from Space. National Academies Press, Washington, DC, USA, <https://doi.org/10.17226/24938>.
- Rani, S.I., Jangid, B.P., Kumar, S., Bushair, M.T., Sharma, P., George, J.P., et al., 2022: Assessing the quality of novel Aeolus winds for NWP applications at NCMRWF. *Quarterly Journal of the Royal Meteorological Society*, 148(744), 1344–1367. Available from: <https://doi.org/10.1002/qj.4264>.
- Rennie, M. and Isaksen, L., 2020: The NWP impact of Aeolus Level-2B winds at ECMWF. Tech. rep., Aeolus Data Innovation Science Cluster DISC, Ref: AED-TN-ECMWF-NWP-025, <https://doi.org/10.21957/alift7mhr>.
- Rhodes, B., and Candido, S., 2021: Loon Stratospheric Sensor Data [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.3755988>.
- Santek, D., García-Pereda, J., Velden, C., Genkova, I., Wanzong, S., Stettner, D., and Mindock, M., 2014: A new atmospheric motion vector intercomparison study, in: 12th International Winds Workshop, 16–20 June 2014, Copenhagen, Denmark, CIMSS, Technical Report,

- <http://www.nwcsaf.org/aemetRest/downloadAttachment/225> (last access: 18 December 2020).
- Santek, D., Dworak, R., Nebuda, S., Wanzong, S., Borde, R., Genkova, I., García-Pereda, J., Negri, R. G., Carranza, M., Nonaka, K., Shimoji, K., Oh, S. M., Lee, B.-I., Chung, S.-R., Daniels, J., and Bresky, W., 2019: 2018 Atmospheric Motion Vector (AMV) Intercomparison Study, *Remote Sens.*, 11, 2240, <https://doi.org/10.3390/rs11192240>.
- Santek, D., Dworak, R., Wanzong, S., Rink, T., Lukens, K., Reiner, S., and García-Pereda, J., 2022: NWC SAF Winds Intercomparison Study Report: 2021. International Winds Working Group, available at http://cimss.ssec.wisc.edu/iwvg/Docs/CIMSS_AMV_Comparison_2021_Report_02Nov2022.pdf.
- Santek, D., Hoover, B., Zhang, H., and Moeller, C., 2021: Evaluation of Aeolus Winds by Comparing to AIRS 3D Winds, Rawinsondes, and Reanalysis Grids, in: 15th International Winds Workshop, 12–16 April 2021, Virtual, CIMSS, <https://www.ssec.wisc.edu/meetings/iwvg/2021-meeting/presentations/oral-santek/>.
- Šavli, M., Pourret, V., Payan, C. and Mahfouf, J.-F., 2021: Sensitivity of Aeolus HLOS winds to temperature and pressure specification in the L2B processor. *Atmospheric Measurement Techniques*, 14, 4721–4736. <https://doi.org/10.5194/amt-14-4721-2021>.
- Velden, C. S. and Bedka, K. M., 2009: Identifying the Uncertainty in Determining Satellite-Derived Atmospheric Motion Vector Height Attribution, *J. Meteorol. Clim.*, 48, 450–463, <https://doi.org/10.1175/2008JAMC1957.1>.
- Velden, C. S. and Holmlund, K., 1998: Report from the working group on verification and quality indices (WG II), in: 4th International Winds Workshop, 20–23 October 1998, Saanenmöser, Switzerland, EUMETSAT, https://cimss.ssec.wisc.edu/iwvg/iww4/p19-20_WGReport3.pdf.
- Velden, C., Daniels, J., Stettner, D., Santek, D., Key, J., Dunion, J., Holmlund, K., Dengel, G., Bresky, W., and Menzel, P., 2005: Recent innovations in deriving tropospheric winds from meteorological satellites, *B. Am. Meteorol. Soc.*, 86, 205–223, <https://doi.org/10.1175/BAMS-86-2-205>.
- Velden, C. S., Hayden, C. M., Nieman, S. J., Menzel, W. P., Wanzong, S., and Goerss, J. S., 1997: Upper-tropospheric winds derived from geostationary satellite water vapor observations, *B. Am. Meteorol. Soc.*, 78, 173–195, [https://doi.org/10.1175/1520-0477\(1997\)078<0173:UTWDFG>2.0.CO;2](https://doi.org/10.1175/1520-0477(1997)078<0173:UTWDFG>2.0.CO;2).
- Weiler, F., Kanitz, T., Wernham, D., Rennie, M., Huber, D., Schillinger, M., Saint-Pe, O., Bell, R., Parrinello, T., and Reitebuch, O., 2021a: Characterization of dark current signal measurements of the ACCDs used on-board the Aeolus satellite. *Atmos. Meas. Tech.*, 14, 5153–5177, <https://doi.org/10.5194/amt-14-5153-2021>.
- Weiler, F., Rennie, M., Kanitz, T., Isaken, L., Checa, E., de Kloe, J., Okunde, N., and Reitebuch, O., 2021b: Correction of wind bias for the lidar on board Aeolus using telescope temperatures. *Atmos. Meas. Tech.*, 14, 7167–7185, <https://doi.org/10.5194/amt-14-7167-2021>.