WRFDA Verification and Graphics

Yali Wu, Michael J. Kavulich, Jr.

Special thanks to Syed RH Rizvi

July 22–July 24, 2019
National Center for Atmospheric Research
Boulder, CO

Overview

- WRFDA verification package
  - Verification against what?
  - Verification scores
- WRFDA graphics and plotting tools
  - Observation-space diagnosis
  - Model-space analysis

WRFDA Verification and Graphics Tools

Two executables generated for verification when compiling WRFDA

```
da_advance_time.exe
da_bias_airmass.exe
da_bias_scan.exe
da_bias_sele.exe
da_bias_verif.exe
da_rad_diags.exe
da_tune_obs_desroziers.exe
da_tune_obs_hollingsworth1.exe
da_tune_obs_hollingsworth2.exe
da_update_bc_ad.exe
da_update_bc.exe
da_verif_grid.exe
da_verif_obs.exe
da_wrfvar.exe
gen_be_addmean.exe
gen_be_cov2d3d_contrib.exe
gen_be_cov2d.exe
gen_be_cov3d2d_contrib.exe
gen_be_cov3d3d_bin3d_contrib.exe
gen_be_cov3d3d_contrib.exe
gen_be_cov3d.exe
gen_be_diags.exe
gen_be_diags_read.exe
gen_be_ensmean.exe
gen_be_ensrf.exe
gen_be_ep1.exe
gen_be_ep2.exe
gen_be_ep2_serial.exe
gen_be_etkf.exe
gen_be_hist.exe
gen_be_stage0_gsi.exe
gen_be_stage0_wrf.exe
gen_be_stage1_1dvar.exe
gen_be_stage1.exe
gen_be_stage1_gsi.exe
gen_be_stage2_1dvar.exe
gen_be_stage2a.exe
gen_be_stage2.exe
gen_be_stage2_gsi.exe
gen_be_stage3.exe
gen_be_stage4_global.exe
gen_be_stage4_regional.exe
gen_be_vertloc.exe
gen_mbe_stage2.exe
```

$WRFDA/var/da/da_verif_grid/ da_verif_grid.f90
$WRFDA/var/da/da_verif_obs/ da_verif_obs.f90

WRFDA Verification and Graphics Tools

Download the WRFDA_TOOLS from github

https://github.com/wrf-model/WRFDA_TOOLS

TOOLS/var/scripts

da_run_wrfvar.ksh
da_run_wrfvar_psot.ksh
da_run_psot.ksh
da_run_wps.ksh da_run_wrf.ksh
da_run_obsproc.ksh
da_run_real.ksh
da_run_suite_verif_obs.ksh
da_run_suite_wrapper_qc_obs.ksh
da_run_suite_wrapper_verif_obs.ksh
da_verif_obs_plot.ksh

da_verif_anal_plot.ksh
da_plot_psot.ksh
da_tune_obs_hollingsworth.ksh
da_tune_obs_desroziers.ksh

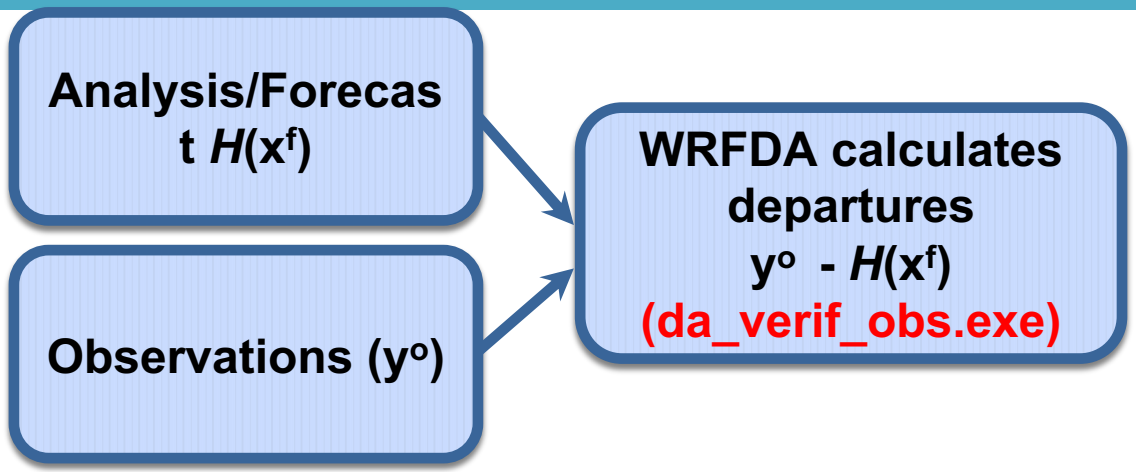
TOOLS/var/graphics/ncl

README
WRF-Var_plot.ncl
plot_cost_grad_fn.ncl
plot_obascii_loc.ncl
plot_gts_omb_oma.ncl
plot_rad_diags.ncl
plot_rad_varbc_param.ncl
verif_obs_time_series.ncl
verif_obs_time_average.ncl
verif_anal_vert_profile.ncl
verif_anal_time_series.ncl
verif_anal_time_average.ncl
verif_anal_vert_profile.ncl
convert_gts_omb_oma_to_netcdf.ncl

WRFDA verification

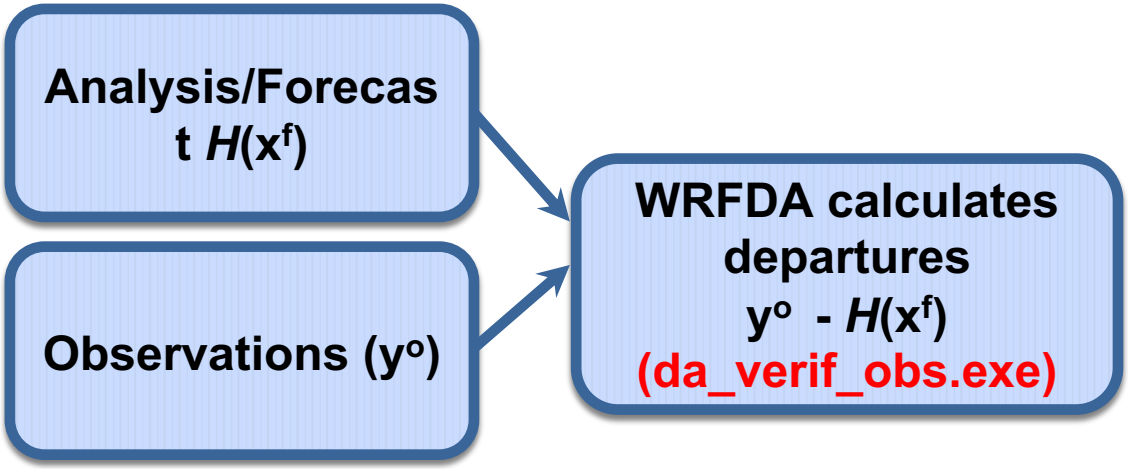
- What variables can we verify?
  - Most conventional observation variables
    - surface and upper-level U, V, T, P, Q;
    - gpsref….
- What can we verify against?
  - Observations
  - Analyses
- What scores can we use?
  - Root mean square error (RMSE)
  - Mean bias
  - Absolute mean bias
- What graphics can we have?
  - Times series, time averages, profiles, histograms

Observation-based verification



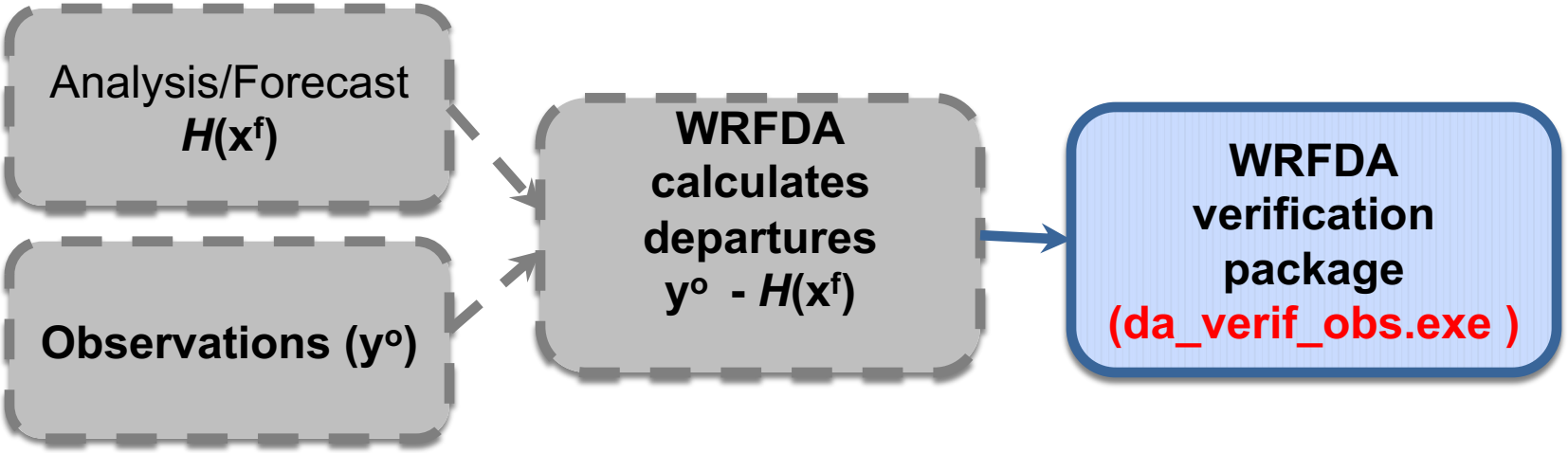
- The general idea of verification against observations
  - Interpolate model analysis/forecast from grids to irregular observation locations
  - Calculate differences between the analysis/forecast and observations at observation locations
- Why can we use WRFDA for observation-based verification?
  - The above procedure is actually similar to the calculation of innovation/departures at observation space in a general DA system
  - Difference is: use your analysis/forecast as the background, i.e., OMF

Observation-based verification



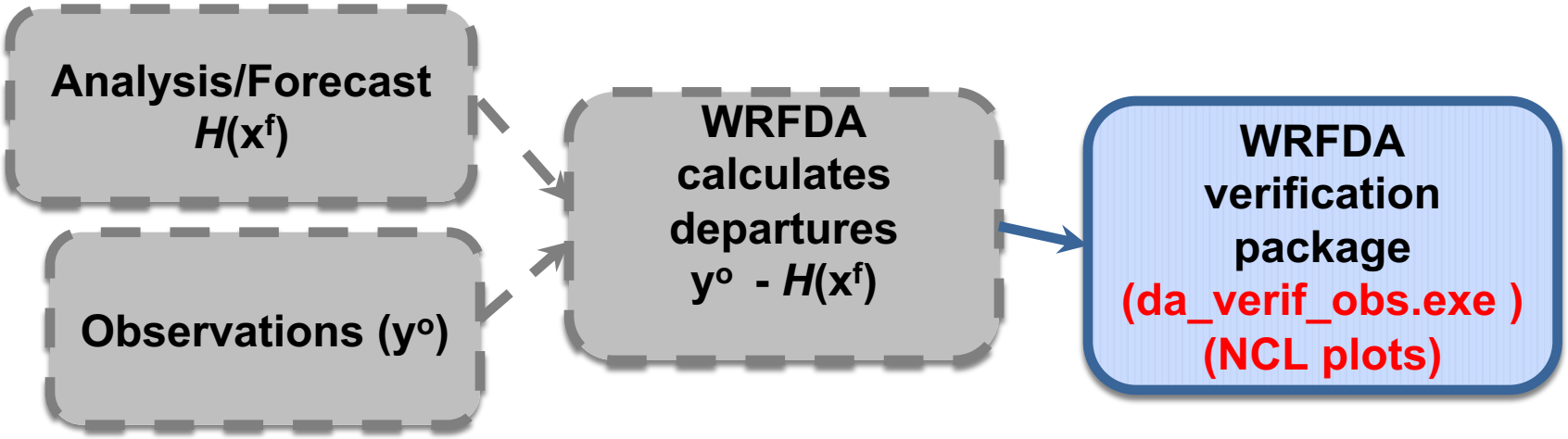
- Observation verification can be run for most types of conventional observations and for most observation variables:
  - SYNOP, METAR, SHIPS, BUOY, SOUND, SCAT, AIREP, PILOT, PROFILER, AMV, TAMDAR, GPSPW, GPSRO
- Advantages: consistent with WRFDA QC, observation operators; Has its own NCL package;
- Disadvantage: can only verify against observations that WRFDA can assimilate

Observation-based verification



- da_verif_obs.exe reads diagnostics from WRFDA
- da_verif_obs.exe calculates verification scores: RMSE, Bias, ABias
- Use namelist for selecting your interest observation type for verification: surfce_type, upper_type, gpspw/gpsref separately
- Significance test & error bars
- Source code for da_verif_obs.exe can be found in WRFDA/var/da/da_verif_obs/da_verif_obs.f90

Observation-based verification



- NCL plots read diagnostic statistics from da_verif_obs.exe
- Surfce_type, upper_type, gpspw/gpsref are plotted separately
- Select your interest scores: RMSE, Bias, Abias
- Different experiments are analyzed on the same figure

Observation-based verification

- This verification is run in three steps
  1. A set of "**filtered_obs**" files are produced by running WRFDA with the option:

     &wrfvar17

     analysis_type = "QC-OBS"

     - A "filtered_obs" file is an observation file in WRFDA-input ASCII format containing observations that have undergone basic QC checks, ensuring that there is a consistent set of observations used for verification among different experiments
  2. For each experiment, WRFDA is run in "verify" mode:

     &wrfvar17

     analysis_type = "VERIFY"

     - WRFDA reads filtered_obs as ob.ascii. Verify mode forces the variables check_max_iv=.false. and ntmax=0; this means that no minimization will be performed, but WRFDA will output OMB for this experiment in the file **gts_omb_oma_01** ⟶ **OMF, not OMB/OMA as in DA**
     - This step can be controlled by the script "da_run_suite_verif_obs.ksh

Observation-based verification

Ob.ascii:

```
TOTAL =   9061, MISS. =-888888.,
SYNOP =    757, METAR =   2416, SHIP  =     54, BUOY  =    341, BOGUS =      0, TEMP  =     86,
AMDAR =     14, AIREP =    205, TAMDAR=      0, PILOT =     85, SATEM =    106, SATOB =   2556,
GPSPW =    187, GPSZD =      0, GPSRF =      3, GPSEP =      0, SSMT1 =      0, SSMT2 =      0,
TOVS  =      0, QSCAT =   2190, PROFL =     61, AIRSR =      0, OTHER =      0,
PHIC  =  40.00, XLONC = -95.00, TRUE1 =  30.00, TRUE2 =  60.00, XIM11 =   1.00, XJM11 =   1.00,
```

Filtered_obs:

```
TOTAL =   7970, MISS. =-888888.,
SYNOP =    463, METAR =   1735, SHIP  =     39, BUOY  =    279, TEMP  =     86, TEMP_SFC  =     86,
AIREP =    216, PILOT =     85, GeAMV =   2499, PoAMV =      0, GPSPW =    187, GPSRF =      3,
PROFL =     61, QSCAT =   2126, SSMT1 =      0, SSMT2 =      0, SATEM =    105, BOGUS =      0,
AIRSR =      0, MTGIRS=      0, TAMDAR=      0, TAMDAR_SFC=      0, OTHER =      0,
PHIC  =  40.00, XLONC = -95.00, TRUE1 =  30.00, TRUE2 =  60.00, XIM11 =   1.00, XJM11 =   1.00,
```

Observation-based verification

- This verification is run in three steps
  3. Diagnostic statistics are computed with da_verif_obs.exe, and plots are made from the results
     - da_verif_obs.exe reads the statistics found in gts_omb_oma_01 created in the previous step and outputs detailed diagnostic files
     - These diagnostic files can then be read by a series of NCL scripts, which produce plots of the verification details
       - verif_obs_time_series.ncl
       - verif_obs_vert_profile.ncl
       - verif_obs_time_average.ncl
       - verif_obs_vert_profile_gpsref.ncl
     - This step can be controlled by the script “da_verif_obs_plot.ksh”

Observation verification

Variables declared in first script (da_run_suite_verif_obs.ksh):

INITIAL_DATE: Verification starting date (yyyymmddhh)
FINAL_DATE: Verification ending date (yyyymmddhh)
CYCLE_PERIOD: The period in hours between forecasts
EXP_DIR: Full path of experiment directory name
FILTERED_OBS_DIR: Directory where the ("filtered_obs" files) against which verification will be done are located
VERIFY_HOUR: 00 for analysis; 12, 24, etc. corresponding to the desired forecast hour verification
BE_DIR: Location of background error file be.dat
NL_E_WE, NL_E_SN,
NL_E_VERT, NL_DX, NL_DY: Used to set the necessary domain namelist values
e_we, e_sn, e_vert, dx, and dy, respectively
NL_ANALYSIS_TYPE: =verify; this tells the script to run WRFDA in verify mode for this step

Observation verification

Variables declared in second script (da_verif_obs_plot.ksh):

WRFVAR_DIR: WRFDA main directory (full path)

REG_DIR: Directory holding sub-directories for each experiment generated in Step 1

For example: "gts_omb_oma" file corresponding to experiment "verify_12" (directory for 12 hr forecast verification) for "2005081700" should be in $REG_DIR/verify_12/2005081700/wrfvar

RUN_DIR: Full path of the directory where plots will be generated

NUM_EXPT: Total number of experiments (Maximum 10)

EXP_NAMES: Experiment directory names as they exist in REG_DIR (blank separated)

EXP_LEGENDS: Legend strings for each experiments respectively (comma separated)

START_DATE: Starting date ("YYYYMMDDHH") for verification

END_DATE: Ending date ("YYYYMMDDHH") for verification

INTERVAL: Time interval (in hours) for incrementing date/time.

Observation verification

Variables declared in second script cont'd (da_verif_obs_plot.ksh):

NUM_OBS_TYPE: Number of observation types for verification
OBS_TYPES: Verification observation types like, "synop", "buoy", "sound" etc.
PLOT_WKS: "x11" to display plots on screen, "pdf" to save as pdf files
DESIRED_LEVELS: Pressure levels (in hPa) for plotting diagnostics
DESIRED_SCORES: Diagnostics like "RMSE", "BIAS" or "ABIAS"
EXP_LINES_COLORS: Color sequence for various experiments.
VERIFY_DATE_RANGE: Title of x-axis in the output plots

Note about wrapper scripts

- These verification scripts are designed to work best when called by a wrapper script to declare the necessary variables (not provided by WRFDA_TOOLS, but on the practice website)

```
#!/bin/ksh -aeux
# Wrapper script for running WRFDA obs verification package
# Settings for ./da_run_suite_verif_obs.ksh
export INITIAL_DATE=2013122312
export FINAL_DATE=2013122512
export WRFVAR_DIR=/kumquat/users/${USER}/DA/WRFDA
...  etc ...
# Run the first script
./da_run_suite_verif_obs.ksh

# Settings for da_verif_obs_plot.ksh
export START_DATE=2013122312
export END_DATE=2013122512
export RUN_DIR=$EXP_DIR/plots
...  etc ...
# Run the first script
./da_verif_obs_plot.ksh
```

Observation verification: Output

A successful run will produce a number of different plots in RUN_DIR, dependent on the options you choose

- Time series for surface and all the desired upper air levels
- Vertical profiles
- Time Average for surface and all the upper air levels (Histograms)

gupr_q_oma.diag	gupr_v_omb.diag	surface_u_oma.diag	upr_t_omb.diag
gupr_q_omb.diag	surface_p_oma.diag	surface_u_omb.diag	upr_u_oma.diag
gupr_t_oma.diag	surface_p_omb.diag	surface_v_oma.diag	upr_u_omb.diag
gupr_t_omb.diag	surface_q_oma.diag	surface_v_omb.diag	upr_v_oma.diag
gupr_u_oma.diag	surface_q_omb.diag	upr_q_oma.diag	upr_v_omb.diag
gupr_u_omb.diag	surface_t_oma.diag	upr_q_omb.diag	
gupr_v_oma.diag	surface_t_omb.diag	upr_t_oma.diag	

Data Format: In each row total (1 + 16 x 6 = 97) values for Upper-air
Data Format: In each row total (1 + 1 x 6 = 7) values for Surface:
Date, NUM, BIAS, Absolute BIAS, RMSE, error, & significance
levels: 1000 925 850 700 500 400 300 250 200 150 100 70 50 30 20 10
index 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

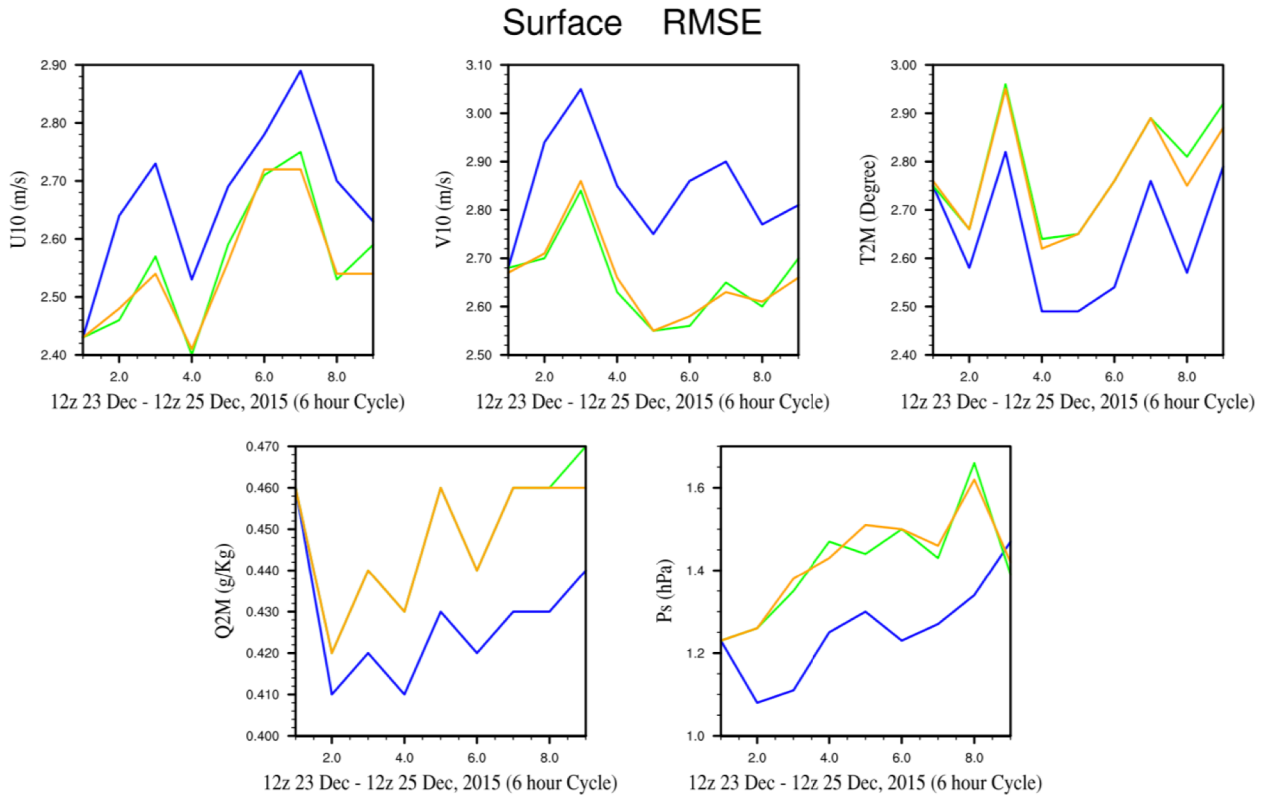
Observation verification: Output

A successful run will produce a number of different plots in RUN_DIR, dependent on the options you choose

- Time series for surface and all the desired upper air levels
- Vertical profiles
- Time Average for surface and all the upper air levels (Histograms)

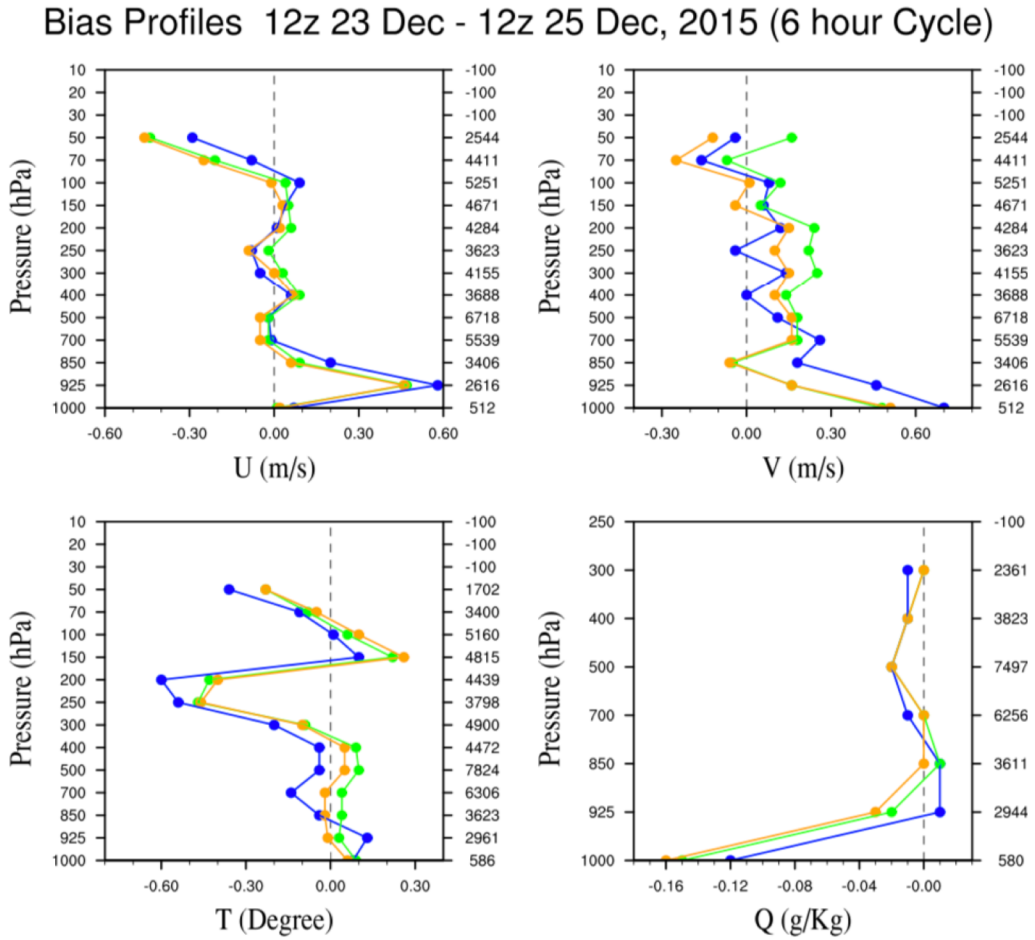
```
-rw-r--r-- 1 rizvi ncar 597691 Oct 13 12:49 Time_Series_SFC_RMSE.pdf
-rw-r--r-- 1 rizvi ncar 291856 Oct 13 12:49 Time_Series_SFC_BIAS.pdf
-rw-r--r-- 1 rizvi ncar 319570 Oct 13 12:49 Time_Series_SFC_ABIAS.pdf
-rw-r--r-- 1 rizvi ncar 1571714 Oct 13 12:49 Time_Series_UPA_RMSE.pdf
-rw-r--r-- 1 rizvi ncar 753440 Oct 13 12:49 Time_Series_UPA_BIAS.pdf
-rw-r--r-- 1 rizvi ncar 769452 Oct 13 12:49 Time_Series_UPA_ABIAS.pdf
-rw-r--r-- 1 rizvi ncar 463151 Oct 13 12:49 Profile_RMSE.pdf
-rw-r--r-- 1 rizvi ncar 467553 Oct 13 12:49 Profile_BIAS.pdf
-rw-r--r-- 1 rizvi ncar 12769280 Oct 13 14:54 Profile_ABIAS.pdf
-rw-r--r-- 1 rizvi ncar 129469 Oct 13 12:49 Time_Average_SFC_RMSE.pdf
-rw-r--r-- 1 rizvi ncar 136679 Oct 13 12:49 Time_Average_SFC_BIAS.pdf
-rw-r--r-- 1 rizvi ncar 142219 Oct 13 12:49 Time_Average_SFC_ABIAS.pdf
-rw-r--r-- 1 rizvi ncar 352928 Oct 13 12:49 Time_Average_UPA_RMSE.pdf
-rw-r--r-- 1 rizvi ncar 402740 Oct 13 12:49 Time_Average_UPA_BIAS.pdf
-rw-r--r-- 1 rizvi ncar 365264 Oct 13 12:49 Time_Average_UPA_ABIAS.p
```

Observation verification: Output



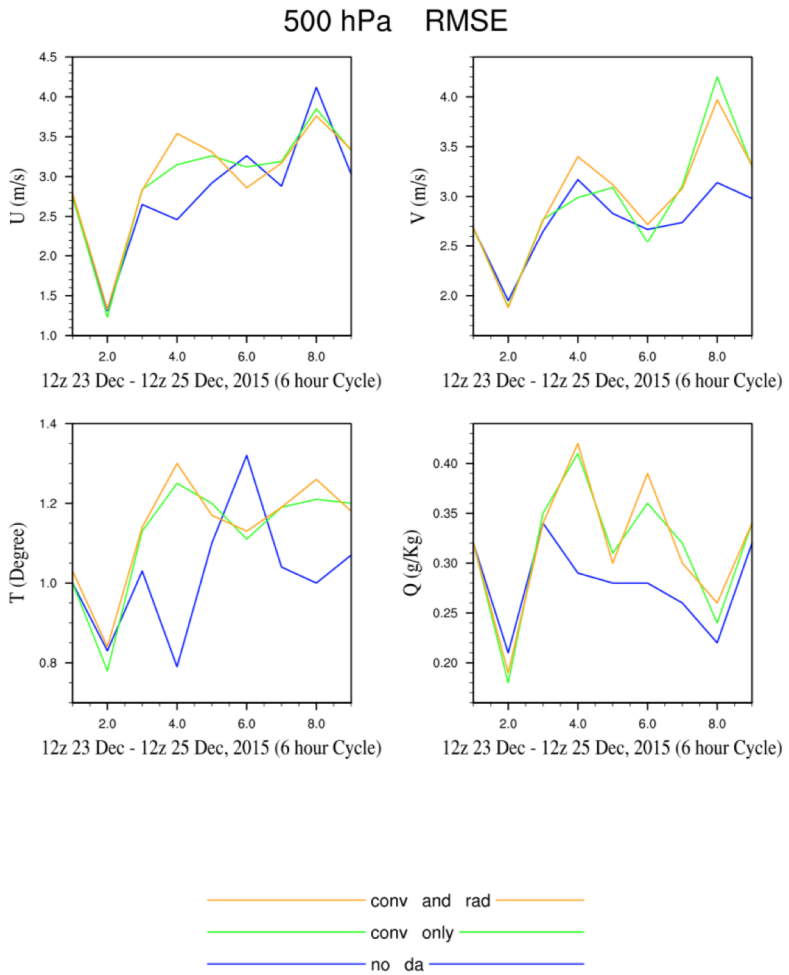
verif_obs_time_series.ncl

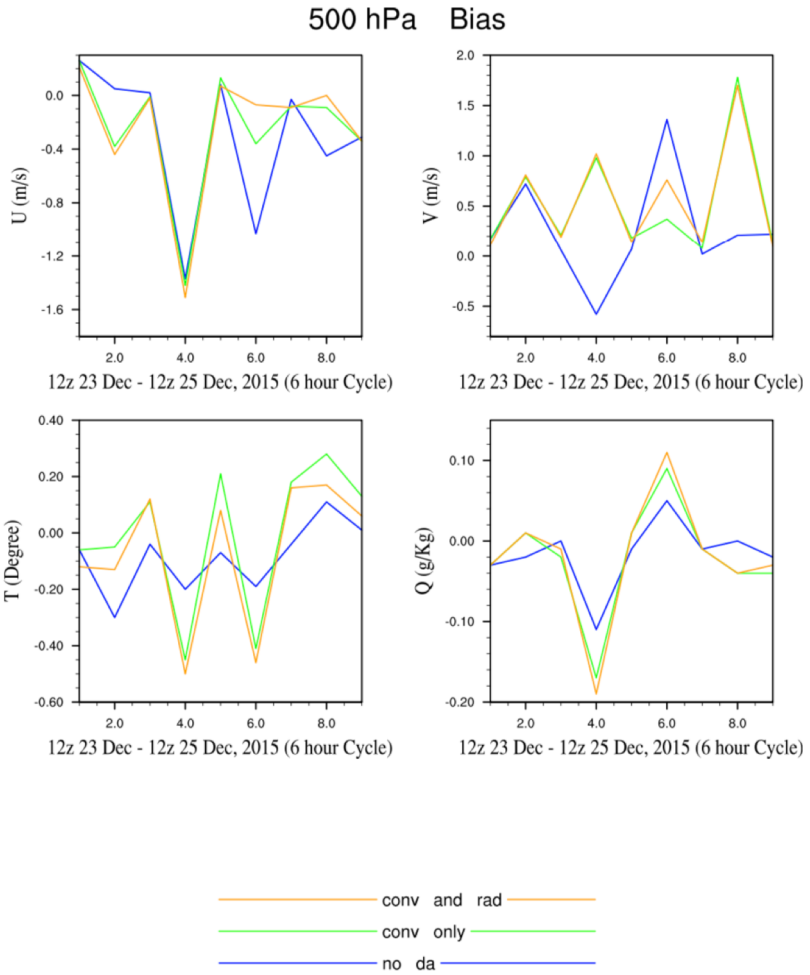
Observation verification: Output



verif_obs_vert_profile.ncl

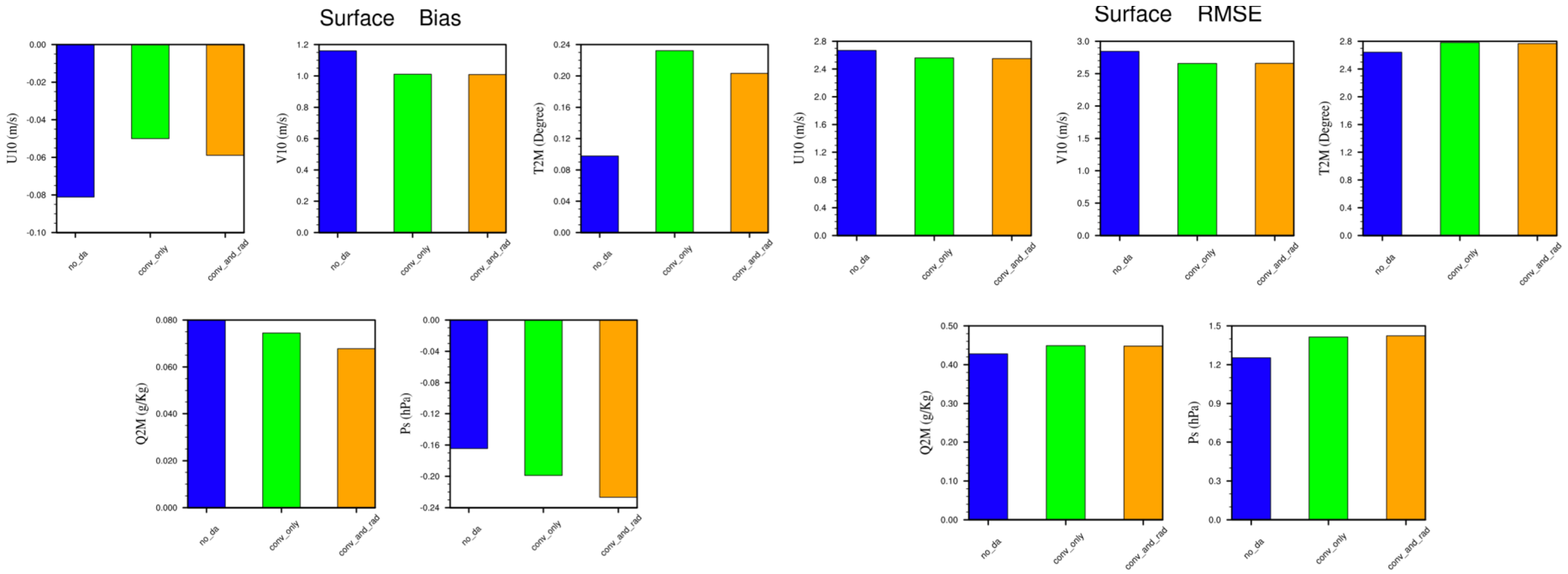
Observation verification: Output





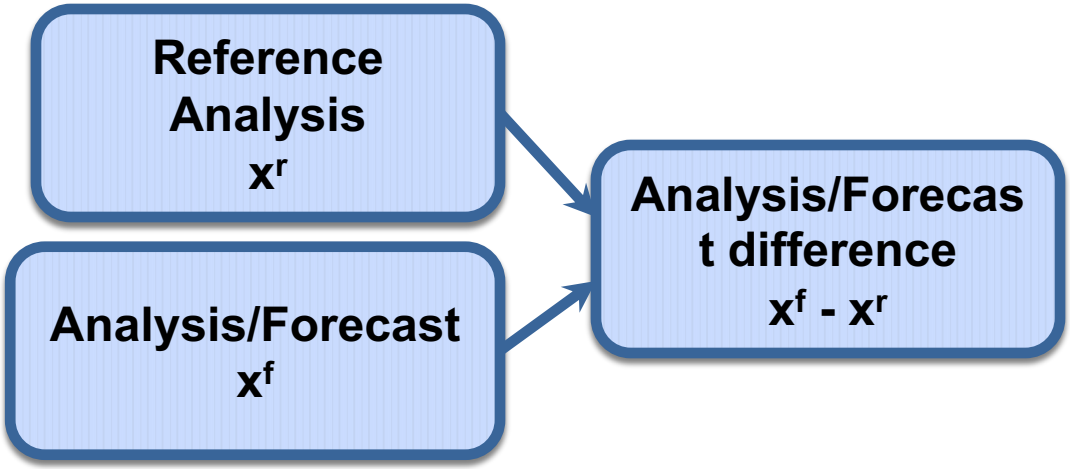
verif_obs_time_series.ncl

Observation verification: Output



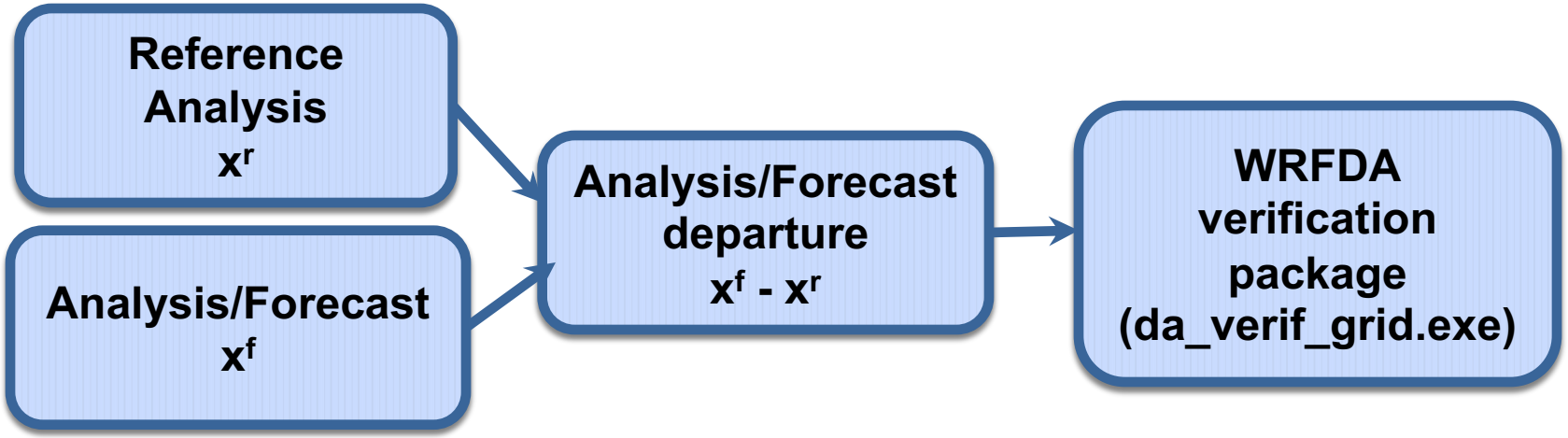
verif_obs_time_average.ncl

Grid-based verification



- Reference analysis can come from NCEP GFS or ECMWF or own analysis
- The general idea of verification against reference analysis
  - Interpolate your reference analysis to the same grid as the model analysis/forecast, or the opposite
  - Calculate differences on the same grids

Grid-based verification



- The verification suite for analysis is run with a script
  - WRFDA/var/script/da_verif_grid.ksh
- Source code for this executable can be found in WRFDA/var/da/da_verif_grid
- Grid-based verification is run using wrfout files
- The analysis domain must be exactly the same as the domain being verified: same horizontal and vertical resolution
  - Four options are available: the whole domain, sub_domain, water, and land
  - da_verif_grid.exe calculates all four domains every time, you do not need to specify

Grid-based verification

- A successful run will produce the following output files containing the variables and scores you choose

MU_time_series_06 T2M_time_series_06 V10M_time_series_06
MU_time_series_06_sub T2M_time_series_06_sub V10M_time_series_06_sub
MU_time_series_06_sub_land T2M_time_series_06_sub_land V10M_time_series_06_sub_land
MU_time_series_06_sub_water T2M_time_series_06_sub_water V10M_time_series_06_sub_water
PSFC_time_series_06 T_time_series_06 V_time_series_06
PSFC_time_series_06_sub T_time_series_06_sub V_time_series_06_sub
PSFC_time_series_06_sub_land T_time_series_06_sub_land V_time_series_06_sub_land
PSFC_time_series_06_sub_water T_time_series_06_sub_water V_time_series_06_sub_water
Q2M_time_series_06 U10M_time_series_06 WV_time_series_06
Q2M_time_series_06_sub U10M_time_series_06_sub WV_time_series_06_sub
Q2M_time_series_06_sub_land U10M_time_series_06_sub_land WV_time_series_06_sub_land
Q2M_time_series_06_sub_water U10M_time_series_06_sub_water WV_time_series_06_sub_water
QVAPOR_time_series_06 U_time_series_06 Z_time_series_06
QVAPOR_time_series_06_sub U_time_series_06_sub Z_time_series_06_sub
QVAPOR_time_series_06_sub_land U_time_series_06_sub_land Z_time_series_06_sub_land
QVAPOR_time_series_06_sub_water U_time_series_06_sub_water Z_time_series_06_sub_water

- For plotting, “fnames_sfc”, “fnames_upr”, “header_main” need to be generated from da_verif_grid.ksh
  - An example from fnames_sfc:

Grid-based verification

- A successful run will produce the following output files containing the variables and scores you choose

MU_time_series_06 T2M_time_series_06 V10M_time_series_06
MU_time_series_06_sub T2M_time_series_06_sub V10M_time_series_06_sub
MU_time_series_06_sub_land T2M_time_series_06_sub_land V10M_time_series_06_sub_land
MU_time_series_06_sub_water T2M_time_series_06_sub_water V10M_time_series_06_sub_water
PSFC_time_series_06 T_time_series_06 V_time_series_06
PSFC_time_series_06_sub T_time_series_06_sub V_time_series_06_sub
PSFC_time_series_06_sub_land T_time_series_06_sub_land V_time_series_06_sub_land
PSFC_time_series_06_sub_water T_time_series_06_sub_water V_time_series_06_sub_water
Q2M_time_series_06 U10M_time_series_06 WV_time_series_06
Q2M_time_series_06_sub U10M_time_series_06_sub WV_time_series_06_sub
Q2M_time_series_06_sub_land U10M_time_series_06_sub_lan
Q2M_time_series_06_sub_water U10M_time_series_06_sub_wa
QVAPOR_time_series_06 U_time_series_06 Z_time
QVAPOR_time_series_06_sub U_time_series_06_sub Z_
QVAPOR_time_series_06_sub_land U_time_series_06_sub_land
QVAPOR_time_series_06_sub_water U_time_series_06_sub_wate

9
5
MU_time_series_06
MU (Pascal)
PSFC_time_series_06
PSFC (Pascal)
U10M_time_series_06
U10M (m/s)
V10M_time_series_06
V10M (m/s)
T2M_time_series_06
T2M (K)
Q2M_time_series_06
Q2M (g/Kg)

- For plotting, “fnames_sfc”, “fnames_u … ain” need to be generated from da_verif_grid.ksh
  - An example from fnames_sfc:

Grid-based verification

- Defining your sub_domain is available
  - For example, defining ISTART/IEND/JSTART/JEND in the wrapper script for specifying them in da_verify_grid.sh:

    &sub_domain
    istart = ${ISTART}
    iend = ${IEND}
    jstart = ${JSTART}
    jend = ${JEND}

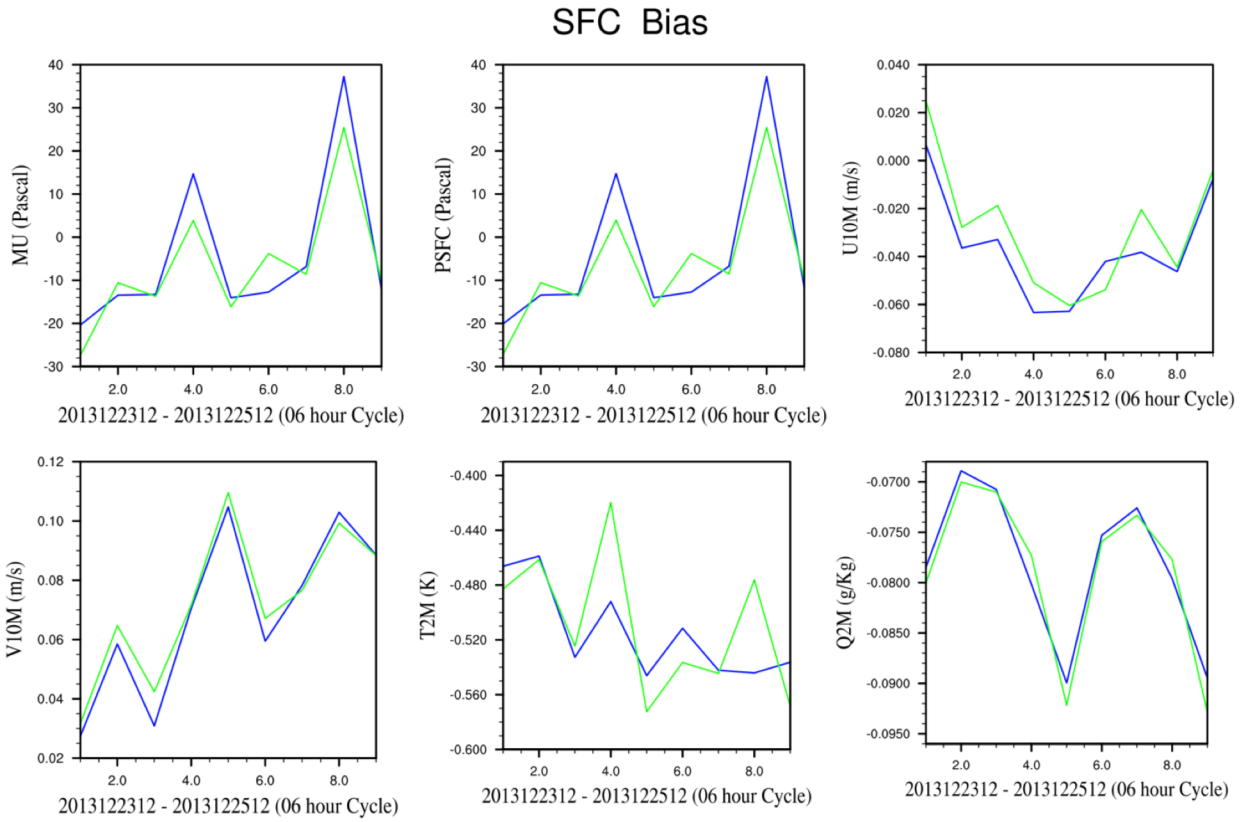
    The default values are ISTART=1, IEND=10000, JSTART=1, JEND=10000 (10000 is a large enough number for assuming the default sub_domain is the whole domain)
- Sub_domain over water/land are also available
  - Note that the variable "DOMAIN" in da_verify_grid.sh means the domain of your experiments, not for selecting the interested domain of your verification
  - The default plots are for the whole domain only, e.g. "T2M_time_series_06" is the file used for plotting T2m. You will need to change the file names in fnames_* from da_verif_grid.ksh, if you want to get error statistics for other domains.

Grid-based verification

Variables declared in script (da_verif_grid.ksh):

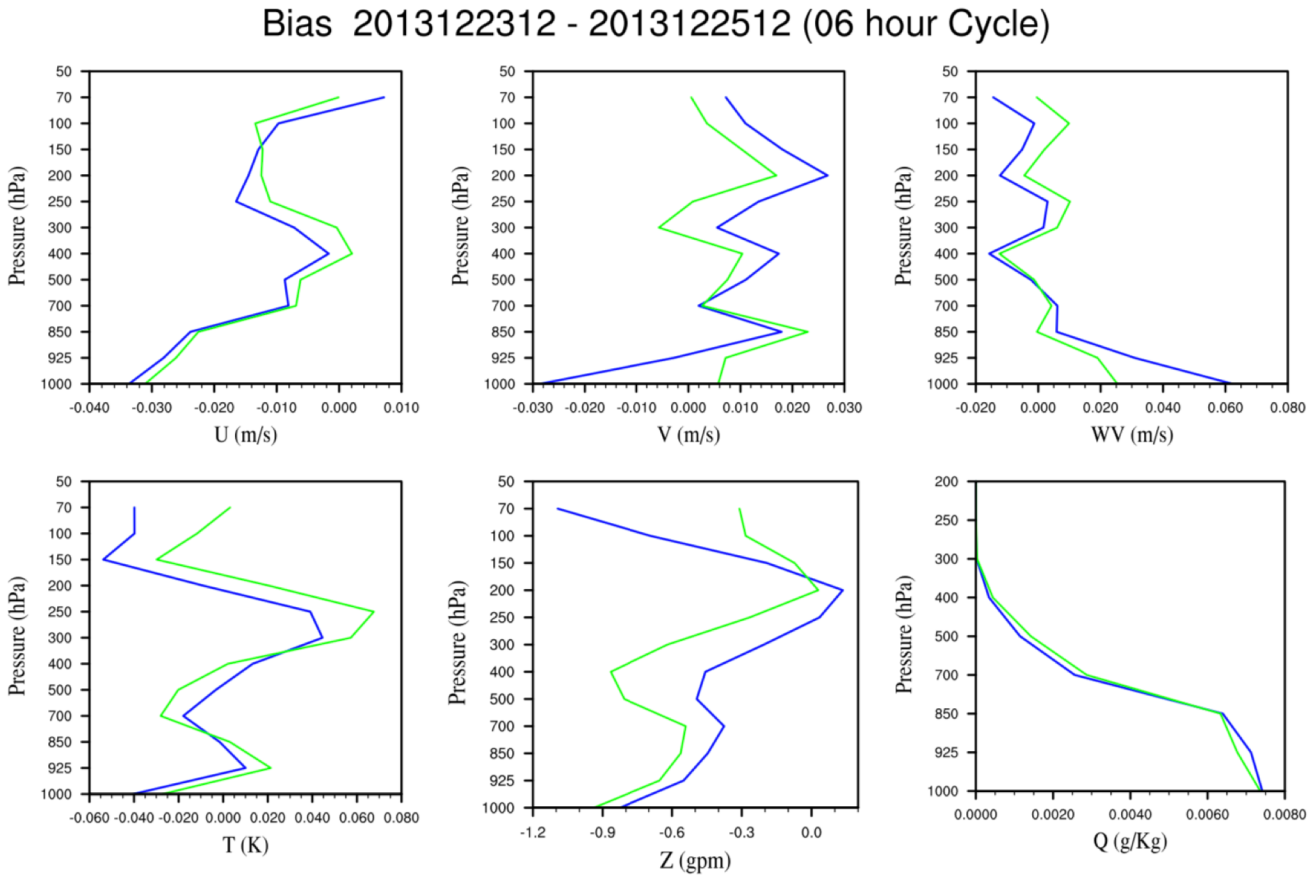
WRFVAR_DIR: WRFDA main directory (source code)
START_DATE: Verification starting date (yyyymmddhh)
END_DATE: Verification ending date (yyyymmddhh)
INTERVAL: Cycling interval in hours
REG_DIR: Directory holding forecast sub-directories for each experiment
NUM_EXPT: Number of experiments
EXP_DIR: Experiment directory name(s) in REG_DIR
EXP_NAMES: Experiment name(s)
RUN_DIR: Where output will be produced
DESIRED_LEVELS: The pressure levels desired for diagnostics (in hPa)
DESIRED_SCORES: Which diagnostics are desired (RMSE, BIAS, ABIAS)
VERIFY_HOUR: 00 for analysis; 12, 24, etc. corresponding to the desired forecast hour verification
CONTROL_EXP_DIR: Directory name of the reference analysis to be used for verification
VERIFY_ITS_OWN_ANALYSIS: Set to “true” if there is no control analysis; the experiment’s own analysis will be used for verification
VERIFY_DATE_RANGE: Title of x-axis in the output plots
PLOT_WKS: “x11“ to display plots on screen, "pdf" to save as pdf files

Grid verification: Output



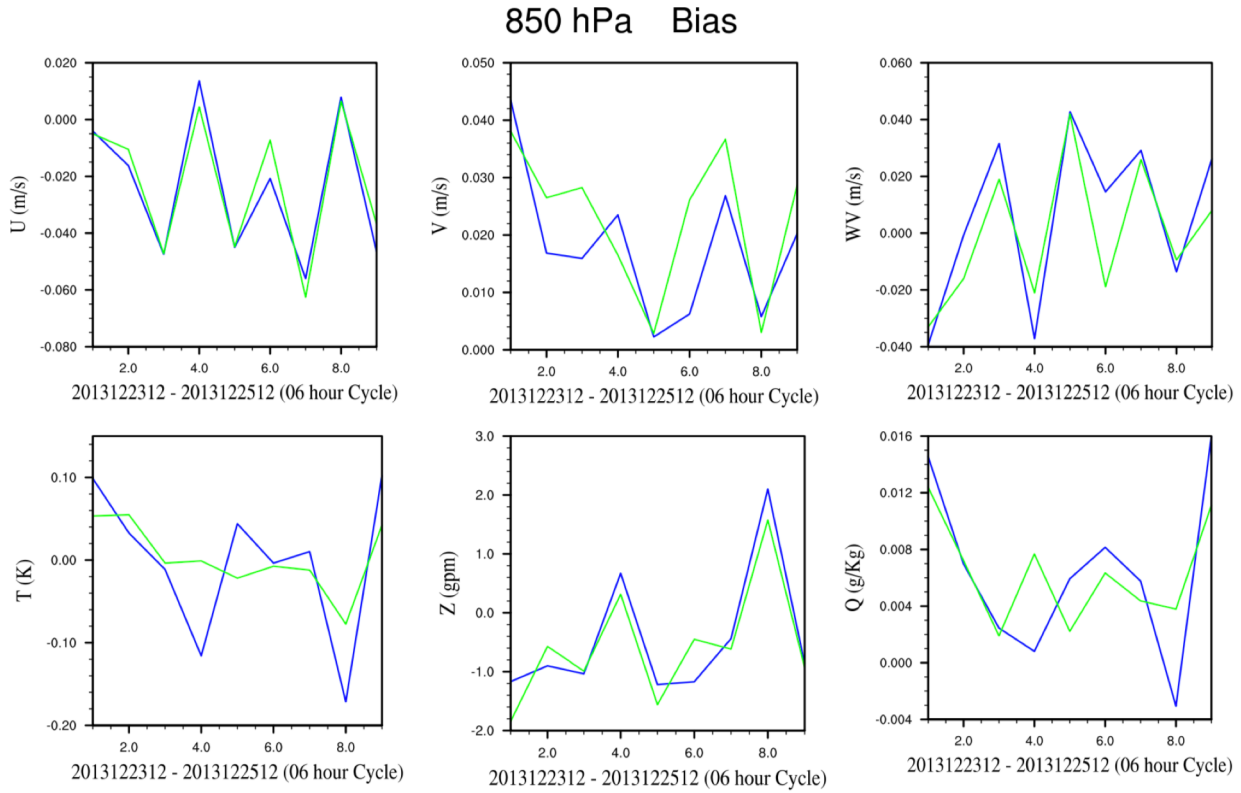
verif_obs_time_series.ncl

Grid verification: Output



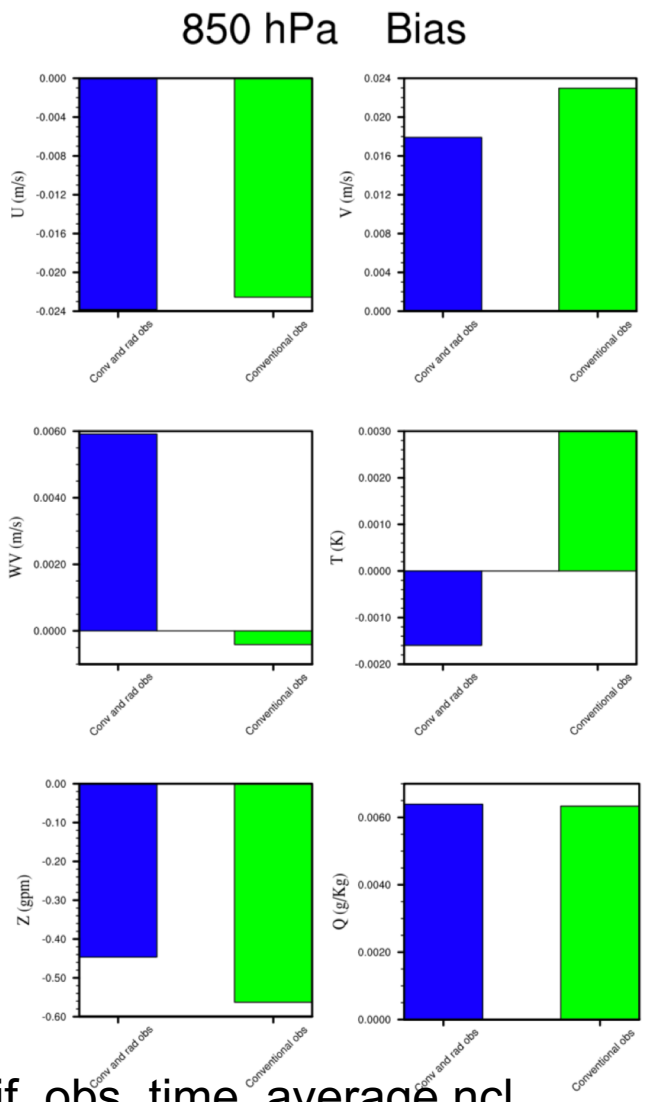
verif_obs_vert_profile.ncl

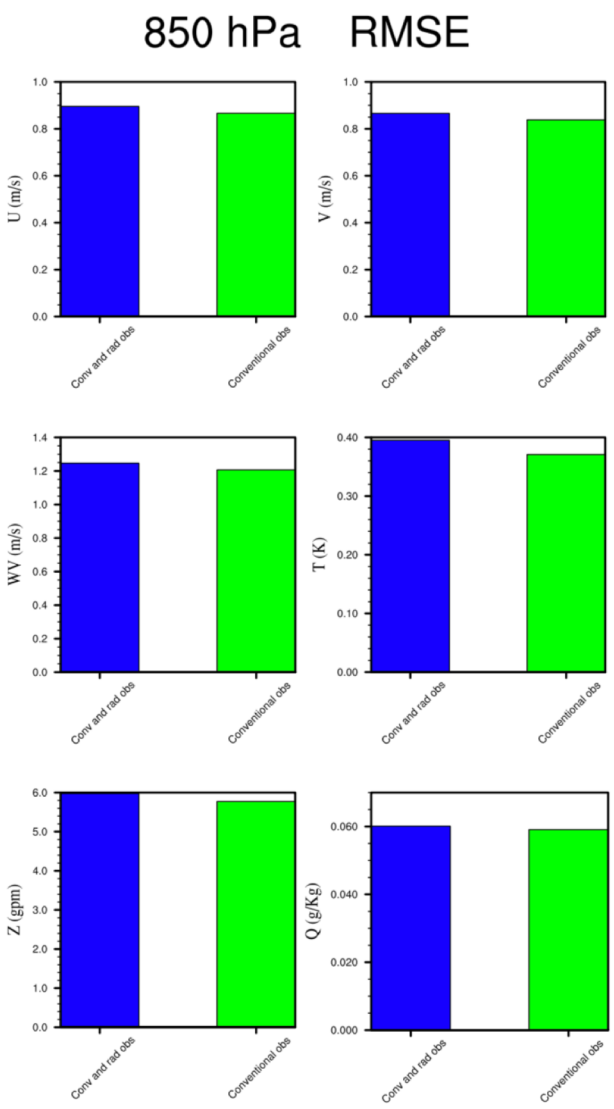
Grid verification: Output



verif_obs_time_series.ncl

Grid verification: Output

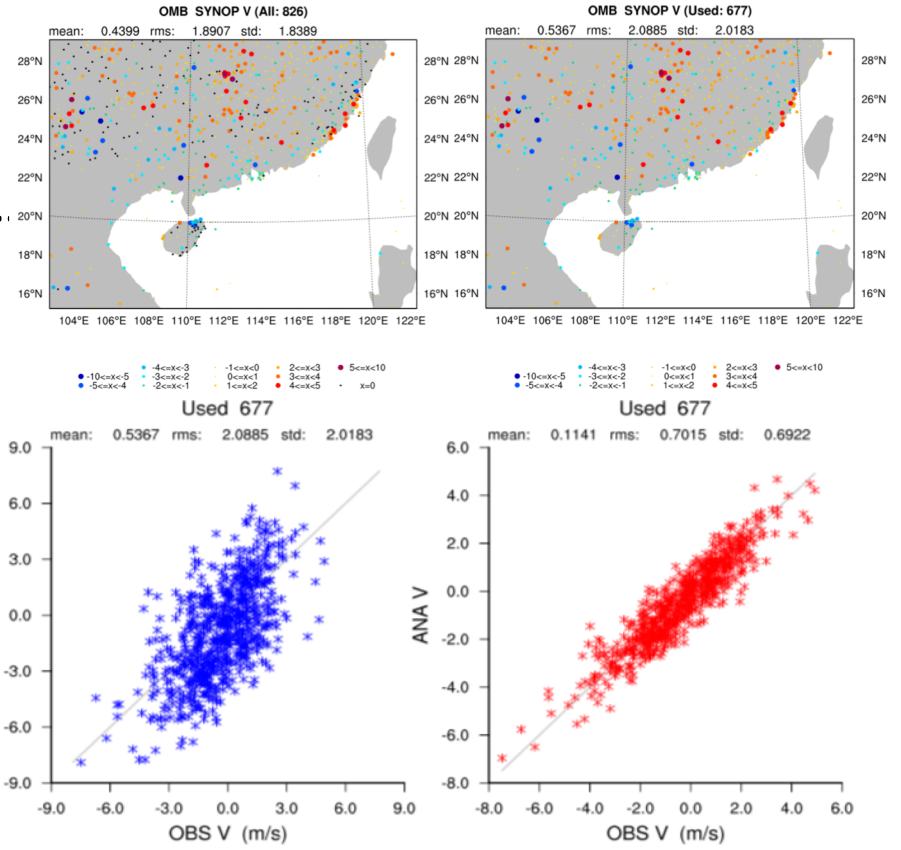


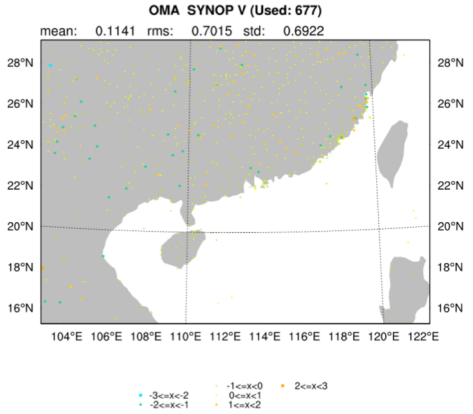


verif_obs_time_average.ncl

Graphics: Observation-space diagnosis

- plot_ob_ascii_loc.ncl
  - Read “ob.ascii” file – input obs
  - Based on WMO code number (‘FM-35’ ..
  - Plot obs locations
- plot_gts_omb_oma.ncl
  - Read “gts_omb_oma_0*” file – output
  - Include radar observations
  - Based on WMO code name (‘synop’....
  - plot obs distributions and scatter plots





A example from synop obs

Graphics: Observation-space diagnosis

- convert_gts_omb_oma_to_netcdf.ncl
  - convert gts_omb_oma_01 to be in netCDF format;
  - Netcdf format is more user-friendly and makes your own plotting easier;
  - Convenient to understand the contents in gts_omb_oma_01 file
  - Each observation type stored separately;

```
(0)     Reading data from ../3dvar/gts_omb_oma_01
(0)     30939 lines read from file
Mon Jun 24 15:51:05 UTC 2019
(0)     ---- Converting 750 observations of type SYNOP to netCDF ----
(0)     ---- Converting 2408 observations of type METAR to netCDF ----
(0)     ---- Converting 51 observations of type SHIPS to netCDF ----
(0)     ---- Converting 2499 observations of type GEOAMV to netCDF ----
(0)     ---- Converting 86 observations of type SOUND to netCDF ----
(0)     ---- Converting 86 observations of type SONDE_SFC to netCDF ----
(0)     ---- Converting 216 observations of type AIREP to netCDF ----
(0)     ---- Converting 85 observations of type PILOT to netCDF ----
(0)     ---- Converting 2126 observations of type QSCAT to netCDF ----
(0)     ---- Converting 61 observations of type PROFILER to netCDF ----
(0)     ---- Converting 336 observations of type BUOY to netCDF ----
(0)     ---- Converting 3 observations of type GPSREF to netCDF ----
Mon Jun 24 15:51:07 UTC 2019
```

- Each observation type stored separately
  - gts_omb_oma_airep.nc
  - gts_omb_oma_buoy.nc
  - gts_omb_oma_geoamv.nc
  - gts_omb_oma_gpsref.nc
  - gts_omb_oma_metar.nc
  - gts_omb_oma_pilot.nc
  - gts_omb_oma_profiler.nc
  - gts_omb_oma_qscat.nc
  - gts_omb_oma_ships.nc
  - gts_omb_oma_sonde_sfc.nc
  - gts_omb_oma_sound.nc
  - gts_omb_oma_synop.nc

- Each observation file contains:
  - NLEV (site)
  - LATS (site); LONS (site)
  - STDI (site, char_len)
  - UOBS (site), UOMB (site), UQCF (site), UERR (site), UOMA (site)
  - VOBS (site), VOMB (site), VQCF (site), VERR (site), VOMA (site)

  ……

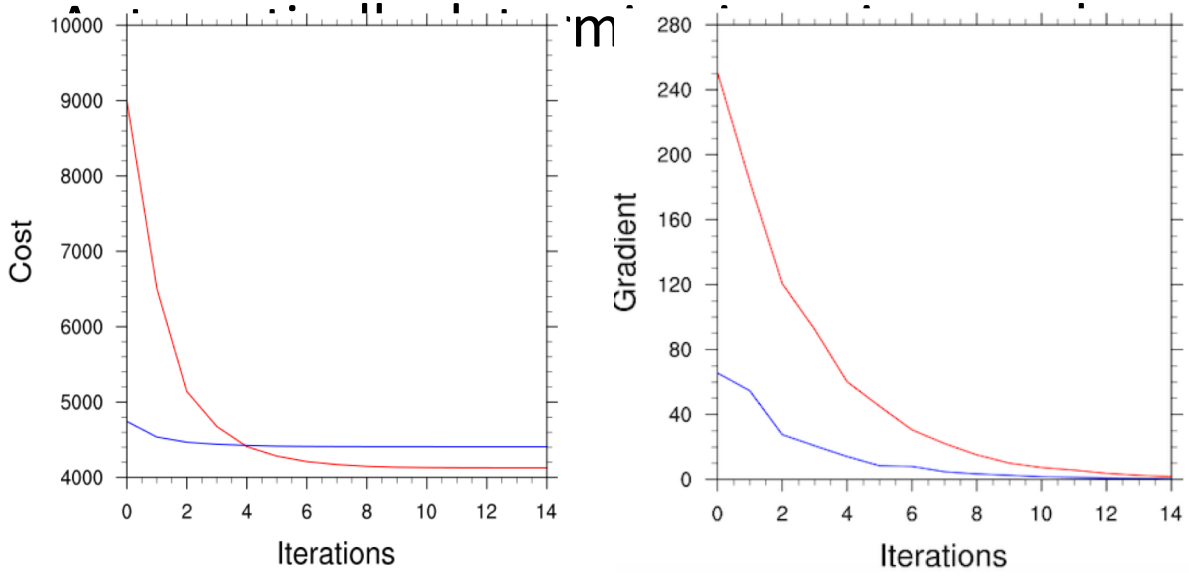
Graphics:Observation-space diagnosis

- plot_cost_grad_fn.ncl
  - Read "cost_fn" and "grad_fn" files generated by turning on

    &wrfvar11

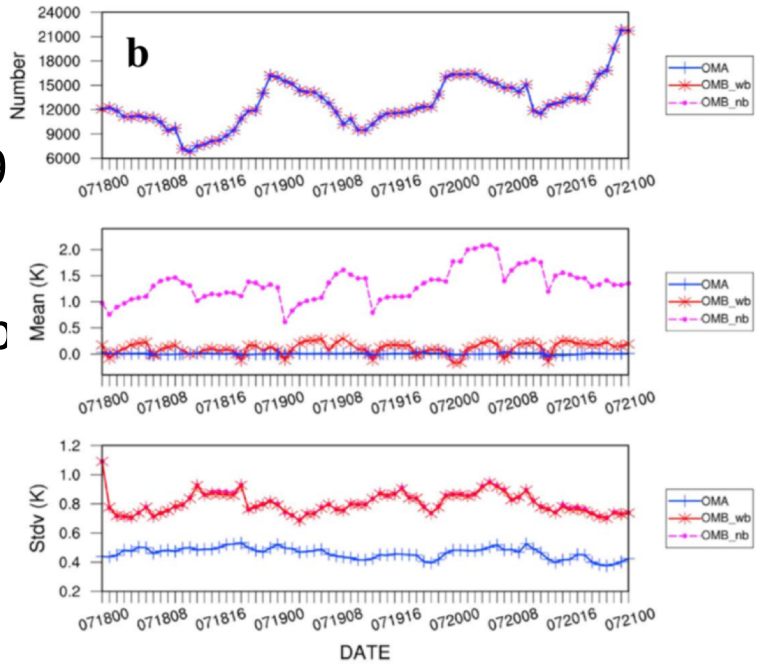
    calculate_cg_cost_fn=true

    write_detail_grad_fn=true

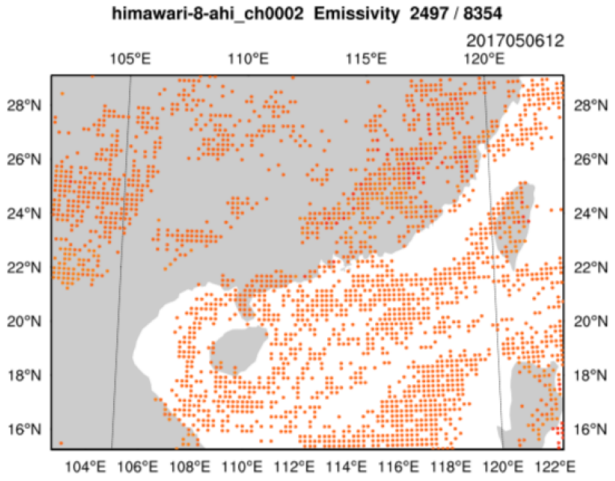


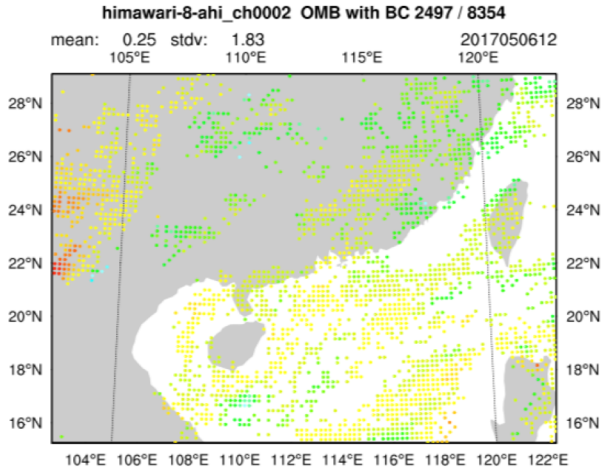
Graphics: Observation-space diagnosis

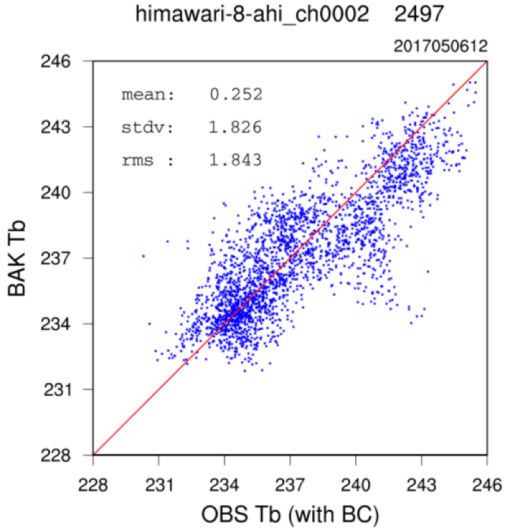
- plot_rad_diags.ncl
  - plot radiance brightness temperature distributions, scatter plots, histograms, and time series.
  - Called by da_rad_diags.ksh
    - Read 0*_inv* and 0*_oma* output ASCII files for different satellites
    - Convert ASCII files to a NETCDF file (var/da/da_monitor/da_rad_diags.f9
    - Specify your settings: DATE, Path, Exp name, Satellite name, outer loop

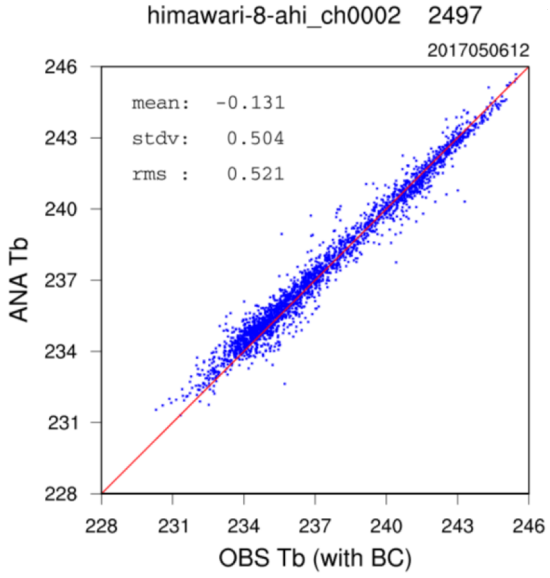


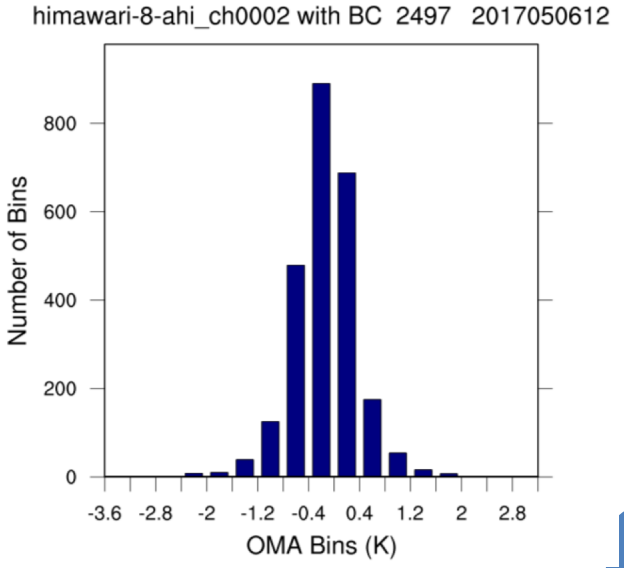
Graphics: Observation-space diagnosis





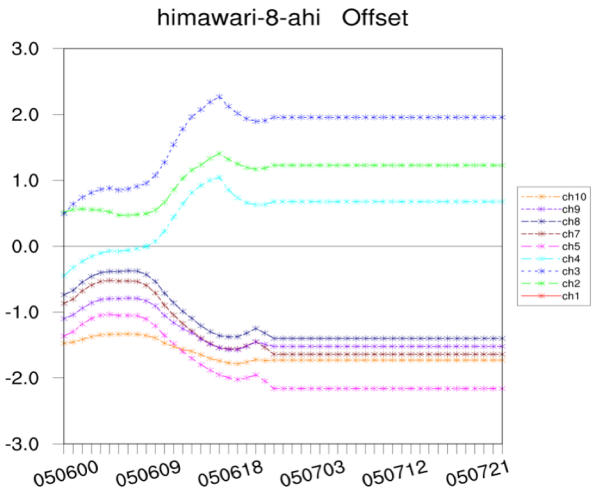


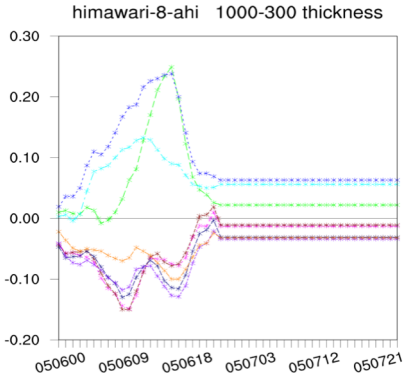


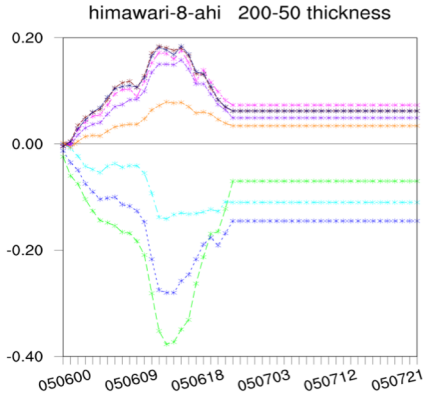


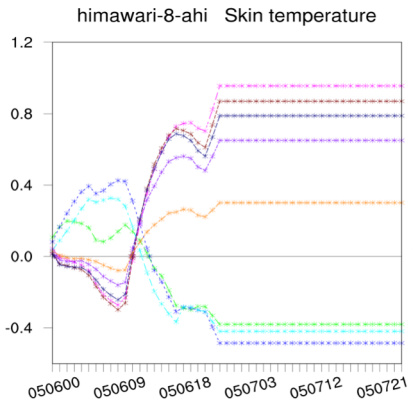
Graphics: Observation-space diagnosis

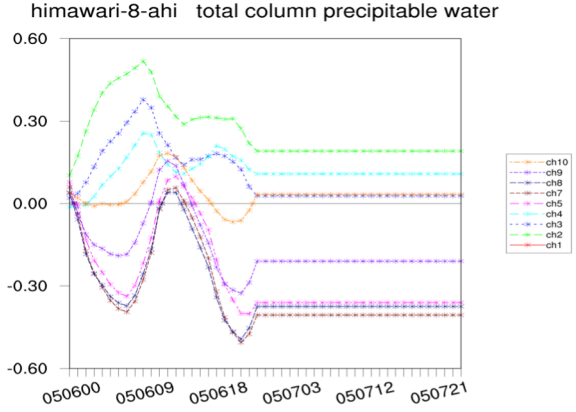
- plot_rad_varbc_param.ncl
  - Read ‘VARBC.*’ file





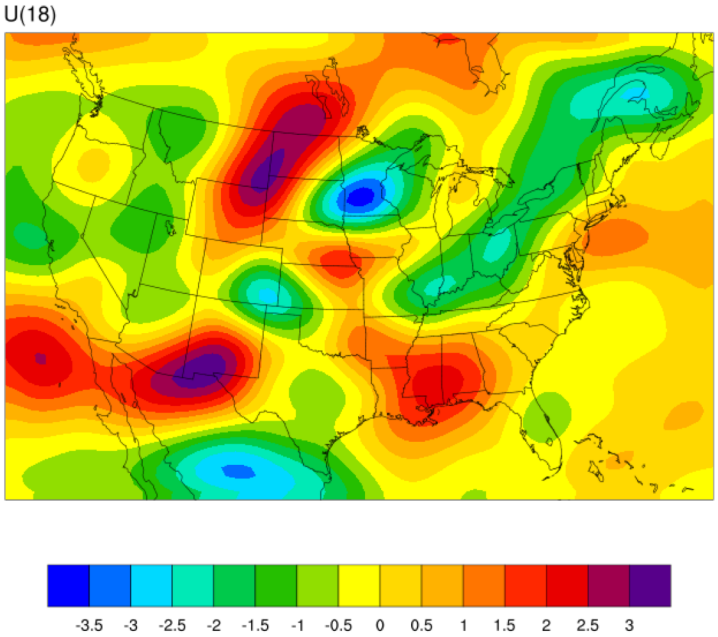






Graphics: Model-space diagnosis

- WRF-Var_plot.ncl
  - Can be used to make a plot of the difference between two WRF files (for example, fg and wrfvar_output)
  - Need to load WRF_contributed.ncl.test
  - Increments on model levels



u increment @ 8th model level