Algorithm (3): WRFDA 4DVAR

Zhiquan (Jake) Liu

NCAR/MMM

WRFDA Tutorial, July 2017

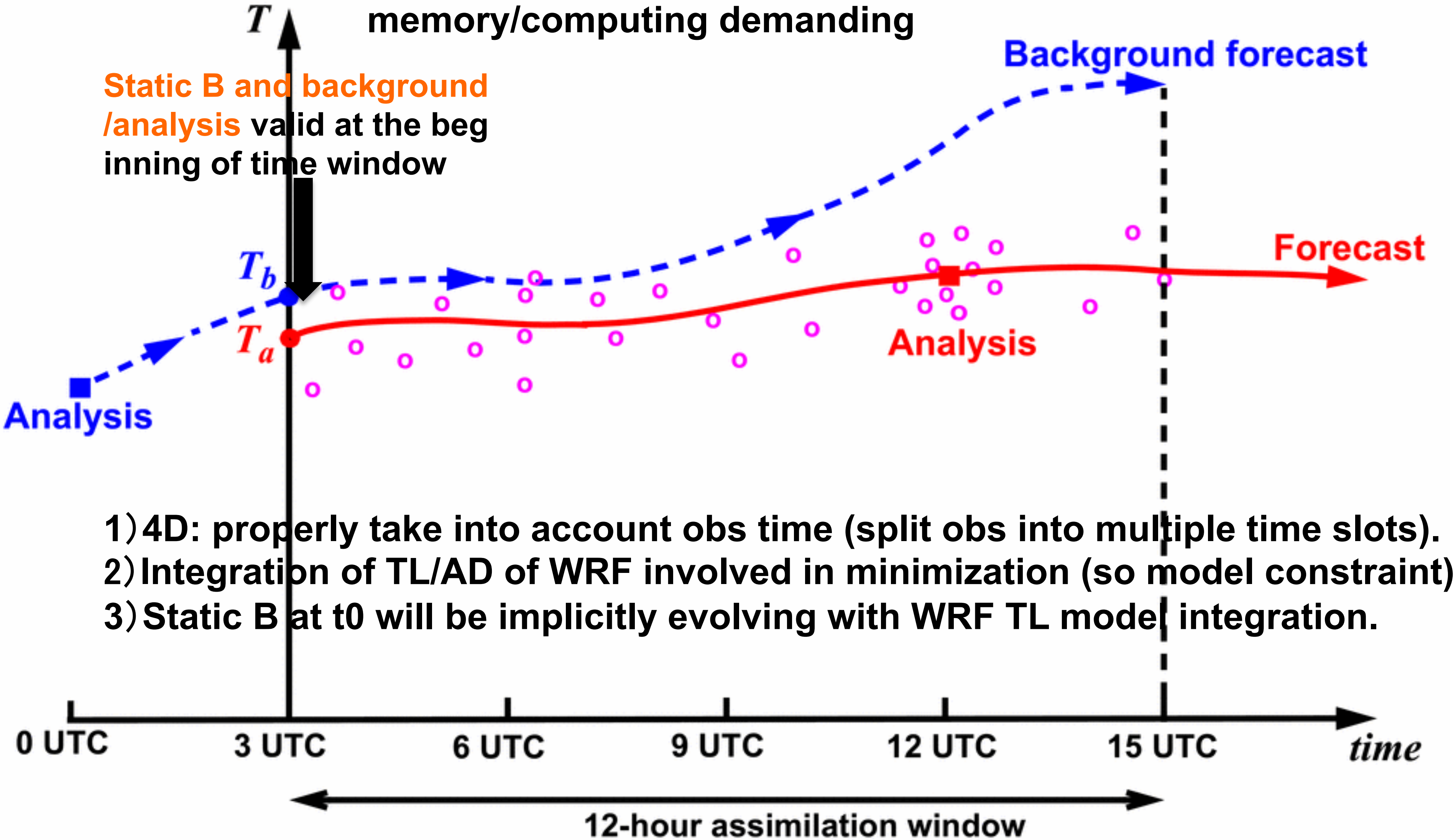
Outline

- Incremental 4DVAR
- Multi-Resolution Incremental 4DVAR
- Introduction to 4DVAR practice

4DVAR

Need develop/maintain TL/AD version of a NWP model.
memory/computing demanding

Static B and background /analysis valid at the beg inning of time window



1）4D: properly take into account obs time (split obs into multiple time slots).
2）Integration of TL/AD of WRF involved in minimization (so model constraint)
3）Static B at t0 will be implicitly evolving with WRF TL model integration.

Non-Linear 4DVAR cost function

$$J(\mathbf{x}_0) = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_0^b)^{\mathrm{T}}\mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + \frac{1}{2}\sum_{i=1}^{N}[H_i(M_i(\mathbf{x}_0)) - \mathbf{y}_i]^{\mathrm{T}}\mathbf{R}_i^{-1}[H_i(M_i(\mathbf{x}_0)) - \mathbf{y}_i]$$

(1) Analysis vector X_0 and B matrix is valid at the beginning of the assimilation time window

(2) NWP model acts as a strong constraint in the cost function

(3) [Obs – Forecast Trajectory] is calculated at different time slots within time window.

Note: *M_i means model integration to time t_i*

Incremental 4DVAR

2.2 Incremental 4DVAR Formulation

Linearization, let $\delta\mathbf{x}_0 = \mathbf{x}_0 - \mathbf{x}_0^g$ and $\delta\mathbf{x}_0^g = \mathbf{x}_0^b - \mathbf{x}_0^g$, thus $\mathbf{x}_0 = \delta\mathbf{x}_0 + \mathbf{x}_0^g$, we have

$$J(\delta\mathbf{x}_0) = \frac{1}{2}(\delta\mathbf{x}_0 - \delta\mathbf{x}_0^g)^{\mathrm{T}}\mathbf{B}^{-1}(\delta\mathbf{x}_0 - \delta\mathbf{x}_0^g) + \frac{1}{2}\sum_{i=1}^{N}[H_i(M_i(\delta\mathbf{x}_0 + \mathbf{x}_0^g) - \mathbf{y}_i]^{\mathrm{T}}\mathbf{R}_i^{-1}[H_i(M_i(\delta\mathbf{x}_0 + \mathbf{x}_0^g)) - \mathbf{y}_i]$$

$$J(\delta\mathbf{x}_0) = \frac{1}{2}(\delta\mathbf{x}_0 - \delta\mathbf{x}_0^g)^{\mathrm{T}}\mathbf{B}^{-1}(\delta\mathbf{x}_0 - \delta\mathbf{x}_0^g) + \frac{1}{2}\sum_{i=1}^{N}(\mathbf{H}_i\mathbf{M}_i\delta\mathbf{x}_0 - \mathbf{d}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathbf{H}_i\mathbf{M}_i\delta\mathbf{x}_0 - \mathbf{d}_i)$$

where $\mathbf{d}_i = \mathbf{y}_i - H_i[M_i(\mathbf{x}_0^g)]$.

(1) OMB is calculated using non-linear forecast trajectory

(2) *H* and *M* are linearized around forecast trajectory

Incremental 4DVAR in control variable space

$$J(\mathbf{v}) = \frac{1}{2}(\mathbf{v} - \mathbf{v}^g)^{\mathrm{T}}(\mathbf{v} - \mathbf{v}^g) + \frac{1}{2}\sum_{i=1}^{N}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v} - \mathbf{d}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v} - \mathbf{d}_i)$$

$$\nabla_{\mathbf{v}} J(\mathbf{v}) = (\mathbf{v} - \mathbf{v}^g) + \sum_{i=1}^{N}\mathbf{U}^{\mathrm{T}}\mathbf{M}_i^{\mathrm{T}}\mathbf{H}_i^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v} - \mathbf{d}_i) = 0$$

$$\mathbf{x}^a = \mathbf{x}^g + \delta\mathbf{x}^a = \mathbf{x}^g + \mathbf{U}\mathbf{v}^a$$ **(All variables at same resolution)**

(1) Control variable transform U is the same as in 3DVAR

(2) Need one TL forward and one AD backward integration to obtain the gradient of cost function in each inner loop iteration

Incremental 4DVAR with control variable transform

Again, control variable transform $\delta\mathbf{x}_0 = \mathbf{U}\mathbf{v}$ and $\delta\mathbf{x}_0^g = \mathbf{U}\mathbf{v}^g$. $\delta\mathbf{x}_0$ indicates that analysis increment is valid at the beginning of the 4DVAR time window. Then the cost function with respect to the control variable $\mathbf{v}$ becomes

$$J(\mathbf{v}) = \frac{1}{2}(\mathbf{v}-\mathbf{v}^g)^{\mathrm{T}}(\mathbf{v}-\mathbf{v}^g) + \frac{1}{2}\sum_{i=1}^{N}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v}-\mathbf{d}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v}-\mathbf{d}_i) \quad (19)$$

NOTE:

(1) For each outer loop, need to store forecast trajectory (each time step) and V^g in the memory.

(2) For each loop, H and M needs to be re-linearized around new forecast trajectory; $d_i=y_i-H_i(X^g_i)$ is also re-calculated and re-do QC (OMB check).

(3) 4DVAR outer loops could run at different (typically lower) resolutions, common practice at operational NWP centers (capability under development with WRFDA)

Advantages of 4DVAR

- Data can be assimilated at appropriate time, so can use frequently reported observations
- Can use “future” observations to constrain the analysis at earlier time
- NWP model as part of constraints, so propagating observation information via model dynamics and physics

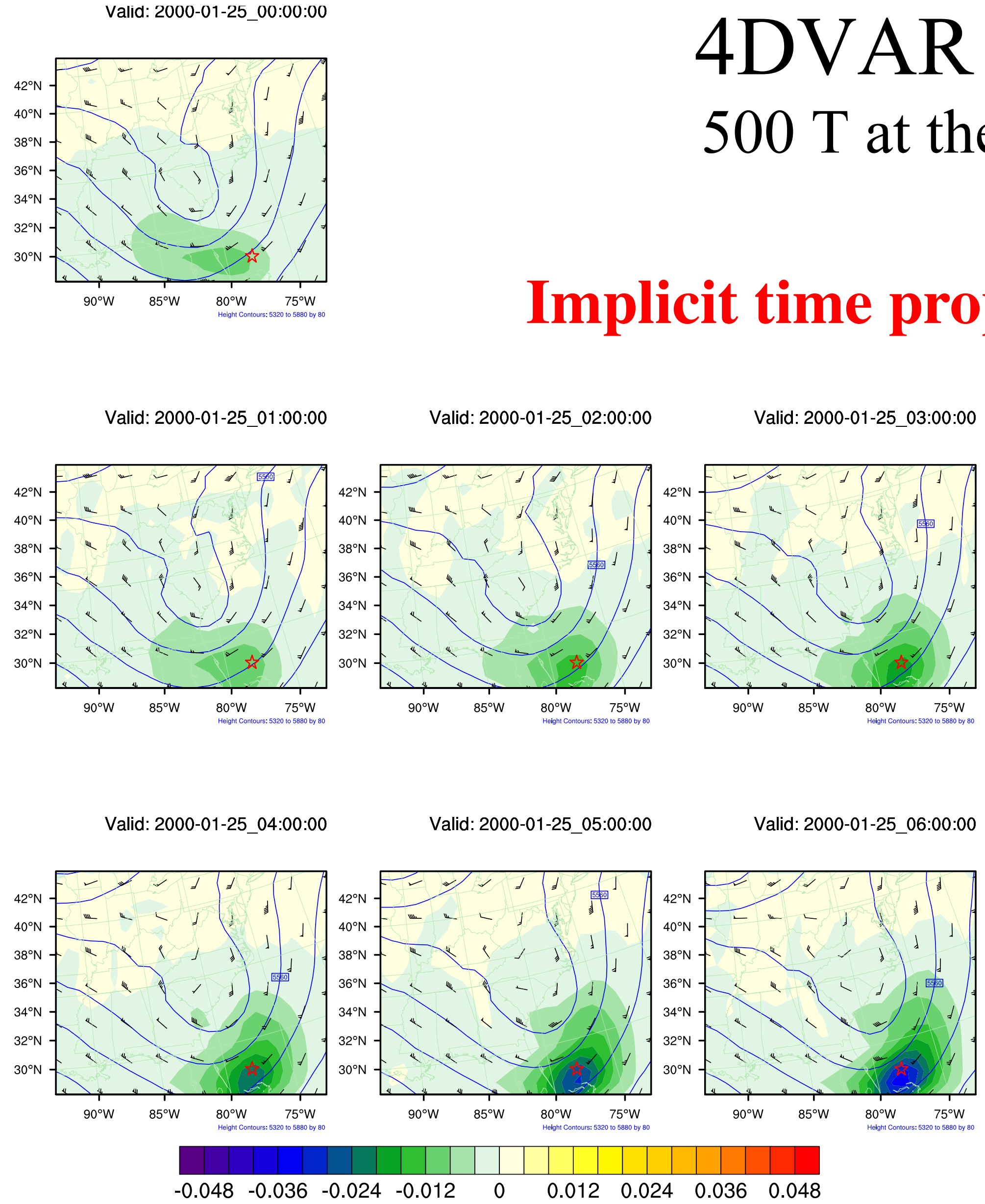
- Background error covariance (BEC) implicitly evolving within time window through linearized model, though B (BEC at the beginning of time window) typically the same for each analysis cycle. BEC at time t_i,

$$\mathbf{B_i = M_i B M_i^T}$$

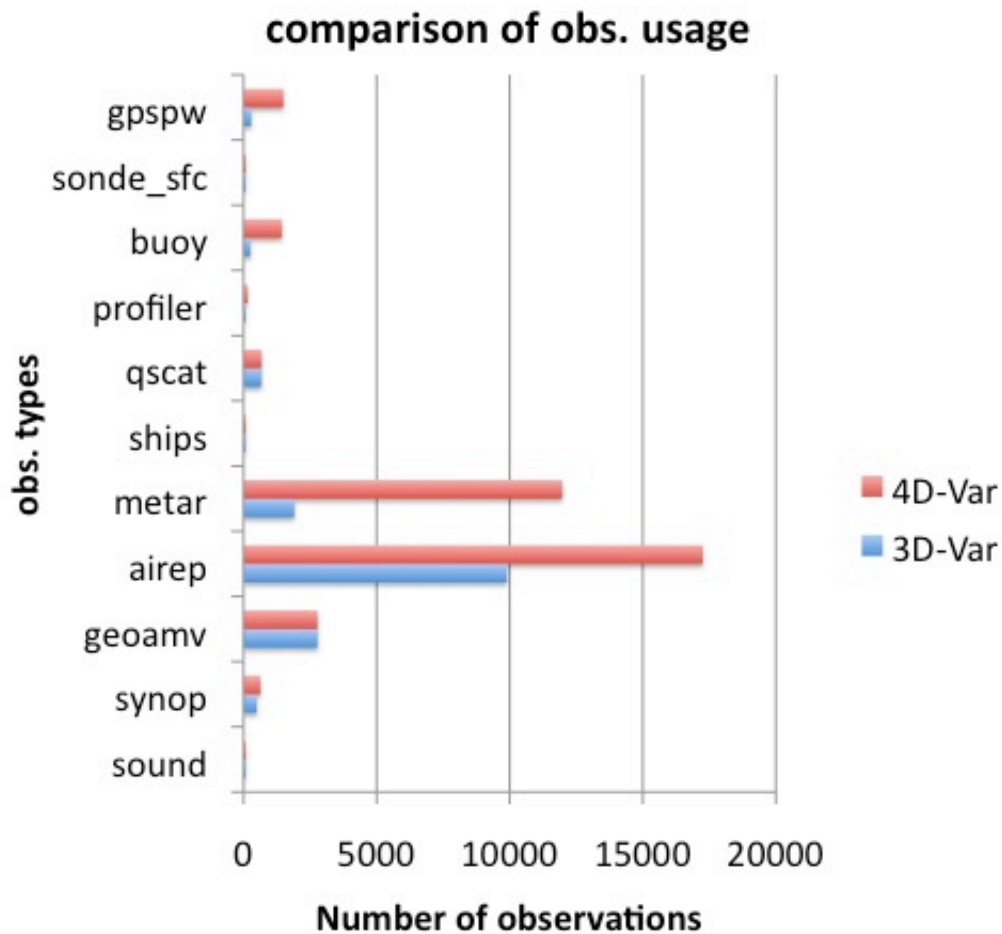
4DVAR Single Obs Test

500 T at the end of time window

Implicit time propagation of B matrix



Number of obs assimilated: 3DVAR vs. 4DVAR



Outline

- Incremental 4DVAR
- Multi-Resolution Incremental 4DVAR
- Introduction to 4DVAR practice

Multi-Resolution Incremental 4DVAR (MRI-4DVAR, under development)

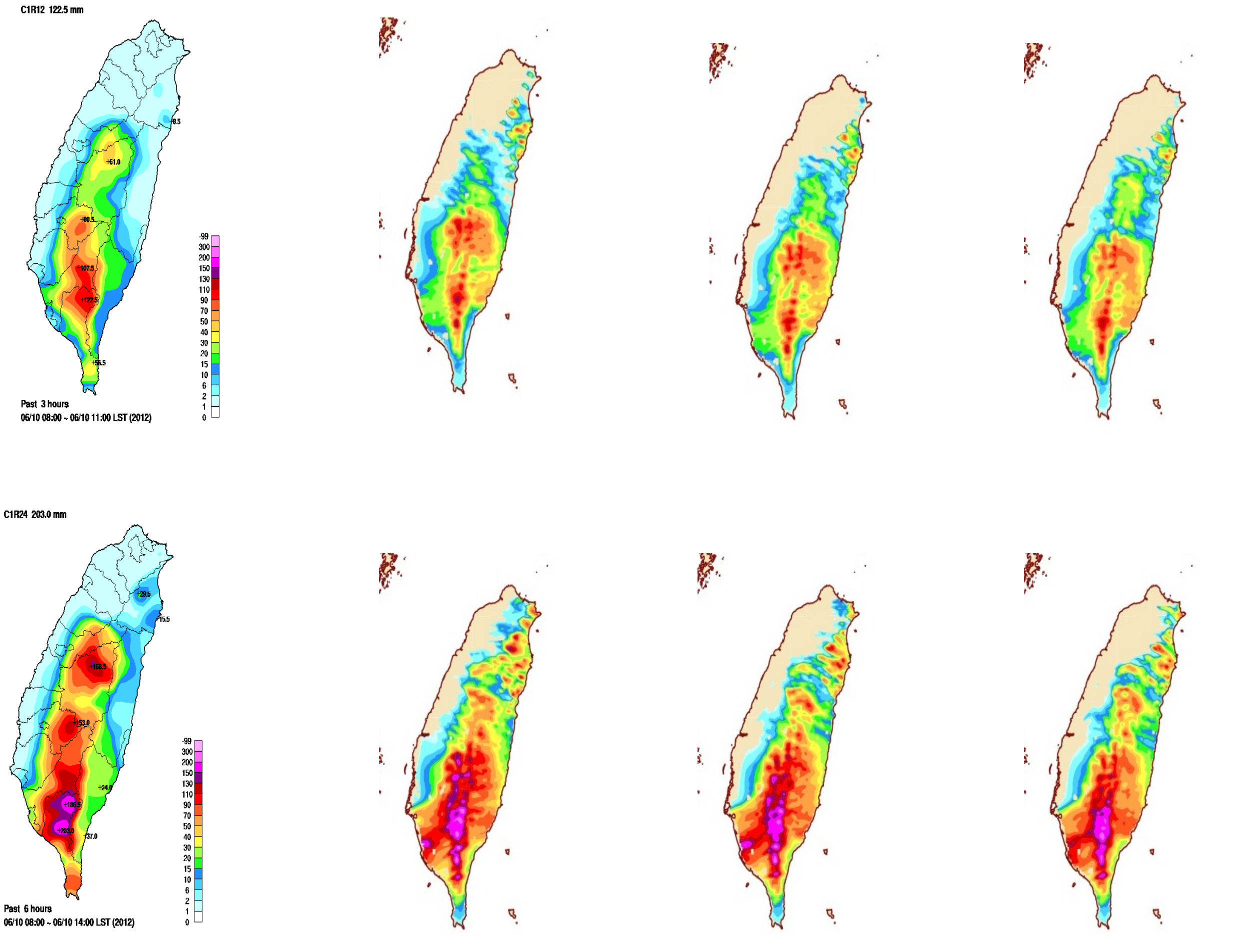
$$J(\mathbf{v}) = \frac{1}{2}(\mathbf{v}-\mathbf{v}^g)^{\mathrm{T}}(\mathbf{v}-\mathbf{v}^g) + \frac{1}{2}\sum_{i=1}^{N}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v}-\mathbf{d}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathbf{H}_i\mathbf{M}_i\mathbf{U}\mathbf{v}-\mathbf{d}_i)$$

- OmB (i.e., d_i) calculation uses high-resolution model trajectory at each outer loop

- 4DVAR minimization (need TL/AD integration for each iteration) runs at lower resolutions to allow substantial speed-up
  - Minimization resolution can be different for different outer loops, i.e., 9km for the 1st loop, 3km for the 2nd loop.

MRI-4DVAR test: Taiwan Rainfall forecast

	OBS rainfall	2km/2km 4DVAR	6km/6km MRI-4DVAR	18km/6km MRI-4DVAR
3-hr acc. Rainfall				
6-hr acc. rainfall				



Computing time: 2012/06/10 case 20 min time window

Experiment	Outer loop/ iteration	CPU	Time
2km2km_vp_sobs	25,25	32	36 hrs
6km6km_vp_sobs	25,25	32	90 mins
18km6km_vp_sobs	25,25	32	50 mins

3-stage MRI-3D/4DVAR: 18km/6km

- Loop1/Stage1: run WRFDA in "**Observer**" mode at full model resolution 2km
  - Run WRF non-linear model at 2km
  - then compute OMB and do QC at different time slots
  - Write out OMB at different time slots (e.g., gts_omb.01.synop, gts_omb.02.synop, …)

- Loop1/Stage2: run WRFDA in "**Minimization**" mode at 18km
  - Thin (not interpolation) 2km fg to 18km
  - Run non-linear WRF integration at 18km using 18km fg/wrfbdy to generate 18km model trajectory, which is used as the base state of WRF TL/AD integration.
  - Read in OMB output from "Observer" step
  - Run minimization

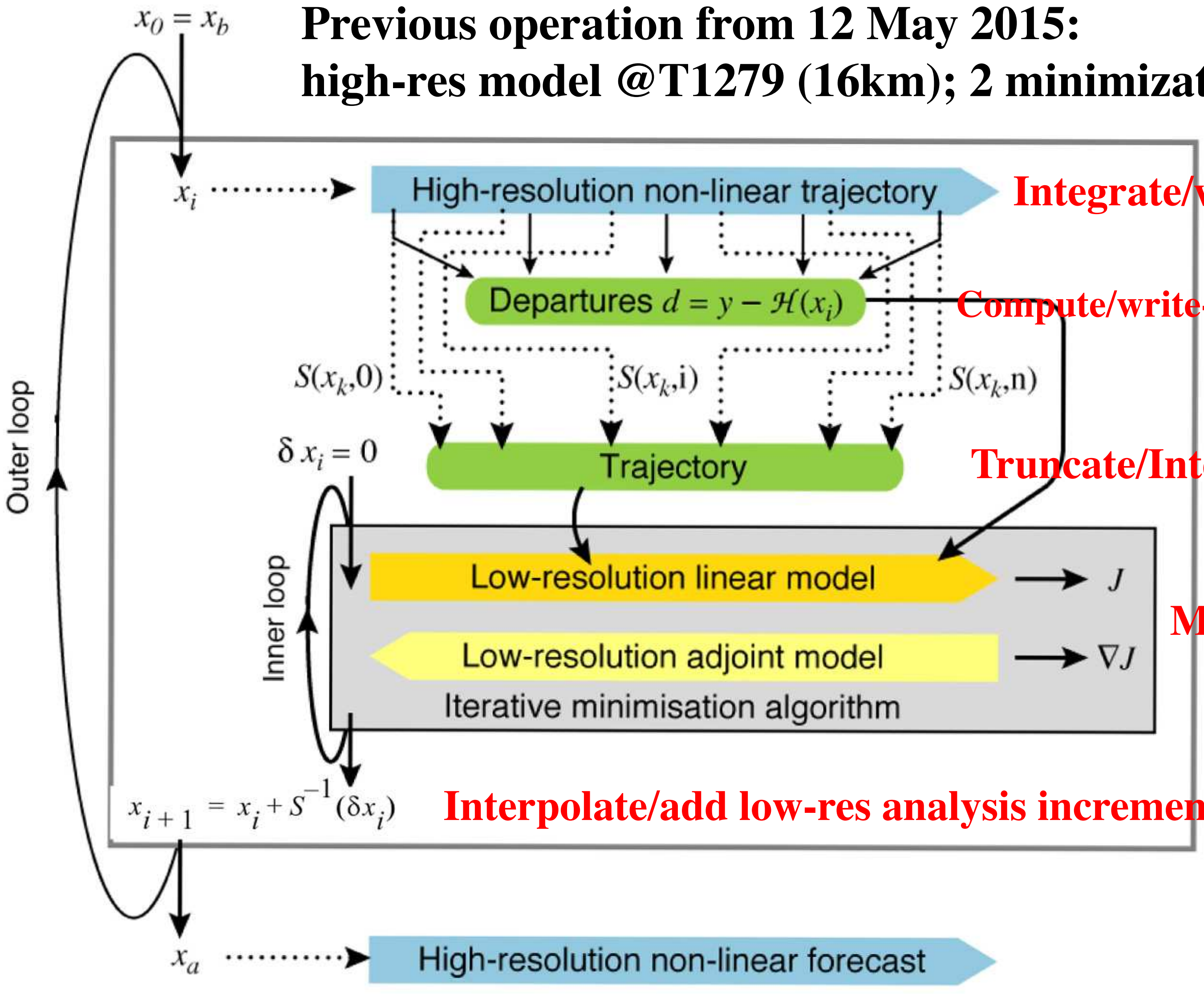
3-stage MRI-3D/4DVAR: 18km/6km

- Stage-3: run **"Regriding"** outside WRFDA
  - wrfvar_output@2km = fg@2km + S (wrfvar_output@18km – fg@18km)
  - vp@6km = S vp@18km
  - S is interpolation operator

- Then go to for the 2nd outer loop
  - Run WRFDA in Observer mode at 2km
  - Run WRFDA in minimization mode at 6km
  - Run regridding

How ECMWF does?

Previous operation from 12 May 2015:
high-res model @T1279 (16km); 2 minimization @T159 & T255



Integrate/write-out high-res traj. @T1279

Compute/write-out OMB with high-res traj. also do QC

Truncate/Interpolate high-res traj. to low-res @T159 or T255

Minimization @T159/T255

Interpolate/add low-res analysis increment to high-res first guess

Some word about WRFDA-3DVAR/4DVAR for WRF/Chem

- Under development for aerosol/chemistry data assimilation

- Including WRFPlus-Chem for GOCART
  - J. J. Guerrette and D. K. Henze, 2015, GMD
  - J. J. Guerrette and D. K. Henze, 2017, ACP

- Will be very useful for air-quality forecast and source emission inversion.

Outline

- Incremental 4DVAR
- Multi-Resolution Incremental 4DVAR
- Introduction to 4DVAR practice

Compile WRFDA in 4DVAR mode

- Download WRFPlus code
  - Include non-linear and TL/AD code of WRF
- Download WRFDA code
- Install WRFPLUS V3.9
  - ./configure (-d) wrfplus
    ./compile wrf (only compile wrf.exe)
  - wrf.exe should be generated under the WRFPLUSV3/main directory.
- for csh, tcsh : setenv WRFPLUS_DIR path of wrfplusv3
  for bash, ksh : export WRFPLUS_DIR=path of wrfplusv3
- Install WRFDA V3.9
  - ./configure (-d) 4dvar
    ./compile all_wrfvar
    da_wrfvar.exe should be generated in the var/build directory.

Notes about WRFPlus

- WRFPLUS only works with regional ARW core, not for NMM core or global WRF.

- WRFPLUS only works with single domain, not for nested domains.

- WRFPLUS can not work with Adaptive Time Stepping options.
- WRFPLUS TL/AD code only has 3 simplified physics processes:
  – surface drag (bl_pbl_physics=98);
  – large scale condensate or Kessler (mp_physics=98 or 99)
  – a simplified cumulus scheme (cu_physics=98)

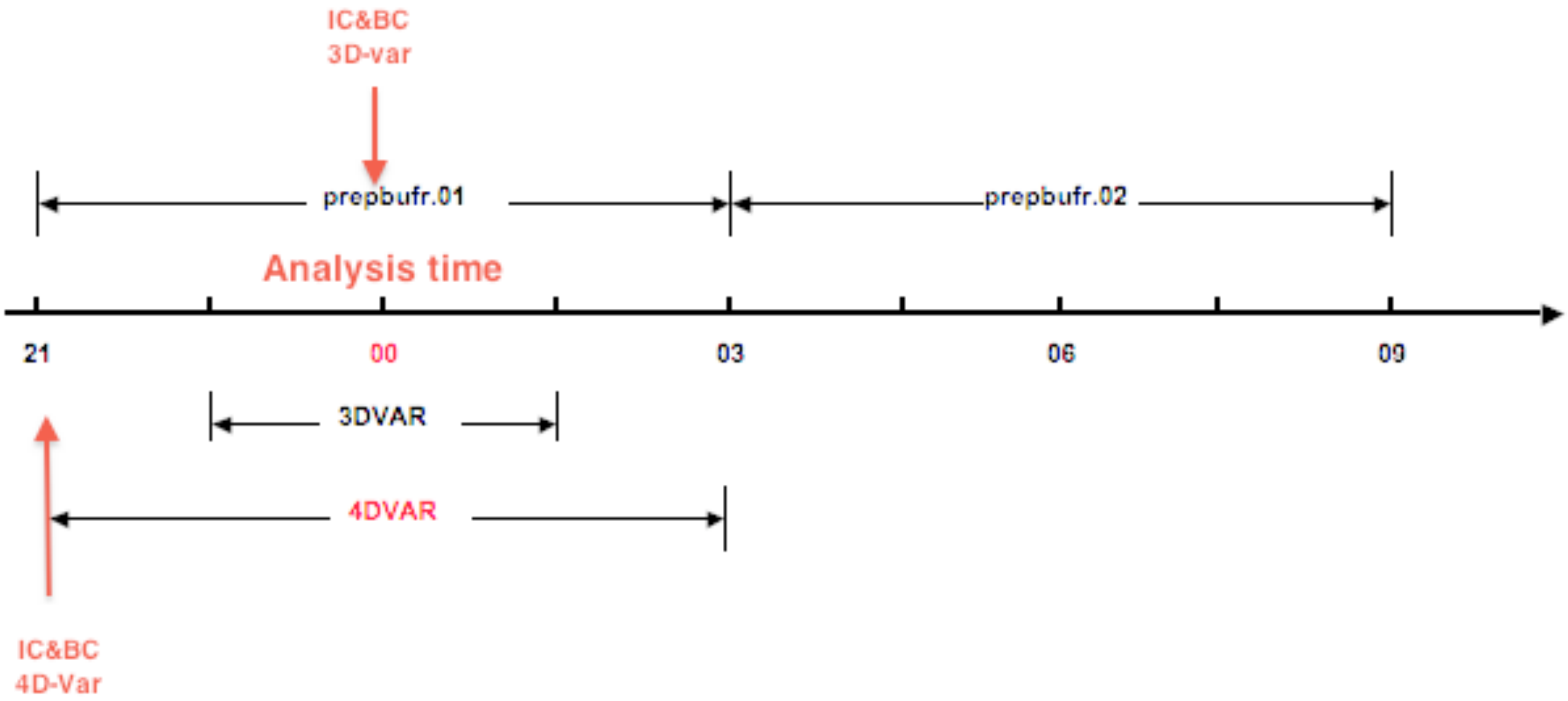
Prepare obs for 4DVAR

- Conventional observations
  - LITTLE_R format
  - NCEP PREPBUFR format

- Satellite radiance BUFR data

- ASCII format precipitation and radar data

4DVAR time window



Run a 4DVAR test case

- enter WRFDA/var/test/4dvar (or working directory of your choice)
- get the test dataset
- ln -fs wrfinput_d01 fg
- ln -fs wrfinput_d01 .
- ln –fs wrfbdy_d01 .
- ln -fs ../../build/da_wrfvar.exe .
- ln -fs ../../run/be.dat.cv3 be.dat
- ./da_wrfvar.exe
- Typically you should run in parallel with MPI (mpirun -np # da_wrfvar.exe) or your system's custom run command (on Yellowstone: bsub))

Run a 4DVAR test case

- WRFPlus/WRFDA compiled in double precision
- So link double-precision version of following files for 4DVAR run
  - ln -sf ${WRF_DIR}/run/RRTM_DATA_DBL RRTM_DATA
  - ln -sf ${WRF_DIR}/run/RRTMG_LW_DATA_DBL RRTMG_LW_DATA
  - ln -sf ${WRF_DIR}/run/RRTMG_SW_DATA_DBL RRTMG_SW_DATA
- And other WRF related files
  - ln -sf ${WRF_DIR}/run/SOILPARM.TBL .
  - ln -sf ${WRF_DIR}/run/VEGPARM.TBL .
  - ln -sf ${WRF_DIR}/run/GENPARM.TBL .
  - ln -sf ${WRF_DIR}/run/LANDUSE.TBL .

Important namelist variables

- &wrfvar1
  - **var4d**: logical, set to .true. to use 4D-Var
  - **var4d_lbc**: logical, set to .true. to include lateral boundary condition control in 4D-Var
  - **var4d_bin**: integer, seconds, length of sub-window to group observations in 4D-Var
- &wrfvar18,21,22
  - **analysis_date** : the start time of the assimilation window
  - **time_window_min** : the start time of the assimilation window
  - **time_window_max** : the end time of the assimilation window
- &perturbation
  - **jcdfi_use**: logical, if turn on the digital filter as a weak constraint.
  - **jcdfi_diag**: integer, 0/1, Jc term diagnostics
  - **jcdfi_penalty**: real, weight to jcdfi term

Important namelist variables

- &physics
  - all physics options must be consistent with those used in wrfinput
    - Non-linear WRF run can use different physics options from TL/AD
  - **mp_physics_ad** =
    98: large-scale condensation microphysics (default)
    99: modified Kessler scheme (new in V3.7)
  - **bl pbl physics** = any : but only surface drag available for TL/AD
  - **cu physics** = any : but only simplified cumulus scheme for TL/AD

- &time control
  - **run_xxxx** : be consistent with the length of the time window
  - **start_xxxx** : be consistent with the start time of the time window
  - **end_xxxx** : be consistent with the end time of the time window

WRFDA adjoint check before 4DVAR run

- &wrfvar10
  - test_transforms=true,
- run da wrfvar.exe

Check results

```
...
 wrf: back from adjoint integrate
 d01 2008-02-05_21:00:00 read nonlinear xtraj time stamp:2008-02-05_21:00:00
 Single Domain < y, y     > =   2.1543550677243 3E+06
 Single Domain < x, x_adj > =   2.15435506772431E+06

 Whole Domain < y, y     > =   2.15435506772433E+06
 Whole Domain < x, x_adj > =   2.15435506772431E+06

da_check_xtoy_adjoint: Test Finished:

 *** WRF-Var check completed successfully ***
```