

20 February 2017

TO: Ranadu documentation
FROM: Al Cooper
SUBJECT: Manual for the “Shiny App” in Ranadu
VERSION: Ranadu 2.5-17-01-16

1 General features of the Ranadu ‘Shiny App’

The Ranadu Shiny App is designed for data analysis using data files from aircraft. It provides a comprehensive set of analysis routines, most building on the capabilities provided by the ‘Ranadu’ package for R (documented separately in the Manual for Ranadu, available at [this link](#)). The ‘Shiny app’ provides these capabilities:

1. plotted time series of measurements
2. flight tracks and time-height displays
3. statistical summaries of measurements
4. histograms of measurements
5. scatterplots of measurement pairs
6. measurements of one variable binned in increments of a matching second measurement, as might be used to construct soundings
7. hydrometeor and aerosol size distributions
8. displays of 2D images of particles [not yet working]
9. a skew-T diagram, optionally with a hodograph and/or display of adiabatically ascending trajectories and calculation of CAPE
10. variance spectra obtained by several methods
11. some utilities for saving subsets of measurements, for opening subsets in other analysis programs, for addition of new calculated variables, and for finding least-squares fit coefficients.
12. generation of new variables from specified formulas in global.R, function specialVar(), and inclusion of those in all the above displays as well as in other analysis programs.
13. via shiny-server, web-based display that can be made available in a web server from a computer having access to the data files.

The plot definitions can be saved and re-used. Most plots can be constructed in multiple panels, with subset plots in a configurable set of rows and columns, and there can be many definitions of these multi-panel plots. The plots can also be configured as desired for line colors, line widths, symbols, etc. In addition, there is a time slider that can be used to select the time displayed or to step through the flight in specified intervals like one-plot-per-hour. Restrictions can also be specified to exclude, for example, turns or slow flight that might affect wind measurements. The time and other restrictions apply simultaneously to all the analysis products, so a particular interval can be selected and examined in the various plots that are provided. Similarly, a given plot can be repeated for different flights and projects.

When the program starts, it checks for data files from flight projects and includes a list of those projects in a drop-down menu so a given plot can be examined in different projects. The research, test, and ferry flights for each project can be selected. There is also an 'F' option that selects files with names like PROJECTrf01F.nc so that special output files can be viewed and compared to the standard ones.

Finally, there are capabilities to produce copies of the various plots in PNG format, and it is possible to save the subset of data actually used in small R-data archives for documentation and re-use.

This manual documents ways of using these capabilities, but in most cases it is sufficiently obvious that it will be evident how to proceed without this documentation. One aspect that needs attention, though, is the expected structure of the directories in the data archive, so those requirements will need to be met before this will function. The other following sections of this document should serve as complete but often unnecessary descriptions of usage. This should be particularly true once some sections are read, because an effort at parallel construction of the subfunctions should help extend usage conventions from one plot to another.

For a video demonstration of the capabilities in this program, see <ftp://ftp.eol.ucar.edu/pub/temp/users/cooperw/RanaduShinyDemo.mkv>.

2 The general structure of the program

This information should not be needed to use the program, but is included here for documentation.

A 'shiny app', provided by RStudio, enables interactive applications where the output can be customized by the user without resort to programming changes. These applications are suited to web presentations so can be made widely available to users who only need a web browser to run the application. The Ranadu application is designed to provide the basic functions needed in analysis of measurements from aircraft, and is tailored to such measurements. In particular, it uses data files archived in standard ways as netCDF files. It is designed in particular to use the archive files produced by NCAR from research flights supported by the Research Aviation Facility.

The program is contained in three R files: *global.R*, containing code executed at the start of a run and also containing functions that are used by other files; *ui.R*, containing the definitions of the components of the user interface, and *server.R*, containing the functions that construct the displays and determine how user-provided changes are handled. Each is discussed briefly below:

global.R This script defines needed libraries (like Ranadu) and searches data locations for available projects. It defines some global variables that are referenced by the other scripts, and it otherwise initializes the global work space. It also defines many functions that are called from *server.R*. There is also a “specialVar()” function where new variables can be defined. A common use for this has been to check the performance of alternate algorithms for calculation of some variables in the data files.

ui.R This script defines the components of the user interface. That definition includes navigation panels, tab panels, drop-down menus, checkbox and event items, and it defines the areas where plots and other displayed information will appear. Each item is associated with an ID, and the next section then responds to changes in the items by recognizing when the item associated with the ID changes.

server.R The core of the calculations is in this section. The plots as defined by items set in *ui.R* are constructed here and placed in appropriate locations on the user interface. If, for example, the time range is changed, a new plot is generated automatically that shows the new time range. In some cases, routines are called in the global environment; for example, reading of 2D-image data is accomplished by a call to `'readRecord()'` in the global environment, but this is only for convenience.

There is a general use `'data()'` function that provides measurements from the netCDF-format file for all functions except the particle images, which are not contained in those files. The server maintains a current `data.frame` containing all the variables needed by any routine in the application, and if a new one not previously used is requested via the user interface then the netCDF file is re-read to add the new variable to the `data.frame`.

There are two types of restrictions that can be applied to the data. First, there is a slider that can specify the time range to use, and there are associated numeric-entry lines that alternately can be used to specify the time interval. The interval is applied, via the `'data()'` function, to all analysis components. Second, there is a capability, described in more detail later, for restricting the data to periods meeting a set of criteria that the user can specify. For example, if you want the sounding to include only measurements in-flight, you can restrict the measurements to those where the airspeed is above some limit like 110 m s^{-1} to avoid on-the-ground and close-to-takeoff-or-landing measurements.

3 Obtaining and Starting the Program

Ranadu is archived on GitHub, at this URL: <https://github.com/WilliamCooper/Ranadu.git>. The Shiny App is now included in the master branch. The easiest way to use this is via RStudio. (See the Ranadu Manual for instructions on obtaining and installing that package.) This structure is recommended:

1. Establish a directory in your home directory named RStudio.

2. Start RStudio (program name 'rstudio'; e.g., type 'rstudio' at a shell prompt).
3. Create a new directory via RStudio, using the 'File -> New Project' sequence as described in the Ranadu Manual.. This will download the Ranadu package as well as the shiny app.
4. Build and install the Ranadu package as described in the Ranadu Manual.
5. Be sure that the 'shiny' package for R is available, or else download it from CRAN.

In addition, a particular structure is assumed for the locations of data files and names. For NCAR netCDF files, the assumed structure is as follows:

1. The root directory for data files can be /Data, /home/Data, ~/Data, or /scr/raf_data. The programs will search for a matching location.
2. Under that root, there should be directories for each project, named with the same name that is used for the data files. For example, there might be directories named DEEPWAVE, MPEX, CONTRAST, CSET, etc., for those projects.
3. In each directory, the data files should be netCDF files with names like CSETrf05.nc for the CSET research flight 5. Test and ferry flights can also be included; e.g., CSETtf01.nc, DEEPWAVEff01.nc. These names can be links if the original files have different names.

The application also handles files from FAAM and UWYO, in preliminary fashion. In those cases, the files need to be loaded in a similar directory structure located below the root data directory and named FAAM or UWYO; for example, /Data/FAAM/RICO containing files named RICOrfxx.nc for some two-digit xx.

The app can be run in several ways:

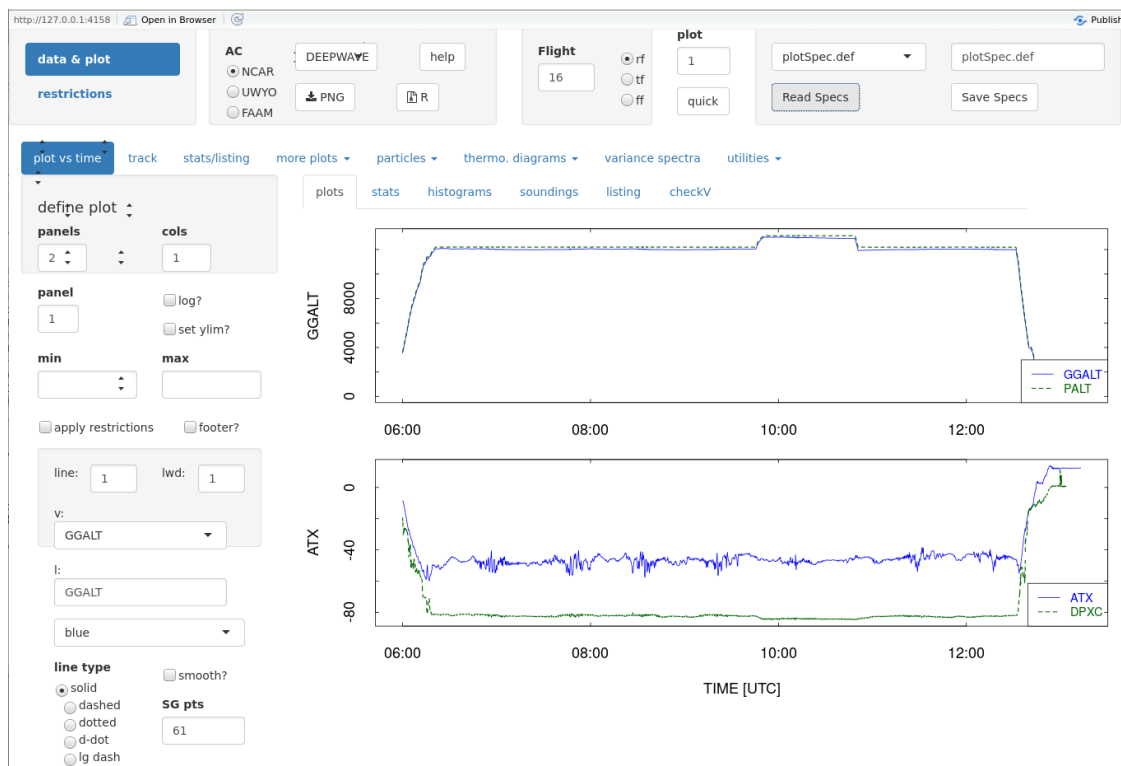
- In the RStudio console window (normally bottom-left panel), type 'shiny::runApp()' without the quote marks. This will run the trio of programs that constitute the app. See the next item for a method of selecting the desired type of display.
- Select one of {*global.R*, *ui.R*, *server.R*} from the 'File -> Open File' sequence or the Files menu, so that one of those is displayed in the editing window (normally, top left pane), then click the 'Run App' button at the top right corner of the editing pane to run the program. Note that, at the right of that button, there is a small drop-down arrow that can be used to select the run mode, with these choices: a separate RStudio window, the 'Viewer' pane of the RStudio display, or an 'External' browser display. Usually, the 'External' display, maximized to fill your viewing screen, will give the best experience.
- The shiny app can also be installed on a shiny server for general use by anyone able to access that server. That installation is not covered in this manual, but information available from the RStudio shiny web site (at this link) and, for example, on Amazon Web Services, describes this process.

This program continues under active development, and to assist in debugging many messages will appear in the RStudio console as it runs. Many of these can be suppressed by setting the variable “Trace” to FALSE near the start of the “global.R” script. There is also a line at the start of that script that clears the global environment, to avoid carryover effects from a previous run. Comment that line if this is undesirable.

4 First Glance at the User Interface

4.1 The display at start-up

Running the application should produce a window like the following:



There are two main parts to this display:

1. At the top, there are controls for selecting the project and flight and for saving and restoring configuration information. The application loads information from a specifications file called 'plotSpec.def', but additional files with names like 'plotSpec.CSET1' can be used to save configurations for re-use, and those configurations can be restored via the 'read specs' button and drop-down menu above it. There are two panels in this part of the display: In addition to 'Data and Plot' (shown), there is a second tab named 'restrictions' that provides controls for

setting the time interval and for defining a set of restrictions that cause only data that meets those restrictions to be used in all functions. The functions in this part of the display, both for 'Data and Plot' and 'restrictions', are discussed in more detail in section 4.2.

2. The lower part of the display includes a set of tabs for various types of displays: 'plot vs time', 'track', etc. The content of the display changes depending on which tab is selected, but is always split into two parts:

- (a) the left sidebar, containing controls for how the display is generated; and
- (b) the right part, where the plot is shown.

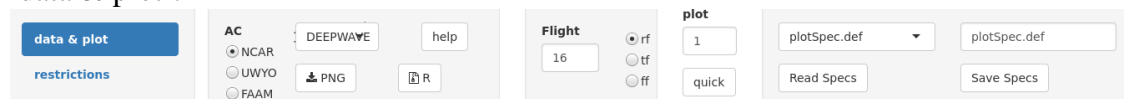
In the example shown, two stacked plots are displayed. The format of the plot (up to 6 panels in up to three columns), the variables included in each, and the line type, color, and width for each variable can be defined in the sidebar panel. Other tabs have similar controls and displays. In four cases, the tab is itself a drop-down menu for other functions; for example, the 'more plots' tab provides a menu allowing selection of 'histogram', 'scatterplot', or 'bin-average plots', each with its own sidebar control panel and display panel. Each of these is discussed in detail in this manual, but their use should mostly be obvious without these detailed instructions. The details are provided for reference in case something doesn't work as expected.

In the case of the 'plot vs time' tab, the display window also has a secondary set of tabs: 'plots', 'stats', 'histograms', 'soundings', and 'listing'. These apply only to the plot variables that are selected; more general plots of this type can be produced by the main tabs. However, they provide a quick way of looking at the selected plot variables.

4.2 The two tabs in the top panel

The following provide more details regarding the top part of the display:

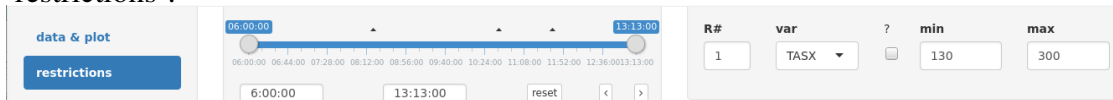
1. 'data & plot':



- (a) supports selection of the source institution providing the aircraft data, the project, flight number, and type of flight ({rf, tf, ff} for research, test, and ferry flight). To use files from FAAM or UWYO instead of NCAR, it is best to start by loading the appropriate specifications file; e.g., plotSpecFAAM.def, because the standard sets of variables used by those flight facilities differ from those used by NCAR. When the program is started, it searches the standard data directory for projects and adds those found to the project menu. Clicking on the menu displays a list from which the project can be selected. The 'Flight' entry can be incremented by using the up-down arrows, by placing the cursor in the entry window and using the up and down keyboard keys, or by selecting and typing the entry. These controls permit use of a specified plot in various projects or various flights of a particular project without restarting or changing the plot specifications.

- (b) provides a 'help' button that creates a display of the file 'RanaduShinyManual.pdf' (the file you are viewing), a detailed description of this application with instructions. This is separate from the file 'RanaduManual.pdf', which is a manual for the R package of routines called 'Ranadu'. The latter can be seen via a link near the start of the RanaduShinyManual.
- (c) includes buttons for saving the displayed plot to a PNG-format file, and also for saving the data.frame in use as an R data file. The latter is useful because the data.frame is much smaller than the full netCDF file and, if the netCDF file is not found, will be used instead. It contains only the subset of variables in current use, however, so when using the R data file it may not be possible to add new variables to plots. The R data file contains information needed to produce size distributions for particles or hydrometeors as well as the other plots, but not for display of 2D images. One common use for this data.frame is to customize the plot (e.g., by changing the location of the legend, labels for the axes, etc.) to a format desired for publication.
- (d) allows specification of a plot number, so that multiple plots can be defined. These plot definitions can then be saved for future use.
- (e) via a 'quick' button, supports pop-up display of the plotted time series for a variable selected from a menu. More flexible plots can be generated using controls discussed later in this manual, but this is often a useful way to see a particular variable while working with other functions in the display.
- (f) allows a user to save the specifications for future use. When the program first starts, it loads the specifications in 'plotSpec.def'. The left pair of entries allow selection of a different saved specifications file (provided it has a name starting with 'plotSpec') by selecting the desired file and reading it. The right pair of entries allow saving of the specifications, either to the default or with a different name (e.g., plotSpec.CSETrf01) that can be entered in the top right window..

2. 'restrictions':



- (a) allows specification of the time interval to be used.¹ This can be done in several ways:
 - i. via adjustment of the start and end sliders by dragging them.
 - ii. by entry of a time into the windows for the start and end times. It is not necessary (but acceptable) to type the ':' in the time entry; an entry like 143000 will be accepted and converted to 14:30:00.
 - iii. by "brushing" the section of a plot to display by holding down the left mouse button while moving over the time interval in the plot. The selected time interval is displayed when the cursor is released.

¹The convention used is that the time interval includes the start time but excludes the end time.

- iv. by using the small left- and right-pointing arrows below the time bar. If an interval has already been selected, these arrows will display the next or previous similar-duration interval; for example, if a 15 min interval is selected like 14:00:00 to 14:15:00, clicking the forward button will reset the time interval to 14:15:00 to 14:30:00.

There is also a “reset” button for returning the slider to cover the full flight. The “F4” key also triggers this reset and can be used even when the “restrictions” tab is not displayed.

- (b) supports the definition of restrictions to be applied to the data. For example, in the case shown, TASX is required to be between 130 and 300. This can be useful for eliminating slow-flight conditions where flaps may be deployed, for example. Additional restrictions can be added by incrementing the 'R#' number, specifying a new variable from the drop-down menu of all available variables, and entering the minimum and maximum values. The '?' checkbox determines if the restriction will be included, and is provided so that the effects of restrictions can be tested without removing them from the specifications. Most control sidebars have a checkbox for 'apply restrictions' that must be checked for these restrictions to be applied to the data.

5 Description of the tabs in the main part of the display

5.1 'plot vs time'

The sidebar control panel

panels	Up to six panels can be specified. Each panel is an independent plot and can be defined via controls listed below. The panels will be arranged in a grid with the number of columns specified by the next entry. All panel definitions are retained if this entry is changed, but only the top panels are displayed; for example, if panels=2, but six panels are defined, only the first two are displayed.
cols	The panels as specified above are displayed in 'cols' columns.
panel	The characteristics that follow apply to the panel number specified here. As a special case, if 'panels' (above) is 1, the panel selected here is displayed in a full-size plot even if “panel” is larger than 1.
log	Select this check-box to get a logarithmic scale for the ordinate. Errors will result if there are negative or zero values in the variables being plotted. This applies only to the selected panel and can be set independently for each panel.

- set ylim? Select this check-box to force the plot limits on the ordinate axis to the values listed in the following entry windows labeled 'min' and 'max'. Provide values before selecting the check-box to avoid divide-by-zero errors. In the base of a logarithmic scale, enter the values (e.g., 0.001) not the log10 value (-3). This applies only to the panel selected so limits can be set individually for all panels.
- min. max The ordinate-axis limits to use when the preceding check-box is selected.
- apply restrictions Select this check-box to apply the restrictions as defined in the top 'restrictions' tab.
- footer Do you want a footer added to the plot indicating project/flight/date/times etc? This is sometimes useful when saving plots via the "PNG" button, to identify the individual plots.
- line: The rest of the entries apply to an individual plotted trace in the selected panel. Select the line number you want to change, then use the remaining controls to define that line. If a line number is entered that exceeds the number of lines currently defined, a new line is defined and added to the plot. Initially it will duplicate the last defined line, but you can then change the variable, line width, color, etc.
- lwd: The width to use for the plotted line. "1" is often a good starting point. Fractional values can be used.
- v: [variable name] The next drop-down menu includes selections for all the variables in the netCDF file as well as any that have been defined in "specialVar()" or via the "create new variable" function discussed below. There is a special entry at the top of the list named 'omit' which will remove the currently selected variable from the current panel. When this occurs, the display changes to 'select' to indicate that this line is no longer defined, and you should decrement the line number to eliminate the variable from the plot.
- l: [y-axis label] The default label is the variable name of the first variable in the panel, but that can be changed using the entry in this text-input line. There is some limited capability to use R plotmath expressions. These must be preceded by 'expr:', and this is followed by an expression of this form: 'paste("temperature [",degree,"C"])' without the single-quote marks but including the double-quote marks.
- [color menu] The next drop-down menu presents a set of color choices. The line will be plotted in the selected color. For now, this list is a fairly short selection of colors. Others will likely be added in later versions.
- line type The variable can be plotted using the indicated choices, where d-dot refers to a dash-dot pattern and lg-dash to large dashes.

- smooth? A checkbox that, when selected, applies a Savitzgy-Golay-polynomial filter to the selected line. This does not affect the values used for statistics or histograms, described below, but can remove hash that makes comparison of almost-overlapping lines difficult. See the next control to determine how much smoothing will be applied. This can be set individually for each line.
- SG pts The number of points used in the Savitzgy-Golay filter. A third-order filter with the specified number of points is used. A rough approximation is that the effective filter period is about 1/3 the number of points.

The tabbed display panel

Normally the 'plots' tab will be used. Other tabs are provided to show characteristics of the selected plot variables, but these are available in better format in more general plots defined by the tabs in the main panel. All the tabs in the display panel are controlled by the entries in the sidebar panel. Explore these if you are interested, but no further discussion is provided here except to explain that the 'soundings' tab shows bar-and-whisker plots of the plot variables partitioned into increments in altitude (defined by GGALT).

The tab "checkV" has a special use. If the mouse is left-clicked on a time-series plot, the values of the plot variables in one-minute averages centered on the clicked time will be available in the "checkV" window. This window also allows entry of a desired time for the average with better resolution than can be provided by the mouse-click. Only those variables used in the current plot are included.

5.2 'track'

The sidebar defines the flight-track in terms of self-explanatory entries, with a few exceptions. The flight track will be plotted with a background map showing geographic information, either land-mass boundaries (for non-US plots) or state boundaries (for US flights). This background is fairly coarse; better and aesthetically appealing plots can be constructed instead by the KML files that NCAR/EOL/RAF and other groups provide for research flights. This plot capability is included here for quick context when viewing the plots. The plots are generated by the Ranadu script plotTrack.R, so that can be consulted for more details. The displayed track observes the restrictions defined in the "restrictions" panel, e.g., by plotting only the limited time range selected there.

There are two tabs, one for a normal plan-view flight track, described below, and the other for a simple time-height plot showing the geometric (not pressure) flight altitude as a function of time.

The controls are as follows:

- apply restrictions As for all tabs, checking this check-box applies the restrictions in the top 'restrictions' tab to the data, and will leave gaps in the plotted track where data are excluded

by those restrictions. Time restrictions are always applied, independent of this control check-box.

footer? As for 'plot vs time'

drifting? If this check-box is checked, the plot is constructed by dead-reckoning from the initial point, and so will give a flight track that drifts with the wind. For that reason, no background geography is shown for this selection.

center long,

center lat. These entries support fixing the plot relative to given center coordinates (specified as fractional longitude and latitude in degrees, with west longitude or south latitude as negative numbers). If these are not specified, the routine centers the track appropriately based on the data ranges, so these are not usually needed.

size As for center coordinates, the routine makes a reasonable choice for the size so this usually is not needed except for special uses. The size must be specified in degrees and applies to both latitude and longitude.

time between labels The track will be plotted with time labels at the interval specified.

scale for wind flags For an entry other than 0, wind flags along the direction of the wind and with magnitude proportional to the wind speed will be plotted along the track, using the numerical entry as a scale controlling the length of the wind flags. These are flags, not wind barbs, which would be in the opposite direction. A value of 5 specifies that the length of the wind flag will be 5% of the plot size in latitude for a wind speed of 10 m/s, with the longitudinal component adjusted appropriately to avoid distortion of the wind direction.

5.3 'stats/listing'

This tab provides two kinds of text tables, either statistical summaries of variables over the selected time period (including mean, standard deviation, minimum and maximum) or a listing (optionally averaged over a specified interval) of selected measurements. The choice between these two types of display is made by the radio buttons at the bottom specifying 'type of output'. The 'apply restrictions' check-box functions as for 'plot-vs-time'.

The time interval selected in the top 'restrictions' tab controls the averaging time for the statistical summaries or the length of the listing of values. The other key control is provided by the 'select variables' button. Clicking on this produces a display of all the available variables in the netCDF file, with the currently selected variables highlighted in yellow. (These are determined initially by the 'specPlot.def' file and once changed can be saved back to that or another specPlot file.) There is an 'Actions' menu at the top left corner of this 'Available Variables' window. Possible actions include 'return selections and hide window' to accept the choices that are displayed, 'select ALL

variables' and 'clear ALL variables' to select everything or clear all selections to start over, and 'Quit without saving' to dismiss the window without making any changes. 'select ALL' and then returning with that selection will show everything in the 'stats' display, in a multi-page display, and using this for the 'listing' display will produce a very wide output window with a scroll bar that makes it possible to view everything. The 'Search' window can be used with a time like 14:30:00 to search for that time in the displayed listing.

5.4 'more plots -> histogram'

This tab produces histograms. Many of the controls in the sidebar panel function as for the 'plot vs time' control panel, so that earlier section should be consulted for 'apply restrictions', 'footer?', 'panels', 'cols', and 'panel'. Furthermore, the line controls at the bottom function as described for 'plot vs time'. There are three differences vs 'plot vs time' controls:

plot density? When this check-box is selected, the ordinate of the histogram is the density of events; i.e., the fraction of events per unit change in the abscissa. The sum of the density in each bin multiplied by the width of the bin is 1.

include cdf Selecting this check-box adds a cumulative-distribution line to the histogram and adds a right axis ranging from 0 to 1 to provide coordinates for the line. By default, the cumulative distribution function (giving the fraction of measurements smaller than the indicated abscissa value) is plotted as a thin line of the same color and type as the histogram line.

set xmin,

xmin, These entries define the range of the abscissa axis and are applied if the check-box is selected.

xmax

There is no way (yet) to define the number of bins in the histogram or the size of the bins, but this will be added eventually.

5.5 'more plots -> scatterplot'

The sidebar panel controlling production of scatterplots follows the same pattern as for the preceding tabs; c.f. 'plot vs time' above. The new or different aspects are these:

log x, log y Either or both of the scales can be set to be logarithmic.

fixed If this check-box is selected, the limits provided in the 'xmin', 'xmax', 'ymin', 'ymax' windows at the bottom will be used to define the plot ranges. If this is not selected, a reasonable choice will be made based on the ranges of the values being plotted.

- pair Instead of a definition for each line, a definition is provided for each pair of measurements that are to be plotted. When multiple pairs are defined for a particular panel, they all share the same abscissa variable but can have different ordinate variables for each pair. Therefore two entries are provided for 'x' (abscissa) and 'y' (ordinate) for each pair.
- [color] This drop-down menu supports selection of the plot color for each pair.
- symbol size The size to be used for the symbol plotted for each pair of measurements. Fractional values are accepted as long as they are larger than or equal to 1.
- symbol The entry should be an integer corresponding to the R symbol list. Possible values are 1—25 for various symbols (20 is a good small dot), 48–57 for numbers 0–9, 65–90 for capital letters A–Z, 97–122 for lower-case letters a–z.

At the top of each plot, a title is included that gives the regression fit with correlation coefficient and residual standard error.

5.6 'more plots -> bin-average plots'

The controls in the sidebar panel are almost identical to those in the 'scatterplot' sidebar, except for the as-yet-inactive 'horizontal' check-box, so they won't be discussed again. Refer to Section 5.5 and preceding sections for explanations of these controls.

Instead of a scatterplot, each pair of variables is binned into equal intervals in the variable labeled 'x' (often GGALT), and then box-and-whisker plots are constructed for each bin. The box extends from the first to the third quartile of values, with the median denoted by a bar. The whiskers then extend to the data point at most 1.5 times as far from the box as the interquartile range. Symbols connected by a line denote the means in each bin. The terminology is somewhat confusing because the variable labeled 'x' is actually plotted as the ordinate and the one labeled 'y' as the abscissa; this is because it is so often useful to use these plots to display profiles with height. Titles at the top of the plots list the results of regression fits to the pairs of measurements.

Future additions will include controls over the number of bins and perhaps implementation of the 'horizontal' function to rotate the plot so that bins are defined in the abscissa of the plot.

5.7 'particles -> size distributions'

At initialization or when switching projects, the data variables are searched for those providing particle or hydrometeor size information and those are displayed as check-boxes in the sidebar control panel. The displayed plot shows the average over the full time interval. For viewing particle size distributions, it is often useful to set a time like 15 min or 1 h and then use the arrow buttons in the time control bar to step through the file. The only control provided at present is

for the types of plot axes, either linear or logarithmic. All concentrations are shown in units of $\text{cm}^{-3}\mu\text{m}^{-1}$ for consistency; for the 2DC plot (actually showing the distribution variable C1DC), for better scaled units of $\text{L}^{-1}\text{mm}^{-1}$, multiply values by 10^6 .

Because the data formats used by FAAM and UWYO are different, this portion of the program will not work yet for data from those flight facilities.

5.8 'particles -> hydrometeor images'

The application provides a simple way of viewing images from the 2D probes. The data for the 2D images are stored in separate files, constructed by stripping the 2D image records from the original tapes and incorporating them into a new data file normally suffixed with .2d. These are often saved in a subdirectory 'PMS2D' of the production directory. Once selected, it may be useful to save the configuration as applied to a particular project and flight because then the file name is saved and it is not necessary to search for it again. This function requires a special format for data and is not available for FAAM or UWYO data.

Two types of display are implemented, 'page' and 'record'. In 'record' mode, 2D images are displayed one record at a time; in 'page' mode, a set of up to sixteen records is displayed, two on each line, but at reduced image size. The other modes ('second' and 'image') are placeholders for not-yet-implemented displays.

The numeric input window labeled 'min seconds between displayed records' supports skipping records so at most one record per second is displayed. This is often useful in cases where the concentration is very high or when spurious images (e.g., streakers or stuck bits) are produced at a high rate. The specified minimum time applies to records in the page view also, so a page view will span at least 16 times the value set here. The time restrictions provided by the time slider and controls are also obeyed, so it is possible to skip to a future point in the file. Skipping backward in this way is not supported (except circuitously by for example changing to and from another project to reset the display). However, stepping forward or backward by single records is possible via the arrow controls at the bottom, except that the minimum time will still apply so it needs to be set to 0 for backward skips.

5.9 'thermo. diagrams -> skewT diagram'

This display produces a skew-T diagram with measurements of temperature and dewpoint (by convention, variables ATX and DPXC) plotted on the diagram as functions of pressure (PSXC). The 'show hodograph' superimposes a hodograph showing the wind direction and speed as a function of pressure; by convention WDC and WSC are used. The 'CAPE/LCL/adiabats' checkbox adds a box in which CAPE (convective available potential energy), CIN (convective inhibition), LCL (lifted condensation level), LFC (level of free convection), and some other properties of the sounding are listed. This checkbox also superimposes the LCL and temperature lines for pseudoadiabatic and for reversible adiabatic ascent on the plot and plots the LCL.

The skew-T diagram used for the background is not standard and needs some documentation and explanation. Many standard diagrams are based on the Rossby definition of equivalent potential temperature, the Goff-Gratch formulation for vapor pressure, and a molecular weight of air dating to before recent increases in CO_2 modified that average. These have been changed in this diagram through use of the Murphy-Koop formulation of the vapor pressure, the Davies-Jones evaluation of equivalent potential temperature, and other adjustments to current values for the gas constant and specific heats. This is documented in 'SkewT.pdf' in the Ranadu "inst" directory, and also at this URL. The diagram is not re-generated but is loaded from a previously generated file. If it is desirable to change this file, see the preceding link, which also links to the code used to generate the skew-T background.

For those not familiar with the skew-T log-p diagram, here is a short explanation:

- The logarithm of the pressure is the ordinate, so isobars are level lines.
- Lines of constant temperature (isotherms) are the blue lines or thin-dashed-blue skewed upward to the right. Their values are labeled along the bottom, right, and top axes.
- Lines of constant mixing ratio are the dash-dot green lines sloping upward more steeply than the isotherms. This property would be constant for a parcel moving vertically without mixing.
- Orange curved lines sloping upward to the left are lines of constant dry-air potential temperature. This quantity would be constant for a parcel of dry air moving vertically without mixing.
- Red lines are lines of constant entropy for a parcel starting at 1000 hPa and moving upward, but for which condensed water is assumed to form at 100% relative humidity and then to be removed from the parcel. They therefore represent pseudo-adiabatic equivalent potential temperature, with values matching the value of the asymptotically equal value of potential temperature for the low-temperature limit. These are sometimes also called equivalent wet-bulb temperature and labeled by the value at 1000 hPa. The extra heat arising from freezing of condensate or from condensation at the frost point rather than the dew point is not incorporated into this diagram. Ascending parcels would follow these lines provided all condensate occurs at 100% relative humidity and that condensate then instantly falls from the parcel as precipitation.

The calculation of convective available potential energy uses the Ranadu routine 'CAPE.R', and help for that function and the RanaduManual.pdf provides further documentation of the procedure used.

5.10 'thermo. diagrams -> mixing and stability plots'

Three types of plot can be generated:

1. “*Paluch*” or “*ThetaQ/Qtot*” plots: The Paluch diagram is a plot of total water mixing ratio (QTOT) vs. wet-equivalent potential temperature (THETAQ). Measurements are averaged in increments of pressure, and points along the plotted trajectory are labeled by the pressure. These two quantities are conserved in the absence of mixing, so properties of mixing processes that may occur in clouds can be deduced by superimposing in-cloud measurements on those from a nearby sounding. The standard reference for this diagram is Paluch, I. 1979. J. Atmos. Sci.. 36. 2467–2478.
2. “*Betts saturation-point*” diagrams: Similar studies of mixing are also possible using this diagram, which uses total water mixing ratio and moist entropy as the two conserved quantities and uses skewed coordinates for these quantities in order to give a diagram where, for large parts of the diagram, pressure and temperature resemble a skew-T diagram and can be represented as standard lines on the diagram. A reference for the diagram is Betts, A. K., 1982: Saturation Point Analysis of Moist Convective Overturning. J. Atmos. Sci., 39, 1484–1505.
3. “*stability profiles*”: Sounding displays useful for assessing stability. These plots include profiles of virtual potential temperature, pseudo-adiabatic equivalent potential temperature, wind components, the Richardson number and the Brunt-Vaisala number. The latter two are defined as

$$\text{Ri} = \frac{\frac{g}{\Theta_v} \frac{d\Theta_v}{dz}}{\left(\frac{du}{dz}\right)^2 + \left(\frac{dv}{dz}\right)^2}$$

$$N = \sqrt{\frac{g}{\Theta_v} \frac{d\Theta_v}{dz}}$$

Values of $\text{Ri} < 0.25$ lead to instability and values >1 to stability, with hysteresis effects between these limits. Positive values of N are stable, and parcels displaced from their equilibrium value will tend to oscillate with an angular frequency of N .

For all three plots, start and end times for a sounding should be supplied, and a ‘bins’ choice should be made to specify the number of bins in which sounding properties are averaged. A choice of 18 gives labels for the Paluch diagram each 50 hPa, 30 gives 30-hPa increments, and 90 gives 10-hPa increments; other choices sometimes lead to awkward labels for this diagram, but a full range of choices can be used for the other two types of plot.

For plot types 1 and 2, a measurement of liquid water content is needed. A selection menu, with default ‘PLWCC’, is provided for selection of this variable. Also, for these plot types, a subset of the flight (not necessarily overlapping with the sounding) can be specified to select times for which to plot in-cloud points. These points are only plotted if the CDP concentration exceeds 5 cm^{-3} . The times selected for the sounding and in-cloud periods should be within the limits set by the time slider in the top ‘restrictions’ tab.

The saturation-point background is generated separately, by a script called BettsWork.R, and is saved in a file named 'satptDiagram.Rdata'. This file is loaded at program start-up, and soundings and in-cloud points are then added to be basic diagram as in the case of the skew-T sounding.

The additional entries in the sidebar apply only to the stability profiles. The values for Ri and N are conventionally calculated using the virtual potential temperature, but for moist processes the pseudo-adiabatic equivalent potential temperature is a more appropriate measure, so this choice can be selected by the choice at the bottom of the sidebar. Also, there is an entry labeled 'avg.' that can be used to control smoothing of the derivatives used to calculate Ri and N; a choice of 7 applies 3rd-order Savitzky-Golay polynomials spanning 7 points to the measurements to provide some smoothing. It seems to work well to choose a relatively high number of bins and then reduce noise with a higher number for the smoothing polynomials. (The number actually used must be odd, so entries will be adjusted upward if necessary to be an odd number; hence entering 8 or 9 for example will give the same smoothing.)

5.11 'variance spectra'

Various displays for spectral analysis are available in this tab. This type of display relies on an external program ("Xanadu") for construction of the variance spectra, so it will not work unless that program is installed and available. The program is legacy software that is difficult to install, so for now it is best to do spectral analysis in some other way. If Xanadu is available, there are many options here for spectral analysis. The controls are these:

variable The drop-down menu below this label provides a list of all available variables in the netCDF file. This variable is used for the primary calculation of spectral variance.

co-variable Some of the functions (e.g., covariance or coherence) provide for analysis of two variables. The second variable is selected in the same way as the first.

[tabs] Next are three tabs for different approaches to spectral analysis, each of which will have individual sidebar control panels and display windows, as discussed in the sub-subsections that follow:

1. 'fft' for conventional analysis via Fourier transformation;
2. 'acv' for analysis by finding the autocovariance function and the Fourier-transforming to find the spectral density, which is formally equivalent to the 'fft' method except for how averaging is handled;
3. 'MEM' for application of the maximum-entropy or all-poles method of spectral analysis pioneered by Burg.

restrictions The check-box to apply restrictions can have undesirable effects because it can introduce discontinuities or gaps into the time series, so it is best to use this only with caution.

'fft'

The 'fft' tab controls spectral analysis by Fourier transformation. The controls are these:

segment length The time series will be broken into segments of this length and the spectral density calculated for each. The results are then combined to produce an estimate that has lower variance than would a single estimate. This reduces the variance in the result, but it also reduces the maximum length or minimum frequency for which a result will be obtained. Values must be equal to powers of 2 (because a fast fourier transform is used for the calculations), and if numbers are entered that do not satisfy this requirement they will be changed to the largest number smaller than the entry that does satisfy the requirement.

[window] The next entry is a drop-down menu that allows choice of a window to be applied to the measurements when performing the above averaging. 'Parzen' is usually a good choice, but other possibilities support exploration of the effect of this choice.

log avg intervals To control the variance in the result, the spectral estimates can be averaged together over intervals in frequency. This is done in logarithmic intervals where the entry in this window specifies the number to use. Using fewer intervals increases the smoothness of the result but reduces the resolution. Smoothing can be eliminated completely by using a large number (e.g., equal to the segment length). The resulting plot will show the spectral estimate before smoothing as a red line and that after smoothing as a blue line.

error bars This control is not yet implemented.

[type] The last drop-down menu controls the type of plot to be generated. These choices are supported:

1. 'fp(f)': The estimate of the spectral density ($p(f)$) is multiplied by the frequency (f) in this display. This is usually the best first choice. The weighting reduces the range of the ordinate and makes conformity with expected inertial-subrange behavior, for example, easier to perceive. This display also has the advantage that the plotted spectral density is equal to both $fp(f)$ and $\lambda p(\lambda)$ where λ is the wavelength. The plot has a wavelength scale at the top, constructed using the average airspeed over the selected interval. The green lines on this plot are lines with a $-5/3$ slope ($-2/3$ after frequency weighting), and when the variable is an air-motion variable like TASX or WIC then the lines correspond to expected behavior in an isotropic inertial subrange, where one line is labeled 10^{-4} for an eddy dissipation rate of $10^{-4} \text{m}^2 \text{s}^{-3}$ and other lines are separated by one order of magnitude.
2. 'p(f)': This provides the corresponding unweighted spectrum.

3. 'both fp(f)': With this choice, the weighted spectral densities for both variables are shown together in one two-pane display.
4. 'cospec. / quad': This selection shows the cospectrum and quadrature for the variable and co-variable. Averaging as specified by 'log avg intervals' is supported.
5. 'coherence / phase' generates plots of the coherence (with range 0 to 1) and phase of the two signals. The sign convention is that a positive phase indicates that the co-variable is ahead of the variable.
6. 'data': Finally, this entry is provided for quick inspection of the time series that are being used. Note that the series as displayed here are not changed from the original measurements, but when used each has a trend and mean removed from the data. The corresponding 'data' entry under the 'acv' tab provides an easy way to see the effect of removing the mean and trend, because that control window includes a check-box to enable or disable that adjustment of the measurements. In the 'fft' section, this is not optional because it is well known that not making such an adjustment can introduce significant distortions to the results. Because calculations in R are normally performed to high precision, removing the mean is not as important as in lower-precision calculations, but it is still useful to avoid loss of precision when large numbers with small variations are used.

'acv'

The 'acv' tab supports an equivalent calculation by starting with the calculation of the autocovariance of the specified variable. The autocovariance is calculated for a maximum lag of 3600 s unless that data segment selected is shorter than that. The estimate of the spectral variance is then found by Fourier transformation. These controls are provided:

remove trend? Select this to remove the trend from the data before calculating the autocovariance or the spectral density. It is usually best to leave this checked.

smoothing time To help control variance, before Fourier transforming, the autocorrelation function can be multiplied by an exponentially decaying function with this time constant (in seconds). Shorter times produce more variance, esp. at small frequency, but limit the smallest frequency where the spectrum can be estimated.

log avg intervals As for the fft spectra, the spectra determined from the autocovariance function can be averaged to reduce variance, and the averaged spectrum will then be shown as a red line superimposed on the blue unaveraged spectrum. An outstanding bug makes it impossible to specify a smoothing time above 900 s for a 1-h series, so it is not possible to suppress the smoothing altogether.

[type] The final control determines the type of plot. Choices are:

1. 'fp(f)': As described in the 'fft' section above.
2. 'p(f)': As described in the 'fft' section above.
3. 'var2 fp(f)': as for (1) but for the co-variable
4. 'var2 p(f)': as for (2) but for the co-variable
5. 'autocorrelation': The autocorrelation functions for the two variables, both with and without smoothing by the exponential function, will be displayed. The dashed lines are an approximate indication of the range where values exceeding those limits can be considered to indicate a significant correlation.
6. 'crosscorrelation': The plotted values show the lagged correlation between the two variables. The value of the unlagged correlation is included in the title.
7. 'data': This plot is provided to support a quick view of the two time series being used. The effect of removing the mean and trend can be seen by using the 'remove trend?' check-box.

'MEM'

This tab won't be discussed here because it is not made available in versions that are distributed publicly. The reason is that copyrighted code from 'Numerical Recipes in C' is used, so it remains to be determined how or if this code can be incorporated into a shiny application. That code cannot be included in the github public repository, but it may still be possible to provide this shiny app via an HTML server, so this may be included in future versions. One useful capability in this analysis method is the ability to add multiple lines representing the spectral density for different variables to a single plot. This capability is not yet available for the other methods.

5.12 'utilities -> run other program'

This capability makes it possible to start other analysis programs with the data in current use. The program 'ncplot' is a particularly convenient program for quick examination and analysis of aircraft data, so it can be useful for quick checks. When run, ncplot will use a special data file constructed from the variables in use in the shiny app, so these will include special variables constructed in global.R as well as those constructed by the "add variable" function. The ncplot program can be obtained from NCAR/EOL via the software web site. Another program long used by the author of this shiny app is 'Xanadu', so this can also be started with the data in current use. Xanadu provides an easy route to other analysis programs, including python, NCL, IDL, and IDV, so those are not yet implemented as direct transfers under this tab but are available if Xanadu is installed.

To use the current set of variables in use by the shiny app, just click the 'start program' button. However, if you want other variables or want to tailor the selections, click the 'select variables' button. This will display a window with all the available variables in the netCDF file. This relies

on the tcltk package and may not function on some systems. See the discussion under 'stats/listing' for details on how to use this display.

5.13 'utilities -> fits to data'

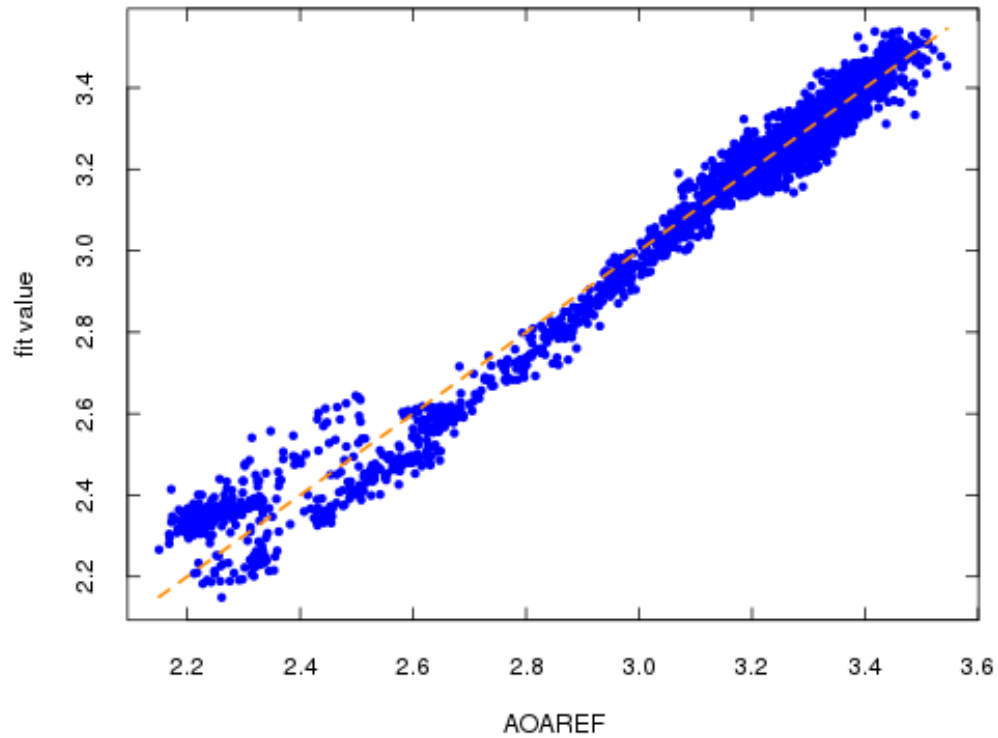
A common task is to find best-fit (in a regression sense) parameters to represent one variable in terms of others. A common example is to find a representation of angle-of-attack as a function of available measurements like the pressure measurements from the radome. The coefficients so obtained can be used for processing to find the vertical wind. That case therefore will be used as an example here.

For that example, one needs a reference variable. One often used is 'AOAREF' given by 'PITCH - GGVSPD / TASX * 180 / pi'. This will equal the angle of attack in the case where the vertical wind is zero, so it should provide a reference value that will usually average to the correct value for angle of attack. Generation of such a variable is described in the next section. For this example, assume that creation of the new variable has already been done, so that there is an available variable 'AOAREF'. (The default in the next section will create this variable.) Then, in the 'fits to data' control sidebar, use these steps:

1. In the 'fit formula' entry window, enter a formula that is a trial representation for AOAREF. An example is shown. It is necessary to enclose compound terms like ADIFR/QCF in enclosing 'I()' functions as shown in the example to prevent the fit from expanding the terms to other combinations of those variables. For a first try, enter 'I(ADIFR/QCF)+MACHX' (without the quote marks), which is the standard empirical formula usually used. (If you were to use powers like MACHX**2, enclose them in the 'I()' function. R accepts '^' in place of ** for denoting powers.
2. For the 'response variable' in the top panel, choose 'AOAREF' from the drop-down menu. This should be available if AOAREF has been created in the create-variable function. Strangely, it won't appear in the 'response variable' list until you enter a formula.
3. Click 'show linear fit'. The display will show the fit result plotted vs AOAREF and a table showing the best-fit coefficients.

You may also want to apply some restrictions. Good choices for this particular fit are TASX larger than 130 and ROLL between -5 and 5, to eliminate slow-flight segments where flaps or gear might be deployed and to eliminate turns.

An example of the result, for ORCAS flight 1, 14:00:00–15:00:00, is shown in the following figure:



response variable: AOAREF
 fit formula: $I(ADIFR/QCF) + MACHX$
 Residual standard deviation: 0.051, dof=3597
 R-squared 0.977
 Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.55373	0.10989	50.53964	0.00000
$I(ADIFR/QCF)$	23.95166	0.06229	384.51473	0.00000
MACHX	-1.20197	0.13692	-8.77854	0.00000

The 'Deming fit' button is not implemented yet, but this capability is available as a function in the "Ranadu" package..

5.14 'utilities -> create new variable'

This utility makes it possible to add new variables to the data being used. The new variable can then be used in the various other parts of this application, for example in time plots or scatterplots. The steps are these:

1. Enter a name for the new variable. The default example is AOAREF, but this can be any name starting with a letter and containing normally valid characters. If you duplicate an existing name, the previous variable will be overwritten. This might be useful if you want to recalibrate by, e.g., adding an increment everywhere, but it can also be very dangerous because there will be no indication that this has occurred.
2. Enter a formula for the new variable. Arithmetic operations are acceptable, as shown in the default example. The symbol π is a recognized R variable. Any variables in the netCDF file can be used in the formula. (For a list, for example see the 'select variables' button in the 'run other program' tab.)
3. Once the formula and variable name are entered, click the 'create new variable' button at the top.

– End of Memo –