Document for Stage0 (WRF model)

in Background Error Statistics Calculation

Yong-Run Guo

Mesoscale and Microscale Meteorology Division

National Center for Atmospherics Research

Boulder, Colorado, USA

1. Overview

The Background Error Statistics (BES) file is one of main input files for WRF 3D-Var system. To a certain extent, the BES determined the performance of a 3D-var system. There are several methods proposed to derive the BES (Derber, Fisher). Especially for the WRF 3D-Var application, a program for BES calculation has been developed. This program includes several parts of the codes: Stage0, Stage1, Stage2, Stage3, Stage4, and Diags. The tasks of each of the parts are

Stage0: Read in the user's model output data, either a series of the forecasts initiated at the consecutive times with a regular interval or a set of ensemble forecasts, and compute the difference fields for streamfunction, potential velocity, temperature, relative humidity, and surface pressure.

Stage1: Create the bins depending on the latitude of the Y-direction, and remove the bias from the difference fields for each of the bins.

Stage2: Calculate the regression coefficients for balanced potential velocity, temperature, and surface pressure with the streamfunction for each of bins, and produce the unbalanced potential velocity, the unbalanced temperature, and the unbalanced surface pressure.

Satge3: Compute the local (depending on the bins) and global eigenvectors/eigenvalues for the control variables: streamfunction, unbalanced potential velocity, unbalanced temperature, relative humidity, and unbalanced surface pressure.

Stage4: Compute the horizontal scale length used in recursive filter for the regional 3D-Var applications or the power spectrum for the global 3D-Var applications for each of the control variables.

Diags: Gather the results from Stage2, Stage3, and Stage4, and write out a BES file.

This program has the following features:

1. The BES produced by this program is applied to the cv_options = 5 for WRF 3D-Var, which means i) the control variables are same as the cv_options = 3 but by using the coefficients projected on the vertical modes; ii) the horizontal covariance is modeled by the first order recursive filter with multiple passes; and iii) the vertical covariance is expressed by the eigenvector derived from the vertical covariance matrix, same as cv_options = 2.

2. The program can be started at any of the stages when the input files have been produced by the previous stages.

3. For Stage3 and Stage4 (the most expensive parts), the codes can be applied to the control variables one by one. So the computation could be completed on several machines simultaneously. It's equivalent to the parallel runs and speed-up the BES file produced.

4. Only the code for Stage0 is an interface for the user's applications. The codes from Stage1 to Diags are transparency for users, and in general, users do not need to touch them.

In this document, we will give the detailed description of the Stage0 code, especially for WRF model's application, and the user's guide to run the Stage0 program.

2. Description of the Stage0 code for WRF model

A, Task of the stage0

There are 5 tasks in Stage0:

1) Read in the forecast fields at the specific times from a series of the model forecast files initiated at the different dates.

At a specific date, for use of the NMC approach, two sets of the forecast fields at the same valid time from two forecast files should be read in while for use of the ENS (ensemble forecasts) approach, one set of the forecast fields from each of the ensemble members.

2) Convert the original model forecast fields to WRF 3D-Var analysis fields at A-grid: u and v components of wind, temperature, relative humidity, and surface pressure.

3) Compute the difference fields.

For the NMC approach, the difference fields between two forecasts at the same valid time are computed for each of the forecast dates.

For the ENS approach, the ensemble mean over the ensemble members is computed first, the difference between the ensemble forecasts and the ensemble mean are computed.

4) Compute the difference fields of the streamfunction and potential velocity based on the u and v difference fields by solving the Poisson equation using SIN_FFT method.

Note that i) all the operations for the conversion from wind components to the streamfunction and potential velocity are linear, so theoretically doing the conversion first and difference second should be same as the difference first and conversion second; ii) the solutions of the streamfunction and potential velocity derived from the u and v components in a limited area are not unique but depend on the lateral boundary specification. Using the SIN FFT to solving the Poisson equation implies that the zero lateral boundaries are used.

5) Write out the difference fields for each of the dates.

B, Details of the code structure

1) Main program: *wrf3dvar/main/gen_be/**gen_be_stage0.F***

The structure of this program is very similar to *wrf3dvar/main/da_3dvar.F* but to replace "*call   da_3dvar_interface*" by "*call   da_gen_be_stats_interface*". The module "*module_da_gen_be_stats_interface.F*" is added under the directory "*wrf3dvar/da_3dvar*". With this module, the subroutine "*da_3dvar/src/da_stats_be/da_stats_be.F*" is called, and then in "*da_stats_be.F*", subroutines: "*da_stats_be/da_init_be*", "*DA_Gen_Be_Stats/DA_Stats_Namelist*", and "*DA_Gen_Be_Stats/DA_Statistics_Step0*" are called sequentially. Actually, the subroutine "***DA_Statistics_Step0***" is the core part of the stage0 program; other parts of the code are only for matching with the wrf3dvar framework, such as the netCDF_IO, mpp capabilities, etc. If users develop their own stage0 program

with their own model forecasts, the procedure can be much simplified, directly starting from the code of ***DA_Statistics_Step0.F*** with the necessary modifications.

The flowchart is summarized as below.

```
┌─────────────────────────────────────────────┐
│   wrf3dvar/main/gen_be/gen_be_stage0         │
└─────────────────────────────────────────────┘
                      │
                      ▼
┌─────────────────────────────────────────────┐
│  wrf3dvar/da_3dvar/da_gen_be_stats_interface │
└─────────────────────────────────────────────┘
                      │
                      ▼
┌─────────────────────────────────────────────┐
│  wrf3dvar/da_3dvar/src/da_stats_be/stats_be  │
└─────────────────────────────────────────────┘
                      │
                      ▼
┌──────────────────────────────────────────────────────────┐
│  wrf3dvar/da_3dvar/src/da_stats_be/da_init_be             │
│  wrf3dvar/da_3dvar/src/DA_Gen_Be_Stats/DA_Stats_Namelist  │
│  wrf3dvar/da_3dvar/src/DA_Gen_Be_Stats/DA_Statistics_Step0│
└──────────────────────────────────────────────────────────┘
```
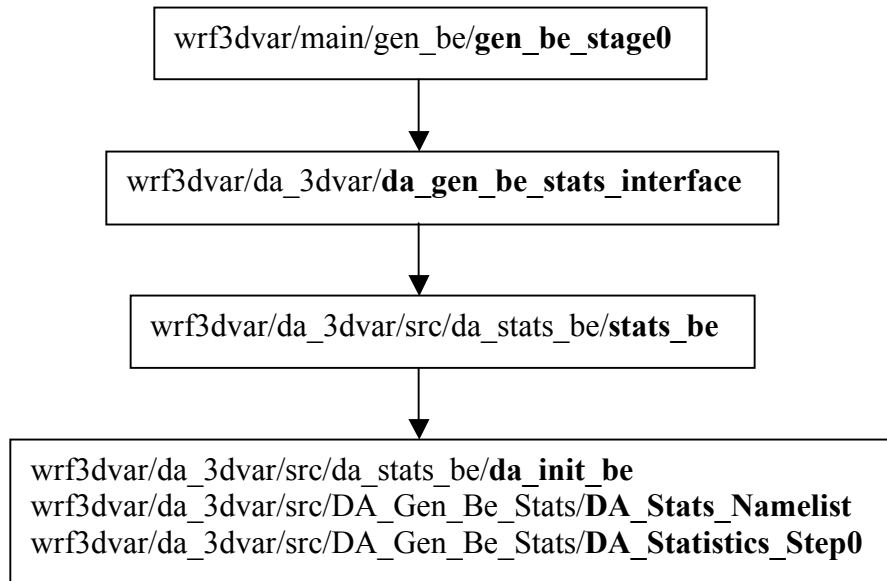
2) Subroutine: ***DA_Statistics_Step0***

This subroutine includes two loops: loop over the *file_date* and loop over the *ensemble member*. Based on the *file_date* and *ensemble member index*, the specific file name will be formed by calling subroutine "*DA_Make_Filename*", then the needed data are read by "call *med_initialdata_input_3dvar*", a standard WRF 3DVar subroutine to read the netCDF file, and transfer to *u, v, t, rh*, and *Psfc* by calling WRF 3DVar subroutine "*DA_Setup_FirstGuess*", and assign these variables to the variables named by ***xb24*** and ***xb12***. For the BACKGROUND_OPTION = 2 (ensemble approach), only variable ***xb24*** need

to be read in and **xb12** is stored the ensemble mean computed by subroutine "*DA_Make_Ensemble_Mean*".

Until now, the two sets of the fields, *u, v, t, rh*, and *Psfc*, for the difference calculation are ready. Then the difference fields are created by calling "*DA_Difference*", and the difference of the *u, v* components will be converted to streamfunction *psi* and potential velocity *chi* by calling "*DA_New_Statistics_Variable*".

Finally, the difference file will be written out for the specific date and ensemble member by calling:"*DA_Write_Diff*". Each of the binary file contains a header record: *time* (character*10), *ide, jde*, and *kde* (the dimensions of domain, integer), and 7 data records, difference of *psi, chi, t, rh, Psfc*, and the full fields of *h* and *lat*.
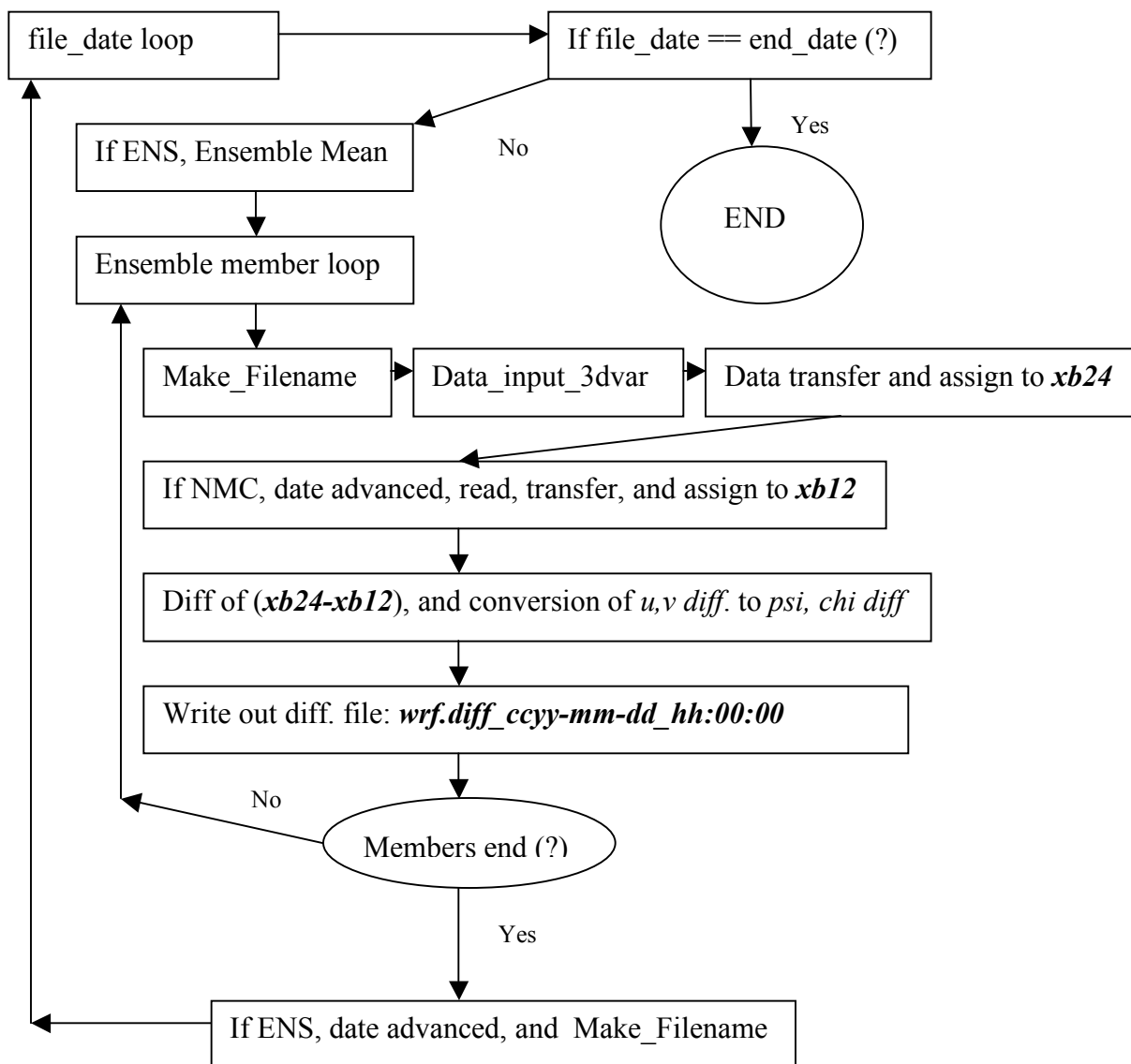
3) Convention of the input and output file names

- Input file:

  The input file name is composed of three parts: i) directory_name and file_head, ii) file_date; and iii) ensemble member index. The directory_name and file_head are provided through the namelist file: namelist.stats. The file_date has 19 characters, like ccyy-mm-dd_hh:00:00, and the ensemble member index are one or two digits, i.e. currently only maximum of 99 members are allowed for each of the times So the input file name is trim(directory_name)/trim(file_head)_ccyy-mm-dd_hh:00:00.trim(index), such as "***GEN_BE_data/wrfout_d01_2002-01-01_00:00:00.5***". For NMC approach, there is no "index" part, i.e. "***GEN_BE_data/wrfout_d01_2002-01-01_00:00:00***".

- Output file name:

The output file names are formed by the program stage0 and recognized by the stage1 program. The file names are hardwired by program as wrf.diff_ccyy-mm-dd_hh:00:00.index, such as "***wrf.diff_2002-01-01:00:00.5***", and for NMC approach, there is no "index" part, i.e. "***wrf.diff_2002-01-01:00:00***".

| file_date loop | → | If file_date == end_date (?) |

If ENS, Ensemble Mean   No   Yes → END

↓

Ensemble member loop

↓

| Make_Filename | ▸ | Data_input_3dvar | ▸ | Data transfer and assign to *xb24* |

↓

If NMC, date advanced, read, transfer, and assign to *xb12*

↓

Diff of (***xb24-xb12***), and conversion of *u,v diff.* to *psi, chi diff*

↓

Write out diff. file: ***wrf.diff_ccyy-mm-dd_hh:00:00***

↓

No ←   Members end (?)

Yes ↓

If ENS, date advanced, and Make_Filename

4) Directories special for program stage0

There are two directories special for program stage0: ***da_3dvar/src/DA_Gen_Be_Stats*** and ***da_3dvar/src/da_stats_be***. As mentioned before, the main program *gen_be_stage0.F* and interface module *da_gen_be_stats_interface*" are resided in ***main/gen_be*** and ***da_3dvar***, respectively.

3. User guide for program Stage0

1) Compile and running

The compiling procedure is similar to rgw wrf3dvar but need *two* compiling steps:

configure

compile 3dvar

compile gen_be

The executables for *gen_be_stage0, 1, 2,…,* are under the directory *main/gen_be*.

2) Shell script for running Stage0

The shell scripts for running Stage0 program is ***/run/gen_be/da_wrf_stage0.csh***

In this script, there are 4 parts need to be modified by users for their applications

i) setup the directories for WRF 3DVar source code, input data, and job running;

ii) setup the domain dimension and grid size;

iii) Setup the information for job running, including the start and end times, which type of the BES produced, the prefix of the filename, file interval, and first forecast time in hours.

iv) Namelist.stats. In general use, users do not need to touch this part since most of namelist variables can obtain the values from the above part 3). Only in case of debugging, you may need to set a non-zero value to *PRINT_DETAIL*, and .true. to *TEST_TRANSFORMS*.

Two more namelist variables: *from_mss* and *mss_directory*, are used to acquire the input data files from the NCAR Massive Storage System (MSS) directly. This may be useful for use of the ensemble forecast approach with the large number of ensemble for many dates. If users do not have NCAR MSS available, ignore these variables.

3) Namelist.stats

There are 3 records in the namelist file:

&FILERECD

DIRECTORY   ; the directory for input data file (character (len=120))

FILE_HEAD   ; the prefix of the input file names (character (len=120)). For example, the file_head is 'wrfout_d01' for input file: 'wrfout_01 _2002-01-01_00:00:00'.

BGN_DATE   ; the starting time for BES calculation (character (len=24)): ccyy-mm-dd_hh:00:00

END_DATE   ; the ending time for BES calculation.

TEST_TRANSFORMS ; .FALSE. not do the transform test, .TRUE. doing the test of the transform between (u,v) and (psi,chi).

PRINT_DETAIL   ; =0 no details printed, >0 print more things for debugging

From_mss   ;.FALSE. not use NCAR MSS, .TRUE. acquire the input data from NCAR   MSS during the execution.

Mss_directory   ; the input data directory in NCAR MSS (character (len=120)).

&TIMERECD

T_FORECAST1   ; the earlier forecast time for BES calculation

T_FORECAST2   ; the later forecast time for BES calculation, i.e. T_FORECAST2 = T_FORECAST1 + FILE_INTERVAL. So when the T_FORECAST1 and FILE_INTERVAL are specified, you don't need to specify this variable to avoid inconsistent.

FILE_INTERVAL ; Time interval in hour of the input data files.

&ANALTYPE

BACKGOUND_OPTION ; BES type: 1 = NMC, 2 = ENS

MEMBERS   ; If BACKGROUND_OPTION = 1, MEMBERS = 1,

                                    If BACKGROUND_OPTION = 2, set the number of ensemble members for each of the times to MEMBERS.

4) Shell script for running Stage1 to Diags

The script for running all other stages is ***run/gen_be/gen_be_sample.csh***

This shell script can be used to run all stages or the select stage.

5) Print and plot the background error statistics

- ***./gen_be_diags_read.exe*** can be used to print the BES in ASCII format to fort.171, 172,…..

- The shell script ***da_3dvar/utl/plot_eigen_in_be.csh*** can be used to plot the global the first five eigenvalues, eigenvectors, and the first five local eigenvalues, and the scale lengths. You just need to copy that shell script to your working directory and edit it for your application. The BES plots are easy to be produced for your check.

- Also there are some *NCL* plotting routines for plotting the correlation coefficients, eigenvalues and eigenvectors.