MJO Air-Sea Interaction Diagnostics Users Guide 1 February 2018
Charlotte DeMott; demott@atmos.colostate.edu

Input Files

Required file naming convention:

- 1 file per variable, time step = 1 day (i.e., daily means)
- filenames must conform to:

caseName.*variableName*.*YYYYMMDD-YYYYMMDD*.*latitude*S-*latitude*N.day.mean.nc

caseName should ideally be short, and should uniquely identify the dataset or experiment. For example, ERAI, MERRA, SPCAM3, SPCAM3_notopo, UKMO_ENTx2, etc.

The *variableName* listed in the file name must match exactly the *variableName* in the netCDF file.

Required 2D input fields: rainfall; aribtrary units

- 10-meter U (U1000 or lowest model level U can be substituted) ; m s**-1
- 10-meter V (V1000 or lowest model level V can be substituted) ; m s**-1
- U850 ; m s**-1
- surface latent heat flux ; W m**-2
- surface sensible heat flux ; W m**-2
- surface net longwave radiation ; W m**-2
- surface net shortwave radiation ; W m**-2
- 2-meter specific humidity (or Q1000 or lowest model Q) ; kg/kg or g/kg
- 2-meter air temperature (same as above) ; K or deg C
- SST or surface skin temperature ; K or deg C
- surface pressure ; Pa or hPa
- top of model net LW ; W m**-2
- top of model net SW ; W m**-2

These fields **DO NOT** require specific variable names. Output generated by the diagnostics will use a standard set of variable names. For surface latent and sensible heat flux, positive can be either into the ocean or into the atmosphere. This will be diagnosed in the code.

Required vertically integrated (VI; from 1000 hPa to 100 or 50 hPa) MSE budget terms:

Vmse:	VI MSE, where MSE = Cp*T + L*q + g*Z ; J/kg/s-1
Vdmdt:	VI MSE tendency, where Vdmdt(i) = [Vmse(i) - Vmse(i-1)] / 86400. ; W m**-2
Vm_hadv:	VI horizontal MSE advection (positive = moistening) ; W m**-2
Vudmdx:	VI horizontal MSE advection by u-wind (positive = moistening) ; W m**-2
Vvdmdy:	VI horizontal MSE advection by v-wind (positive = moistening) ; W m**-2

Vomegadmdp: VI vertical MSE advection (positive = moistening) ; W m**-2
Vlw: VI longwave heating (or netTOA - netSFC LW; mean < 0) ; W m**-2
Vsw: VI shortwave heating (or netTOA - netSFC SW; mean > 0) ; W m**-2

These fields **DO** require the listed variable names, and must also conform to the file naming convention described above.

Case-specific definitions:

In the root directory, modify AirSea_definitions.sh to include a statement block for each case to be analyzed. Note that `FILESUFFSTR` refers to the YYYYMMDD-YYYYMMDD part of the input file name, while `YMDSTRT` and `YMDLAST` indicate the range of dates to be analyzed in the analysis. In most cases, `YMDSTRT` and `YMDLAST` will exactly match the dates listed in `FILESUFFSTR`, but adjusting `YMDSTRT` and `YMDLAST` allows the diagnostics to be performed on a subset of available dates (as shown in the example below). `FILEDIR` points to the directory where the input data files reside (omission of the trailing "/" is intentional). Other variables are described below:

```
if ( $modelname =~ "SPCAM4-ctrl" ) then
  setenv YMDSTRT     "20000101" # user-defined start date
  setenv YMDLAST     "20100630" # user-defined end date
  setenv FILESUFFSTR "19861115-20131231" # date range in file name
  setenv RAINFACTOR  8.64E07 # multiplier to convert rain to mm/day
  setenv FILEDIR     "/Model_output/SPCAM4/spcam4-ctrl/AirSea"
  setenv RAINVARNAME "PRECT"   # rainfall
  setenv UVARNAME    "U10"     # near-surface u-wind
  setenv VVARNAME    "V10"     # near-surface v-wind
  setenv U850VARNAME "U850"    # 850 hPa u-wind
  setenv LHVARNAME   "LHFLX"   # surface latent heat flux
  setenv SHVARNAME   "SHFLX"   # surface sensible heat flux
  setenv LWVARNAME   "FLNS"    # surface net longwave radiation
  setenv SWVARNAME   "FSNS"    # surface net shortwave radiation
  setenv QVARNAME    "QREFHT"  # near-surface specific humidity
  setenv TKVARNAME   "TREFHT"  # near-surface air temperature
  setenv SSTVARNAME  "TS"      # SST or surface skin temperature
  setenv SFCPVARNAME "PS"      # surface pressure (not mean SLP)
  setenv TIMEVAR     "time"    # time coordinate variable
  setenv LEVNAME     "lev"     # pressure-level coordinate variable
  setenv LATNAME     "lat"     # latitude coordinate variable
  setenv LONNAME     "lon"     # longitude coordinate variable
  setenv latSouth    -30       # southern latitude in file name
  setenv latNorth     30       # northern latitude in file name
```

Checking input data:

Note: All files referred to in this section reside in ./AirSea_Diagnostics.

NCL is picky about unit names for latitude and longitude and likes for all variables to have a defined _FillValue attribute. The file InputDataCheck.csh performs several checks on the input data.

- make_L0.0_VarTest.ncl: checks for "acceptable" latitude and longitude coordinates, checks for defined _FillValue attributes, reports the max/min of each variable, and checks for missing time steps. The program outputs a table showing which variables need revised lat/lon coordinate units (fix using NCO or CDO operators) and/or _FillValue attributes (fix using code described in the next step).
- make_L0.1_Add_FillValue.ncl: adds the _FillValue attribute to user-specified variables. Modify the appropriate section of InputDataCheck.csh to include all variables needing _FillValue (example shown below). In most cases, the NCL default _FillValue can be assigned to variables that don't already have this attribute. However, it's possible that missing values are reported as "-9999" (for example), in which case, the user will have to modify make_L0.1_Add_FillValue.ncl so that the right _FillValue is listed in the attribute.

```
foreach varName ( Vmse Vdmdt )
     setenv inName $varName
     ncl -Q ./make_L0.1_Add_FillValue.ncl
end
```

- make_L0.2_FluxSignFix.ncl: infers the sign convention used for surface latent and sensible heat fluxes by computing the long-term mean of these quantities for a location in the subtropical eastern Pacific Ocean (where the sign of the mean flux is unambiguous). The diagnostics require positive fluxes to be out of the ocean. If a change of sign is needed, the original data is copied to a file with an "-original" suffix added to the file name, and the input file is replaced with the sign-adjusted data.

These steps should be done **ONE AT A TIME** by commenting out the other steps, and rerunning InputDataCheck.csh once each make_L0.x program has been completed.

Running the diagnostics:

Note: All files referred to in this section reside in ./AirSea_Diagnostics, unless otherwise noted.

Edit the file AirSea_diagnostics.csh.

The "USER INPUT REQUIRED HERE" section.

A "USER INPUT REQUIRED HERE" section appears at the top of AirSea_diagnostics.csh. For `CASES`, list the dataset or model simulations for which air-sea diagnostics will be computed. For

`PTYPE` (plot type), put either "png" (tightly cropped) or "pdf" (plotted on 8.5x11 "paper"). For "png", see [this post]() by Walter Hannah describing changes that you may want to make to your *~/.hluresfile* to generate "large" files suitable for posters or publications. `DEBUG` may be set to "true" or "false", where "false" suppresses warning messages, but not fatal error messages. Setting `FIGCAP` to "true" adds 1-2 lines of descriptive text to the bottom of some figures, and is recommended for new users of the diagnostics, or new viewers of its output. This option can be turned off for plots that will be used in presentations, papers, etc.

"ExtraVars" can be set to "true" if additional variables (those not strictly required for the air-sea interaction diagnostics) should be included in some analysis steps. Examples include precipitable water, V850, or high-frequency wave activity. These additional variables must conform to the file naming conventions described above.

<u>The "Preconditioning steps" section: generating additional time series</u>

Using the input data, the diagnostics will:

- Compute Reynold's decomposed surface flux time series calculated using the input SST, and a 61-day running mean smoothed SST.
- Compute "background" (i.e., the 61-day running mean) and "anomaly" (departures from the 61-day running mean) time series of all input and computed variables.

These steps requires environmental variables `Make_AirSea_Vars`, `Make_Anomaly_Timeseries`, and `Make_SfcFlux_Components` be set to "true". Once these steps have successfully completed for the given `CASES`, they should be set to "false" even if the diagnostics are repeated for shorter subsets of data (say, focused on a several month field campaign).

Additional fields computed in this section are written to `diri`, defined in AirSea_definitions.sh (located in the root directory), using the same file naming convention described above, except "day.mean" may be replaced by "background.anom" as appropriate.

<u>The L1-L3 diagnostics section: Generating the diagnostics</u>:

The diagnostic framework is described in the DeMott et al. (2016). The diagnostic code creates two subdirectories: `$diri/proc`, and `$diri/plots`. Diagnostic computation (performed in "make" files) is separated from plotting procedures ("plot" files). The output of "make" files is written to `$diri/proc`, while the output of "plot" files is written to `$diri/plots`. Plot file names include the name of the "make" files used to generate their input data.

The diagnostics are arranged into three tiers, or levels, each containing a "make" and "plot" subsection. Each step can be executed or skipped by setting the appropriate environmental variable to "true" or "false", respectively.

Tips for generating diagnostics:

- When first using the diagnostics, or when introducing a new dataset, execute each section in blocks (e.g., execute only the “preconditioning” steps, then the L1 “make” steps, then the L1 “plot” steps).
- Diagnostics can be sped up by reducing the number of variables regressed onto MSE or dMSE/dt, or by reducing the number of wave number-frequency plots, Scroll down to the block of code that is called multiple times (e.g., `Make_LagRegression_Prop`) and reduce the number of variables for which the analysis is performed. Note that this may lead to errors later in the diagnostics.
- Running the diagnostics all at once (all options set to “true”, or for many cases) can take several hours, depending on the input data record, resolution, and processor speed. It may be helpful to direct output to a text file that can later be scanned for any errors that arose during processing.