Ensemble Data Assimilation in MPAS-JEDI: EDA and LETKF

Presented by Jake Liu

Based on materials prepared by Tao Sun

Prediction, Assimilation, and Risk Communication Section

Mesoscale & Microscale Meteorology Laboratory

National Center for Atmospheric Research

NCAR

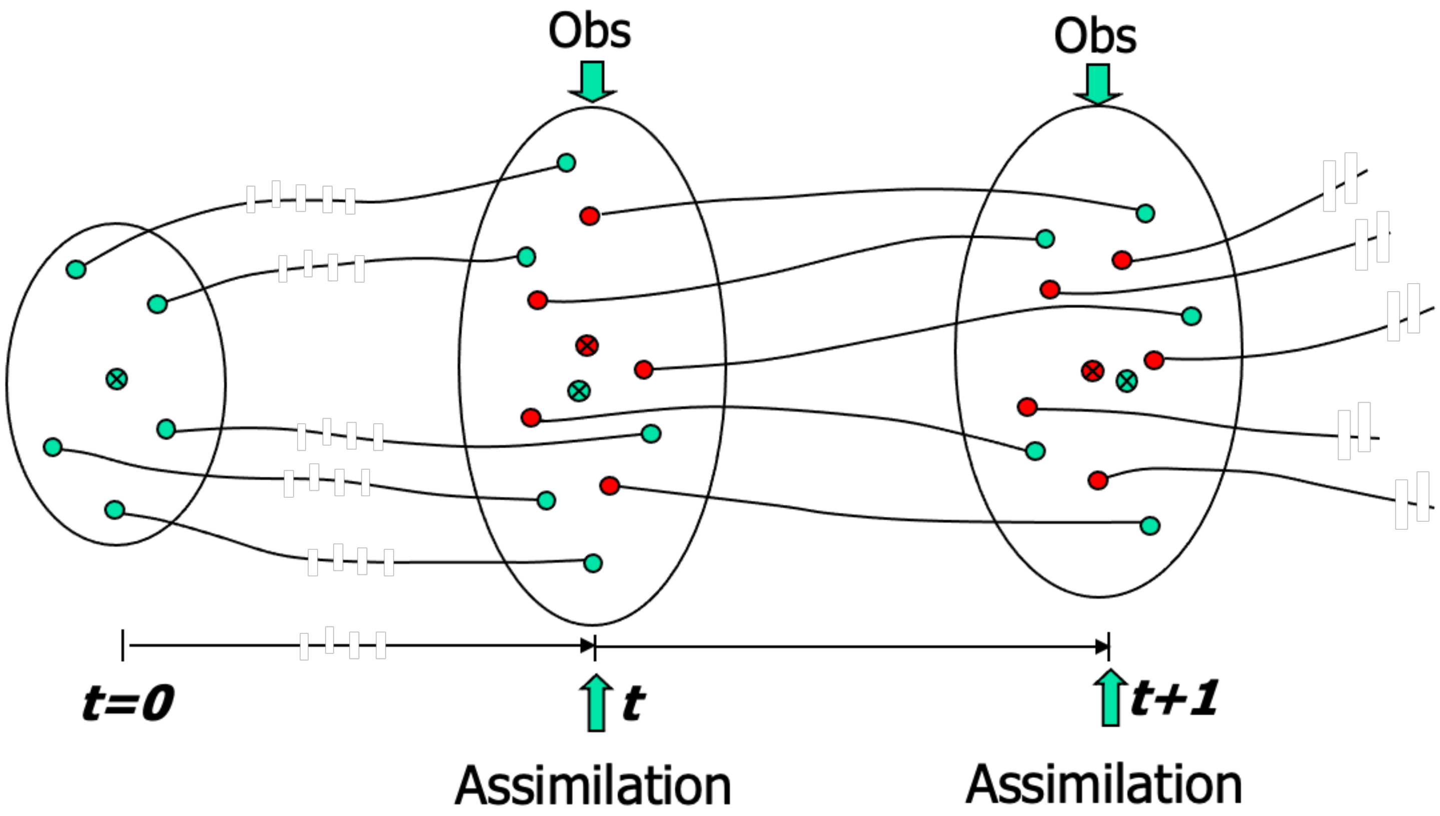
MPAS-JEDI Tutorial, INPE, 15-16 August, 2024

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

Outline

- Ensemble data assimilation methods
- Ensemble of data assimilation (EDA)
- Local ensemble transform Kalman filter (LETKF)
- Comparison between EDA and LETKF

Ensemble data assimilation methods



Flow-chart of ensemble data assimilation

Benefits of ensemble DA:

- ❖ Provide uncertainty estimate of forecast and analysis;
- ❖ Provide flow-dependent background error covariance (BEC) for deterministic DA methods.

Ensemble data assimilation methods

Two methods of ensemble data assimilation available for MPAS-JEDI:

❖ The ensemble of data assimilation (**EDA**) method.

❖ , Local Ensemble Transform Kalman Filter (LETKF) and gain form of LETKF

Outline

- Ensemble data assimilation methods
- Ensemble of data assimilation (EDA)
- Local ensemble transform Kalman filter (LETKF)
- Comparison between EDA and LETKF

Ensemble of Data Assimilation (EDA)

In EDA, the ensemble analysis are obtained by solving *N* independent variational cost functions with perturbed observations, where the *i*th EDA cost function is:

$$J(\mathbf{x}_i) = \frac{1}{2}\left(\mathbf{x}_i - \mathbf{x}_i^b\right)^{\mathrm{T}}\mathbf{B}_i^{-1}\left(\mathbf{x}_i - \mathbf{x}_i^b\right) + \frac{1}{2}[H(\mathbf{x}_i) - \mathbf{y}^o - \boldsymbol{\epsilon}_i]^{\mathrm{T}}\mathbf{R}^{-1}[H(\mathbf{x}_i) - \mathbf{y}^o - \boldsymbol{\epsilon}_i]$$

$\mathbf{x}_i^b$: *i*th background states;

$\mathbf{x}_i$: *i*th analysis states;

$\mathbf{B}_i$: *i*th background error covariance matrix;

$\mathbf{R}$: observation error covariance matrix;

$\mathbf{y}^o$: observation states;

$\boldsymbol{\epsilon}_i$: *i*th random observation errors; $\boldsymbol{\epsilon} \sim N(0, \mathbf{R})$, and $\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\epsilon}_i = 0$.

Guerrette et al. 2023

Configure and Run EDA

For each EDA member, a specific yaml file is needed, i.e., *3denvar_${memebr}.yaml*.

❖ Each EDA member can be done with a single command:
$mpirun ./mpasjedi_variational.x 3denvar_${memebr}.yaml

Configure and Run EDA

The configuration of each EDA assimilation member is very similar to the common variational DA, but some parameters need setting.

❖ Introduction of observation random errors

Set "obs perturbations" to true in the observations section:

```
observations:
   obs perturbations: true
```

For each observation type, the observation error should be

```
obs error:
   covariance model: diagonal
   zero-mean perturbations: true
   member: 1 # index of EDA member
   number of members: 20 # ensemble size
```

Configure and Run EDA

The configuration of each EDA assimilation member is very similar to the common variational DA, but some parameters need setting.

❖ Self-exclusion in ensemble BEC

```
members from template:
    template:
      <<: *memberConfig
      filename: ../../bg/mem%iMember%/bg.2018-04-15_00.00.00.nc
    pattern: %iMember%
    start: 1
    zero padding: 3
    nmembers: 19 #Number of EDA member -1
    except: [1] # Index of EDA member
```

Configure and Run EDA

Posterior Inflation: Relaxation To Prior Perturbation (RTPP)

After all EDA members are updated, a posterior inflation is needed to keep the ensemble spread using an external executable **mpasjedi_rtpp.x**

$mpirun ./mpasjedi_rtpp.x rtpp.yaml

```
_state read: &stateReadConfig
  date: 2018-04-15T00:00:00Z
  state variables: [spechum,surface_pressure,temperature,uReconstructMeridional,uReconstructZonal,pressure_p,pressure,rho,theta,u,qv]
  stream name: background
output:
  filename: ${OUTPUT_DIR}/mem%{member}%/an.$Y-$M-$D_$h.$m.$s.nc
  stream name: analysis        Output directory. Should be the same as the analysis member.
geometry:
  nml_file: namelist.atmosphere
  streams_file: streams.atmosphere
  deallocate non-da fields: true                  List of variables to be inflated
analysis variables: [spechum,surface_pressure,temperature,uReconstructMeridional,uReconstructZonal,pressure_p,pressure,rho,theta,u,qv]
background:
  members:
  - <<: *stateReadConfig
    filename: ${BAKDIR}/mem001/bg.2018-04-15_00.00.00.nc List of all background members
  ...
analysis:
  members:
  - <<: *stateReadConfig
    filename: ${ANA_DIR}/mem001/an.2018-04-15_00.00.00.nc List of all analysis members
  ...
factor: 0.8     RTPP inflation factor
```

Outline

- Ensemble data assimilation methods
- Ensemble of data assimilation (EDA)
- Local ensemble transform Kalman filter (LETKF)
- Comparison between EDA and LETKF

Local Ensemble Transform Kalman Filter (LETKF)

Background error covariance $\mathbf{P}^{b} = \dfrac{1}{N-1}\mathbf{X}^{b}(\mathbf{X}^{b})^{\mathrm{T}}$

$$\mathbf{x}_i^{b} = \bar{\mathbf{x}}^{b} + \mathbf{X}_i^{b}$$

Analysis error covariance $\mathbf{P}^{a} = \dfrac{1}{N-1}\mathbf{X}^{a}(\mathbf{X}^{a})^{\mathrm{T}} = \mathbf{X}^{b}\widetilde{\mathbf{P}}^{a}(\mathbf{X}^{b})^{\mathrm{T}}$

$$\widetilde{\mathbf{P}}^{a} = [(N-1)\mathbf{I}/\rho + (\mathbf{Y}^{b})^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{Y}^{b})]^{-1}$$ **transform matrix**

$$\mathbf{Y}^{b} = \mathbf{H}\left(\mathbf{X}^{b}\right) \approx H(\mathbf{x}^{b}) - \bar{\mathbf{y}}^{b}$$ **obs-space ens. perturbation**

Ensemble mean updating $\bar{\mathbf{x}}^{a} = \bar{\mathbf{x}}^{b} + \mathbf{X}^{b}\widetilde{\mathbf{P}}^{a}(\mathbf{Y}^{b})^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{y}^{o} - \bar{\mathbf{y}}^{b}) = \bar{\mathbf{x}}^{b} + \mathbf{X}^{b}\overline{\mathbf{w}}^{a}$

Ens. Perturbation updating $\mathbf{X}^{a} = \mathbf{X}^{b}[(N-1)\widetilde{\mathbf{P}}^{a}]^{\frac{1}{2}} = \mathbf{X}^{b}\mathbf{W}^{a}$ **weighting vector**

$$\mathbf{W}^{a} = \mathbf{U}\mathbf{S}^{\frac{1}{2}}\mathbf{U}^{\mathrm{T}}$$ **Singular vector decomposition**

To update analysis states at every grid point, the LETKF assimilates only **local observations** within a certain distance from each grid point.

Hunt et al. 2007; Frolov 2023

Configure and Run LETKF

Increment variables, background, and output section:

```
increment variables: ${an_variables}

background:
  members from template:
    template:
      date: &analysisDate YYYY-MM-DDTHH:MN:SSZ
      state variables: [${state_variables}]
      stream name: background
      filename: ${bg_dir}/mem%iMember%/${bg_file}
    pattern: %iMember% # 001, 002, …, 020
    start: 1
    zero padding: 3
    nmembers: 20 # Number of ensemble

output:
  filename: ${an_dir}/mem%{member}%/${an_file}
  stream name: analysis
```

MPAS-JEDI will overwrite analysis variables in ${an_file}, so we need to copy ${bg_file} to ${an_file} before running LETKF.

Configure and Run LETKF

Observation space localization:

The Observation section in JEDI are similar to that in variational DA except for the observation space localization configurations.

Horizontal localization

obs localizations:

localization method: ***Horizontal Gaspari-Cohn***/ *SOAR/ Box car*

lengthscale: ***${horizontal localization scale}***

search method: ***kd_tree***/ *brute_force*

distance type: ***geodesic***/ *cartesian*

max nobs: *maximum obs umber for localization*

…

Vertical localization

obs localization:

localization method: ***Vertical localization***

vertical lengthscale: ***${vertical localization scale}***

apply log transformation: *commonly used for pressure*

ioda vertical coordinate: *height/pressure/…*

ioda vertical coordinate group: *MetaData*

localization function: *Box Car/ Gaspari Cohn/ SOAR*

…

Configure and Run LETKF

Local ensemble DA section:

This section relates to the local ensemble DA methods and the variance inflation schemes.

local ensemble DA:

solver: LETKF/ GETKF

Variance inflation

mult: prior multiplicative inflation

rtpp: post relaxation to prior perturbation

rtps: post relaxation to prior spread

LETKF: *Vertical localization is done in the observation space;*

GETKF: *using modulated ensembles to emulate model-space vertical localization.*

$$\mathbf{P}^{b'} = \alpha \mathbf{P}^{b}$$

$$\mathbf{X}_i^{a'} = \alpha \mathbf{X}_i^{a} + (1-\alpha)\mathbf{X}_i^{b}$$

$$\mathbf{X}_i^{a'} = \mathbf{X}_i^{a}(1 + \alpha \frac{\sigma_b - \sigma_a}{\sigma_a})$$

Configure and Run LETKF

LETKF analysis procedure can be divided into three steps: **Observer**, **Solver**, **and DiagOMA**

Observer

driver:
    run as observer only: true
    update obs config with geometry info: false

This step will only calculate the HofX of all members and ensemble mean and then write them out;

Quality control of LETKF is done based on the ensemble mean states.

In this step, the *observation distribution* can be set to *RoundRobin* to be more efficient.

Solver

driver:
    read HX from disk: true
    do posterior observer: false
    save posterior ensemble: true
    save posterior mean: true

In this step, if "*read HX from disk*" is set to true, it will read the HofX of all members and ensemble mean from the **Observer** step, and then run LETKF solver;

The *obsdatain* should be changed to the *obsdataout* that is used in Observer step.

The *observation distribution* should be set to *Halo.*

Configure and Run LETKF

LETKF analysis procedure can be divided into three steps: **Observer**, **Solver**, and **DiagOMA**

Observer

```
background:
  members from template:
      template: <<: *memberConfig
      filename: ../../bg/mem%iMember%/bg.2018-04-15_00.00.00.nc
    pattern: %iMember%
    start: 1
    zero padding: 3
    nmembers: 20 _obs

_obsdatain &ObsDataIn
    engine:
        type: H5File
        obsfile: ../../dbIn/sfc_obs_2018041500.h5
_obsdataout: &ObsDataOut
     engine:
         type: H5File
         obsfile: ../../dbOut/obsout_da_sfc.h5
```

DiagOMA

```
background:
  members from template:
      template: <<: *memberConfig
      filename: ../../an/mem%iMember%/an.2018-04-15_00.00.00.nc
    pattern: %iMember%
    start: 1
    zero padding: 3
    nmembers: 20

_obsdatain &ObsDataIn
    engine:
        type: H5File
        obsfile: ../../dbIn/sfc_obs_2018041500.h5
_obsdataout: &ObsDataOut
     engine:
         type: H5File
         obsfile: ../../dbAna/obsout_da_sfc.h5
```

Outline

- Ensemble data assimilation methods
- Ensemble of data assimilation (EDA)
- Local ensemble transform Kalman filter (LETKF)
- Comparison between EDA and LETKF

Comparison between EDA and KETKF

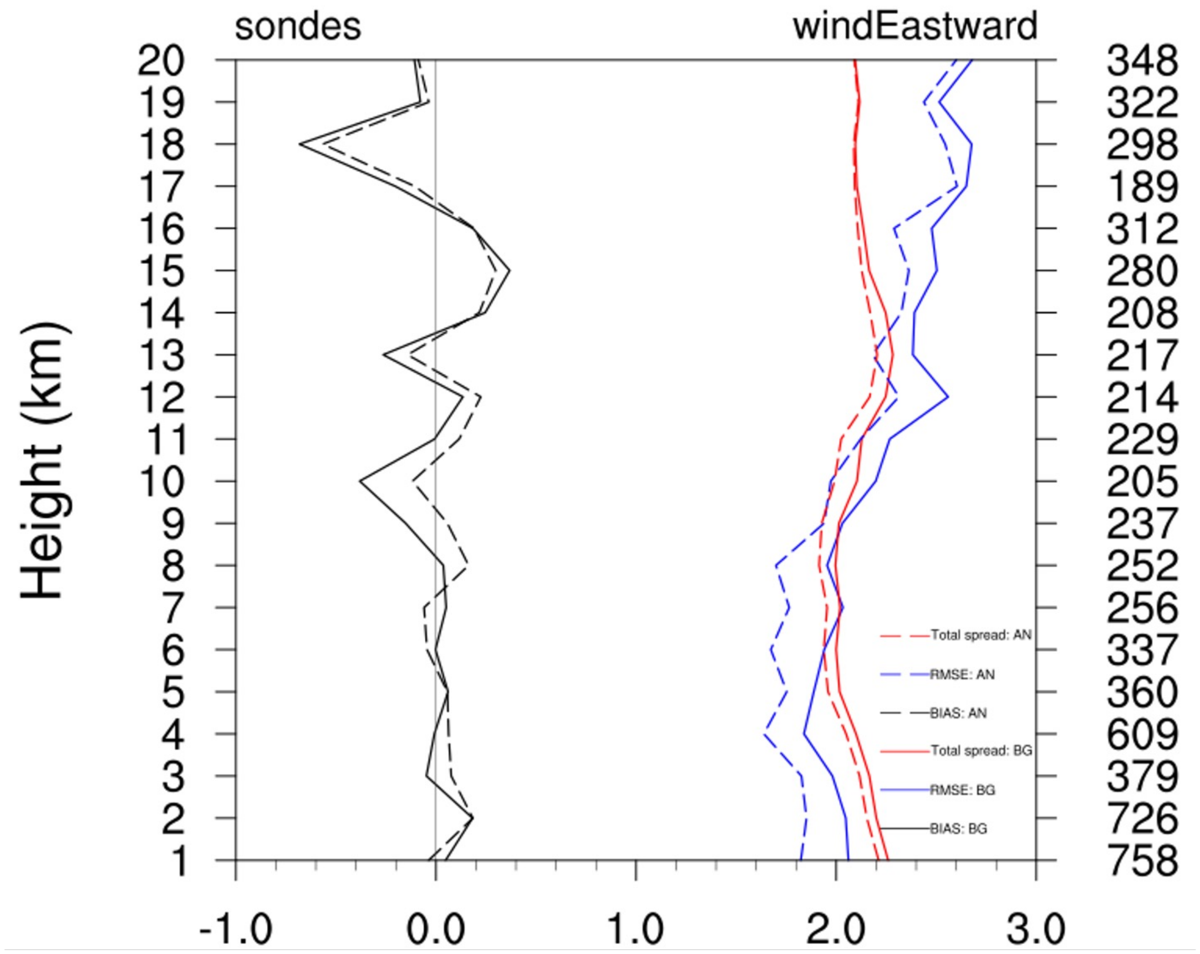
General comparison

DA methods	EDA	LETKF
Algorithm	Based on the variational framework	An EnKF-based method
Uncertainty	use perturbated observations	No need to perturb observations
Localization	Model space localization	Observation space localization
Inflation	External posterior inflation	Interior prior/posterior inflation
Computational cost	All members can be updated in parallel	All members are updated simultaneously

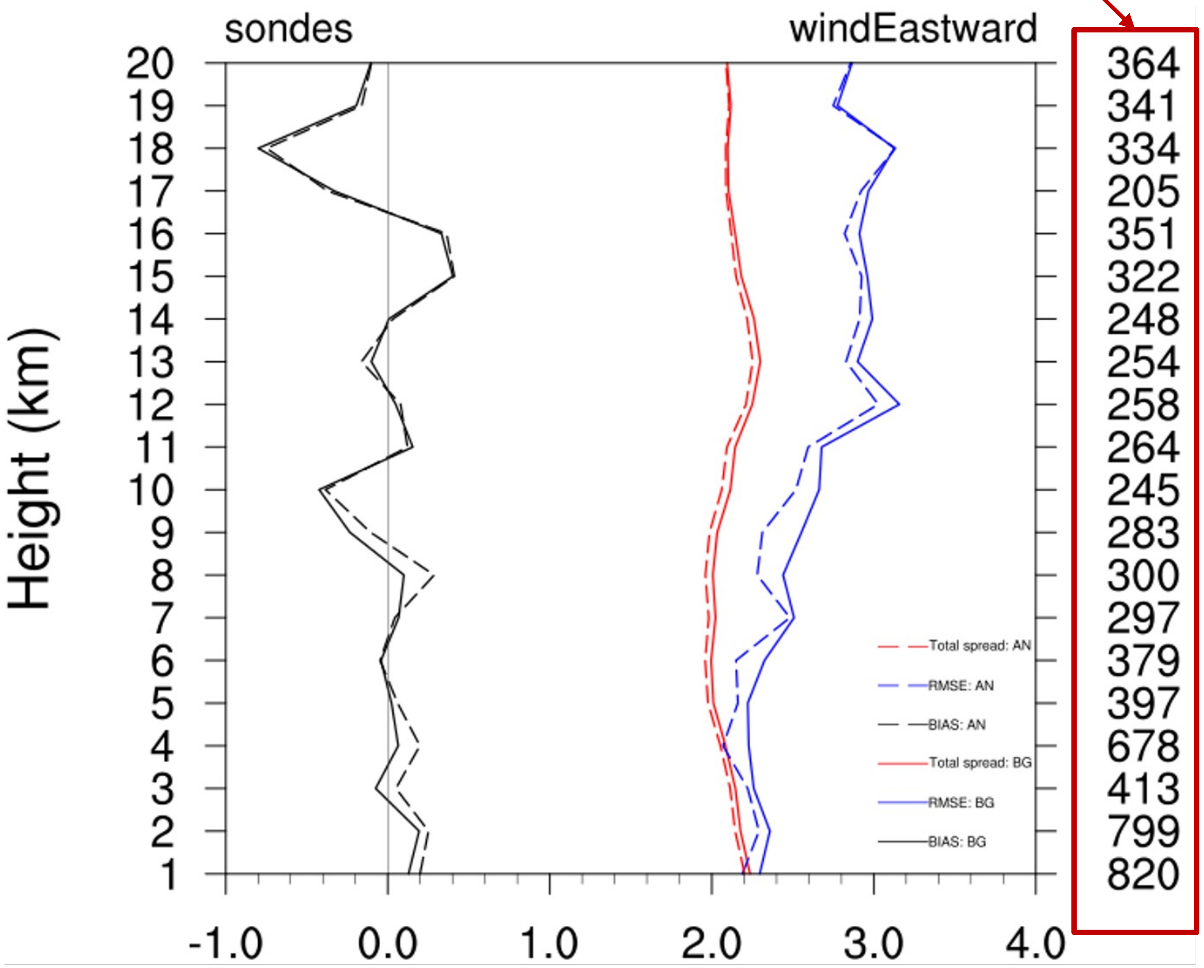
Comparison between EDA and KETKF

Vertical profiles of total spread, RMSE and BIAS

More observations are assimilated in LETKF



EDA



LETKF

References

- Guerrette, J. J., et al., 2023: Data assimilation for the Model for Prediction Across Scales – Atmosphere with the Joint Effort for Data assimilation Integration (JEDI-MPAS 2.0.0-beta): ensemble of 3D ensemble-variational (En-3DEnVar) assimilations, Geosci. Model Dev.
- Hunt B.R., Kostelich E.J., Szunyogh I., 2007: Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter, Physica D: Nonlinear Phenomena, 230, 112-126, https://doi.org/10.1016/j.physd.2006.11.008.
- Sergey Frolov, Anna Shlyaeva, Wei Huang, et al., 2023: Local volume solvers for Earth system data assimilation: implementation in the framework for Joint Effort for Data Assimilation Integration, Journal of Advances in Modeling Earth Systems, under review.
- https://jointcenterforsatellitedataassimilation-jedi-docs.readthedocs-hosted.com/en/latest/inside/jedi-components/oops/applications/localensembleda.html