



**NCAR**  
OPERATED BY UCAR

AUGUST 14, 2026

# Algorithms (2): Static B, 3DVar, and Hybrid-EnVar

Presented by BJ Jung

*MPAS-A and MPAS-JEDI Tutorial, Boulder CO, AUGUST 10-14, 2026*

# Overview

1. 3DVar and static B
2. B designed and how it operates
3. B training and tuning
4. Yaml configuration for 3DVar
5. Yaml configuration for Hybrid-3DEnVar

# 3DVar and Static Background Error Covariance

$$J(\mathbf{x}) = \underbrace{\frac{1}{2}(\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}_b)}_{J_b} + \underbrace{\frac{1}{2}(\mathbf{h}(\mathbf{x}) - \mathbf{y})^T \mathbf{R}^{-1}(\mathbf{h}(\mathbf{x}) - \mathbf{y})}_{J_o}$$

- 3DVar uses  $\mathbf{B} = \mathbf{B}_s$ .
- $\mathbf{B}_s$  has a **climatological** characteristics.
- 3DVar is computationally efficient.
- 3DVar is simple in terms of workflow.
- The hybrid B can be benefit from  $\mathbf{B}_s$ .

# B designed and how it operates (1/6)

$$\mathbf{B} = \langle \mathbf{x}_b - \mathbf{x}_t, \mathbf{x}_b - \mathbf{x}_t \rangle$$

- Role of  $\mathbf{B}$ 
  - weights the importance of  $\mathbf{x}_b$  for a given  $y$  and  $\mathbf{R}$ .
  - spreads the observed information in the vertical and horizontal direction.
  - spreads the observed information to other variables and imposes the balance properties.
- It is difficult to know  $\mathbf{B}$  exactly.
  - We don't know the "true" state,  $\mathbf{x}_t$ .
  - The dimension of  $\mathbf{B}$  is too large.
- Thus,  $\mathbf{B}$  is modeled in a practical way.

*See Bannister (2008a,b) for more detailed review on  $\mathbf{B}$ .*

# B designed and how it operates (2/6)

- MPAS-JEDI's  $\mathbf{B}_s$  is designed following that of the GSI\* (Wu et al., 2002), except the univariate spatial correlation.

$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T \mathbf{K}_2^T \mathbf{K}_1^T$$

- It is constructed as a series of linear variable changes ( $\mathbf{K}_1$ ,  $\mathbf{K}_2$  and their adjoint;  $^T$ ) around the univariate covariance ( $\mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T$ ).
- Note that...
  - $\mathbf{B}$  operates to the MPAS-JEDI's analysis variable ( $\delta \mathbf{x}$ ), a set of  $\{\delta u, \delta v, \delta T, \delta Q, \delta p_s\}$  @ cell center.
  - $\mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T$  is represented with  $\{\delta \psi, \delta \chi_u, \delta T_u, \delta Q, \delta p_{s,u}\}$ .

\*Gridpoint Statistical Interpolation

$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \Sigma \mathbf{C} \Sigma^T \mathbf{K}_2^T \mathbf{K}_1^T \quad (3/6)$$

- $\mathbf{K}_1$  is a linear variable change  
from stream function ( $\delta\psi$ ) and velocity potential ( $\delta\chi$ )  
to zonal ( $\delta u$ ) and meridional ( $\delta v$ ) winds

$$\begin{bmatrix} \delta u \\ \delta v \\ \delta T \\ \delta Q \\ \delta p_s \end{bmatrix} = \begin{bmatrix} -\partial_y & -\partial_x & 0 & 0 & 0 \\ \partial_x & -\partial_y & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \delta\psi \\ \delta\chi \\ \delta T \\ \delta Q \\ \delta p_s \end{bmatrix}$$

- $\mathbf{K}_1$  directly operates on the native MPAS mesh ( [code](#) ).
- $\mathbf{K}_1^T$  is a corresponding adjoint operator ( [code](#) ).

$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \Sigma \mathbf{C} \Sigma^T \mathbf{K}_2^T \mathbf{K}_1^T \quad (4/6)$$

- $\mathbf{K}_2$  applies the vertical cross-variable correlation via linear regression : through BUMP Vertical BALance (VBAL) operator

$$\begin{aligned} \bullet \quad \delta\chi &= \delta\chi_b + \delta\chi_u = \mathbf{L}\delta\psi + \delta\chi_u \\ \bullet \quad \delta T &= \delta T_b + \delta T_u = \mathbf{M}\delta\psi + \delta T_u \\ \bullet \quad \delta p_s &= \delta p_{s,b} + \delta p_{s,u} = \mathbf{N}\delta\psi + \delta\chi_u \end{aligned} \quad \begin{bmatrix} \delta\psi \\ \delta\chi \\ \delta T \\ \delta Q \\ \delta p_s \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{L} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{M} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{N} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \delta\psi \\ \delta\chi_u \\ \delta T_u \\ \delta Q \\ \delta p_{s,u} \end{bmatrix}$$

- $\delta\psi$  is a predictor for the *balanced* part of  $\delta\chi$ ,  $\delta T$ , and  $\delta p_s$ .
- Full matrix for  $\mathbf{M}$  &  $\mathbf{N}$ , diagonal matrix for  $\mathbf{L}$

Following Derber and Bouttier (1999)

- $\mathbf{K}_2^T$  is a corresponding adjoint operator.

$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T \mathbf{K}_2^T \mathbf{K}_1^T \quad (5/6)$$

- $\mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T$  represents the univariate covariance for  $\{\delta\psi, \delta\chi_u, \delta T_u, \delta Q, \delta p_{s,u}\}$ . These variables do not have a cross-variable covariance anymore.
- $\mathbf{\Sigma}$  (and  $\mathbf{\Sigma}^T$ ) is a diagonal matrix with error standard deviation  
: through either BUMP VARiance (VAR) operator or SABER StdDev operator
- $\mathbf{C}$  is a block diagonal matrix. Each blocks represents the spatial correlation for each variables for  $\{\delta\psi, \delta\chi_u, \delta T_u, \delta Q, \delta p_{s,u}\}$ .
  - through BUMP Normalized Interpolated Convolution on an Adaptive Subgrid (NICAS; Ménétrier, 2026) operator
  - **BUMP NICAS directly operates on the MPAS's unstructured mesh.**

$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \Sigma \mathbf{C} \Sigma^T \mathbf{K}_2^T \mathbf{K}_1^T \quad (6/6)$$

- Even with a single variable, the dimension for spatial correlation is still large.
- BUMP NICAS applies the spatial correlation with *a subset of full grid* ( $\mathbf{C}^s$ ), rather than with a full grid ( $\mathbf{C}$ ). The correlation on the subgrid is interpolated ( $\mathbf{S}$ ) to the full grid with a normalization factor ( $\mathbf{N}$ ).

$$\mathbf{C} \cong \tilde{\mathbf{C}} = \mathbf{N} \mathbf{S} \mathbf{C}^s \mathbf{S}^T \mathbf{N}^T$$

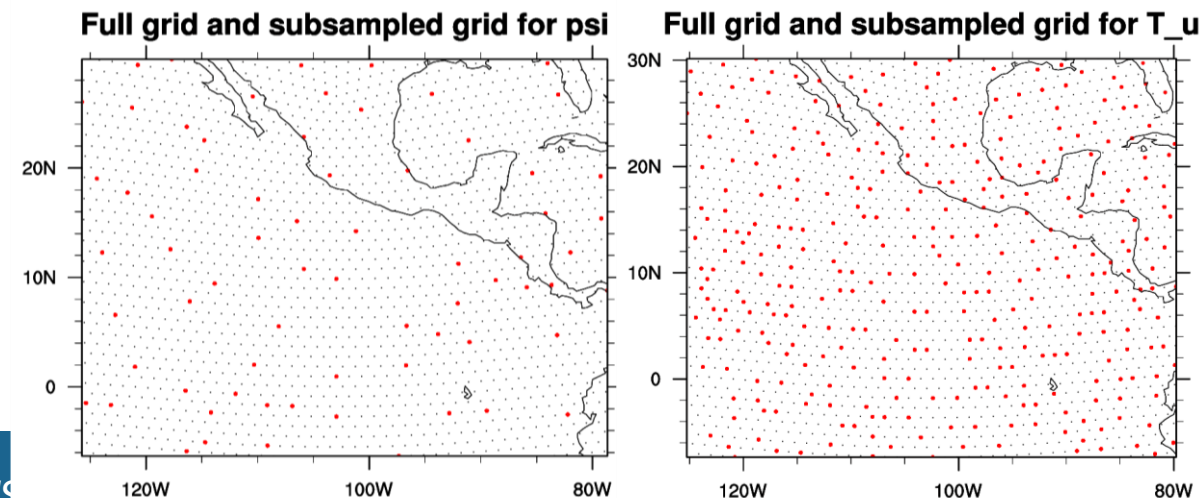

 $\mathbb{R}^{n \times n}$ 

 $\mathbb{R}^{n_s \times n_s} \quad \text{with } n_s \ll n$ 

$\mathbf{N}$  : diagonal matrix for normalization  
(to ensure the diagonal component of  $\mathbf{C}$  equals “1”)

$\mathbf{S} = \mathbf{S}^v \mathbf{S}^h$  : Interpolation from sub-sampled to full mesh

- The correlation function follows the shape of Gaspari-Cohn (1999)’s 5<sup>th</sup> order functions, for the given lengths.



$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T \mathbf{K}_2^T \mathbf{K}_1^T$$

- In previous slides, we learned how  $\mathbf{B}$  is designed and each operator works.
- Some of the operators requires the known parameters.
- $\mathbf{K}_2$  needs the regression coefficients (i.e.,  $\mathbf{L}$ ,  $\mathbf{M}$ , and  $\mathbf{N}$ ).
- The error standard deviation constructs  $\mathbf{\Sigma}$ .
- The horizontal and vertical correlation lengths are required for  $\mathbf{C}$ .
- How can we estimate (or train or calibrate) these parameters ?
  - From *some* samples : proxy for background error
  - Need to apply the inverse of each operator

( Like *GEN\_BE* in WRFDA or GSI )

# B training and tuning (1/6)

- We can estimate the required static B parameters from samples of forecast differences with two different lead times, which is called “*NMC method*”.
- In the hands-on practice, we will take the 48 hour- and 24 hour- forecast differences as a proxy of background error. **More in the hands-on practice**
- Alternatively, we can use the ensemble forecast as a proxy of background error, which is called “*ensemble method*”.
- After a proxy of background error (some form of perturbations) is determined, we can process the following procedures to estimate the static B parameters.

## B training and tuning (2/6)

- $K_1$  does not have parameters. It is basically some spatial derivative operator. However, we still need to apply its inverse to the training samples to estimate the other static B parameters.
- $K_1^{-1}$  ( converting  $u$  and  $v$  to  $\psi$  and  $\chi$ ) is **not trivial** on the unstructured grid.
- So, we...
  - 1) Interpolate  $\{u, v\}$  on the unstructured grid to regular lat/lon grid,
  - 2) Calculate  $\{\psi, \chi\}$  on the regular lat/lon grid through “uv2sfvpf” function of **NCL**,
  - 3) Interpolate  $\{\psi, \chi\}$  on the regular lat/lon to the MPAS unstructured grid.
- For convenience, “temperature” and “specific humidity” are also pre-calculated.
- So, the **perturbation fields**  $\{\delta\psi, \delta\chi, \delta T, \delta Q, \delta p_s\}$  can be obtained by subtracting the 24 hour forecast from 48 hour forecast at the same valid time.

**More in the hands-on practice**

## B training and tuning (3/6)

- The parameters in  $K_2$  can be estimated with BUMP Vertical **BAL**ance (VBAL) driver.
- The regression coefficients ( $\mathbf{L}$ ,  $\mathbf{M}$ , and  $\mathbf{N}$ ) can be obtained by multiplying **inverse of auto-covariance matrix**  $\langle \delta\psi, \delta\psi \rangle$  to the cross-covariance matrices  $\langle \delta\psi, \delta\chi \rangle$ ,  $\langle \delta\psi, \delta T \rangle$ , and  $\langle \delta\psi, \delta p_s \rangle$ .

$$\text{e.g. } \mathbf{M} = \langle \delta\psi, \delta\psi \rangle^{-1} \langle \delta\psi, \delta T \rangle$$

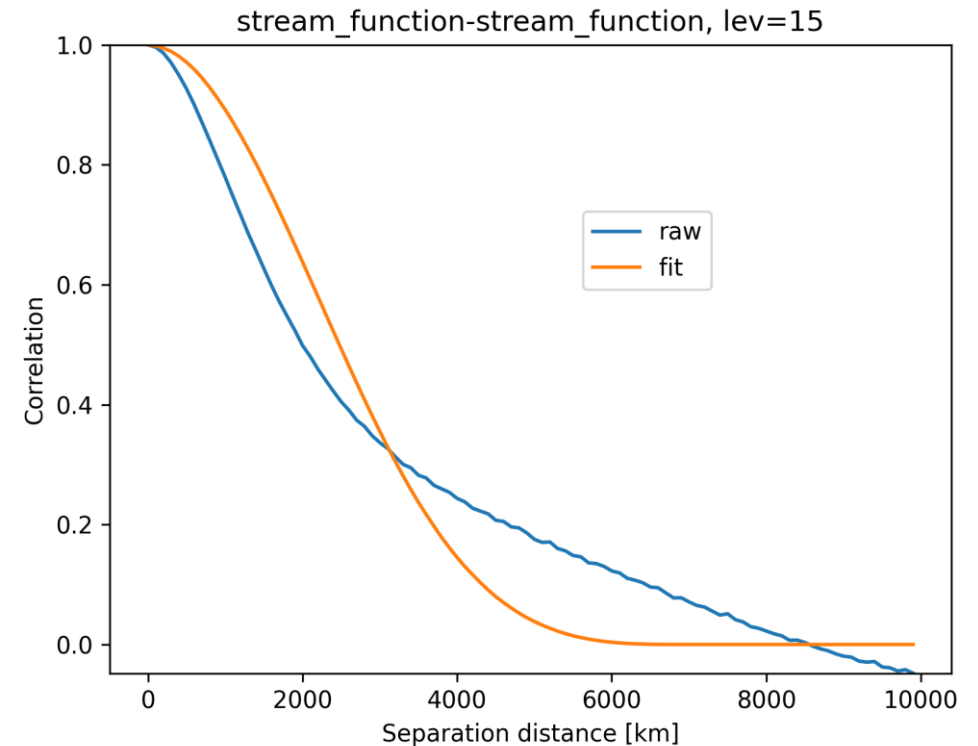
- Usually, the auto- and cross-covariance matrices are averaged over some latitude bands.
- Due to the large condition number of auto-covariance matrix, a pseudo-inverse is used with an Eigenvalue Decomposition.

## B training and tuning (4/6)

- After reading the training samples, it will also apply  $K_2^{-1}$  to the training samples to get the **unbalanced samples**  $\{\delta\psi, \delta\chi_u, \delta T_u, \delta Q, \delta p_{s,u}\}$ , which are used for the next steps.
- $\Sigma$  is estimated from unbalanced samples with BUMP **VAR**iance (VAR) driver.
- The horizontal-, vertical- correlation lengths for the unbalanced variables are estimated from samples with BUMP **Hybrid Diagnostics** (HDIAG) driver.
  - It defines the low-resolution grid for computation, and the diagnostic point is randomly selected for different horizontal separation distances.
  - For different horizontal separation distances, the raw sample correlations are calculated.
  - The vertical correlation is also calculated at each computation grid, between each level and the neighboring levels.
  - These raw sample correlations are averaged over all computation grid or locally.
  - Finally, HDIAG fits the Gaspari-Cohn's function for each averaged correlation curves.

# B training and tuning (5/6)

- Additional tunings are applied.
- Reducing the error STD for all variables by a factor of 1/3
  - To scale the 24 hour difference in the NMC-type perturbations as 6 hour (typical DA cycle)
- Reducing the diagnosed horizontal lengths for  $\delta\psi$  and  $\delta\chi_u$  by a factor of  $\frac{1}{2}$ 
  - To match the wind variance near the small separations better
  - The implied wind variance is proportional to the second derivatives of  $\delta\psi$  and  $\delta\chi_u$ .



Sample correlation structure

Fitted structure to Gaspari-Cohn's 5th order function

## B training and tuning (6/6)

- For a given horizontal and vertical correlation lengths, NICAS driver will pre-generate the correlation values on the subgrid, the interpolation weights and indices between subgrid and full grid, and the normalization factors.
- The subgrid will be determined by a given horizontal and vertical correlation lengths and some yaml keys.

```
nicas:
```

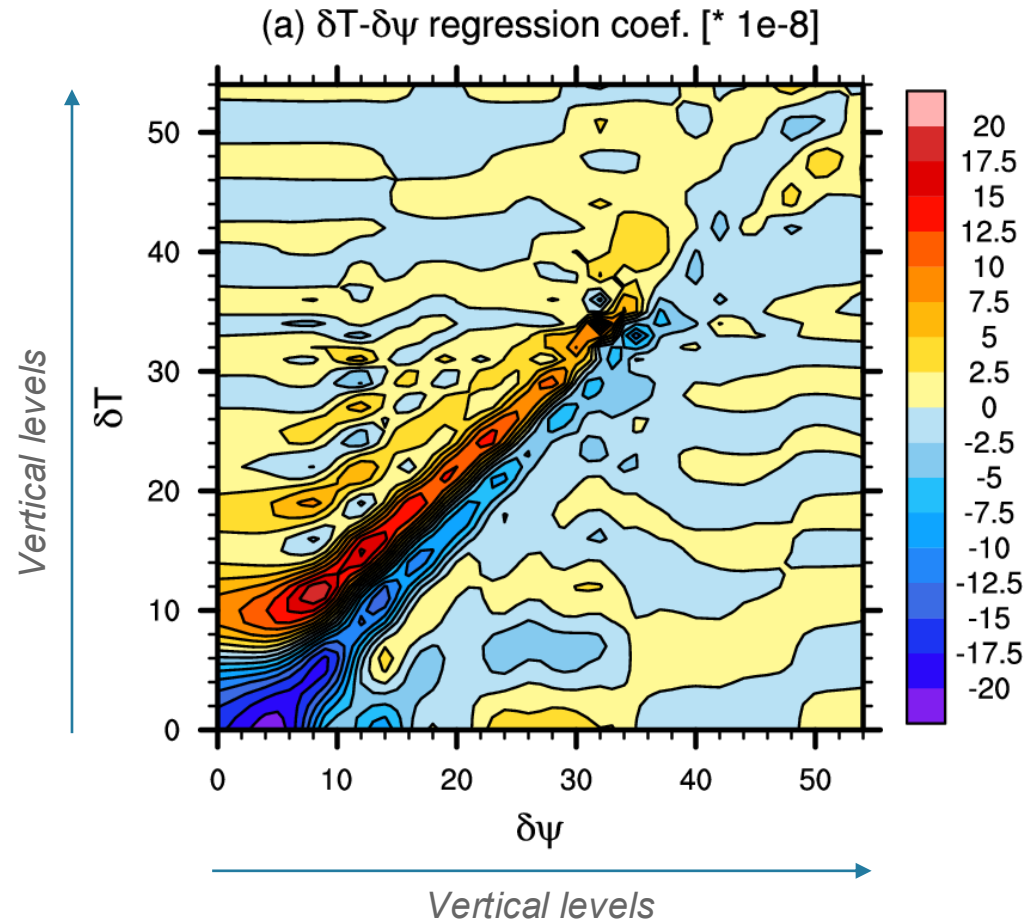
```
  resolution: 8
```

```
  max horizontal grid size: 15000 #default
```

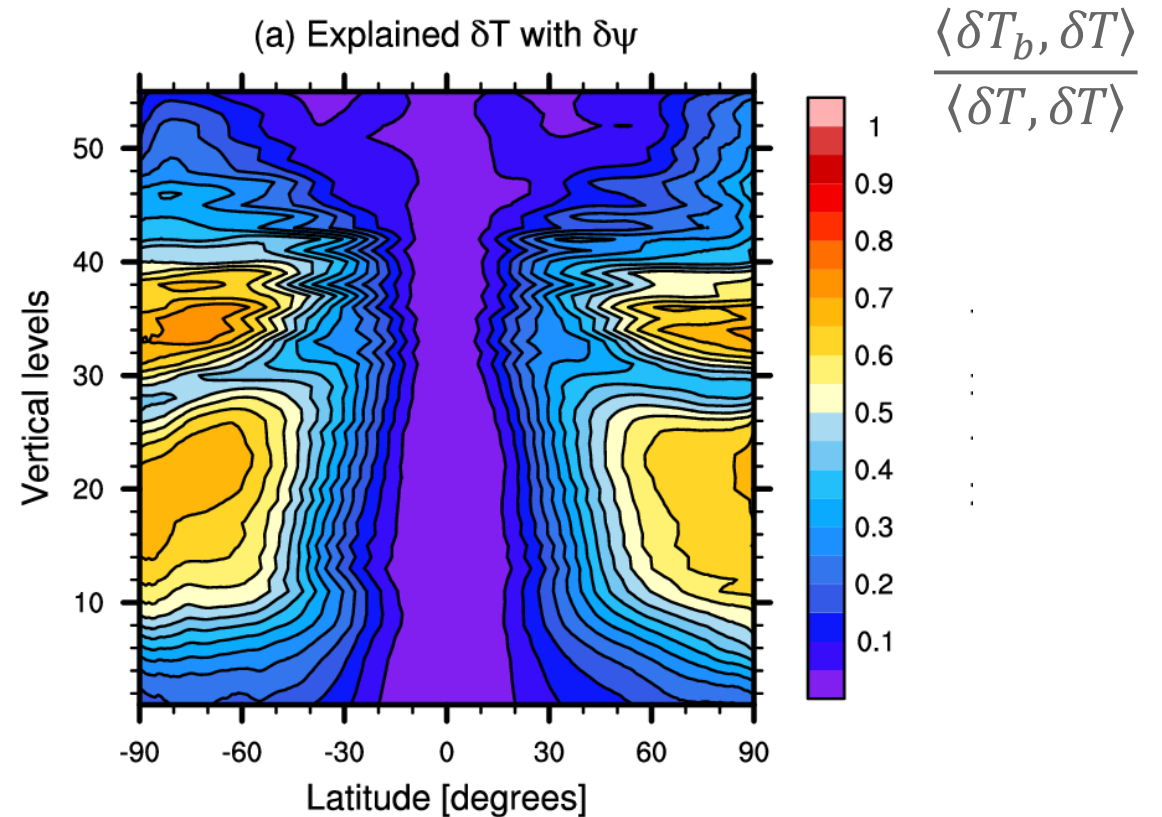
number of grid point to represent a single correlation function

number of maximum horizontal grid to define the “subgrid”

# Regression coefficients $M$ (part of $K_2$ ) at $34.8^\circ$ N latitude



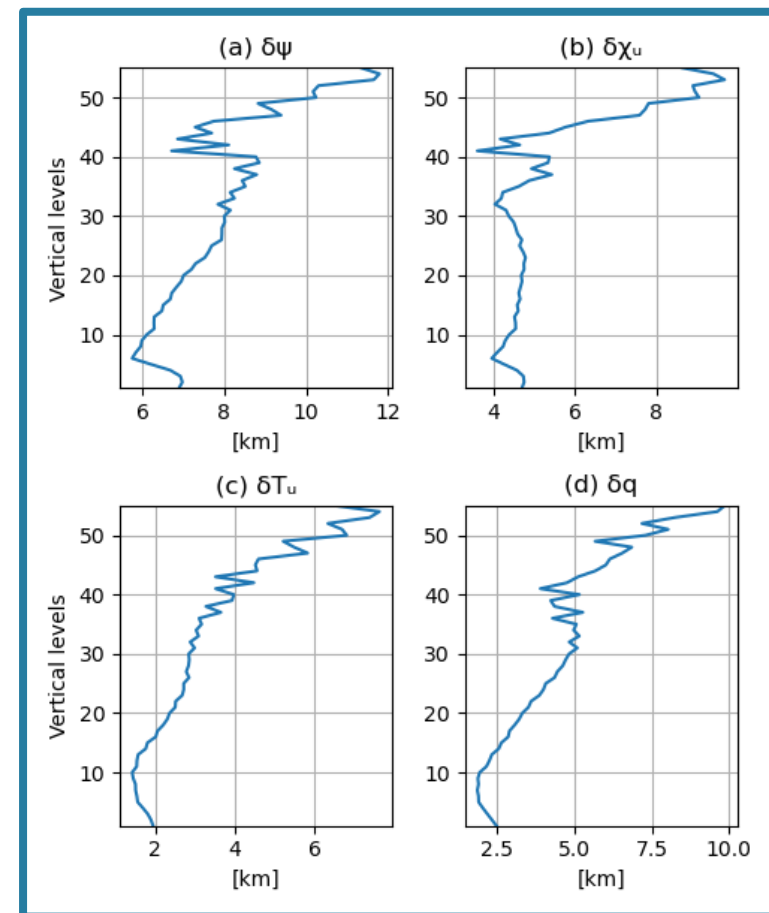
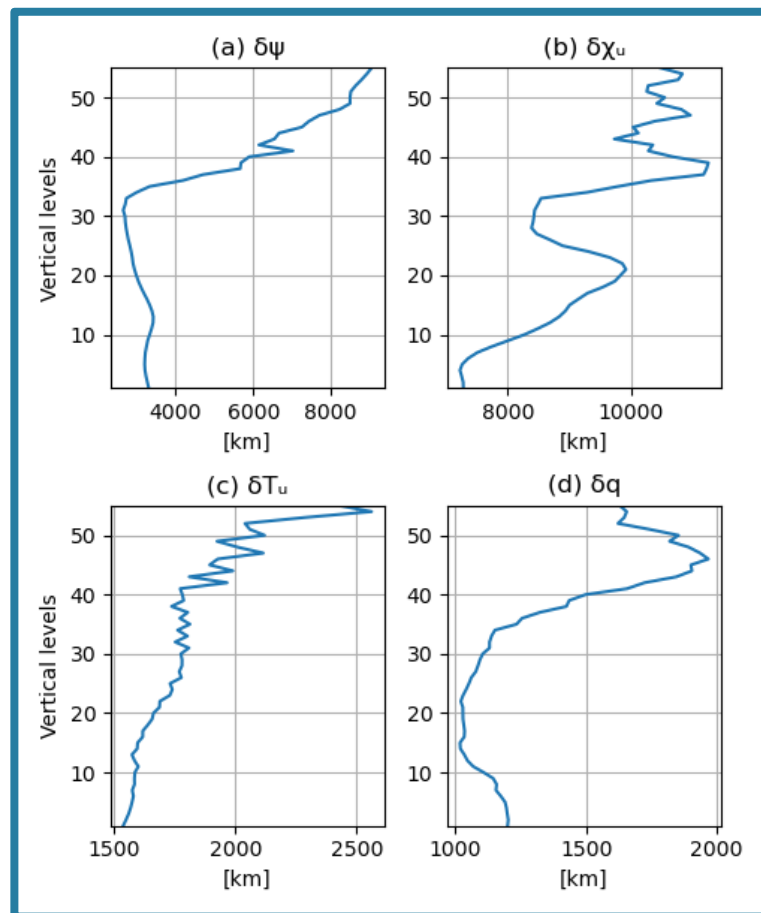
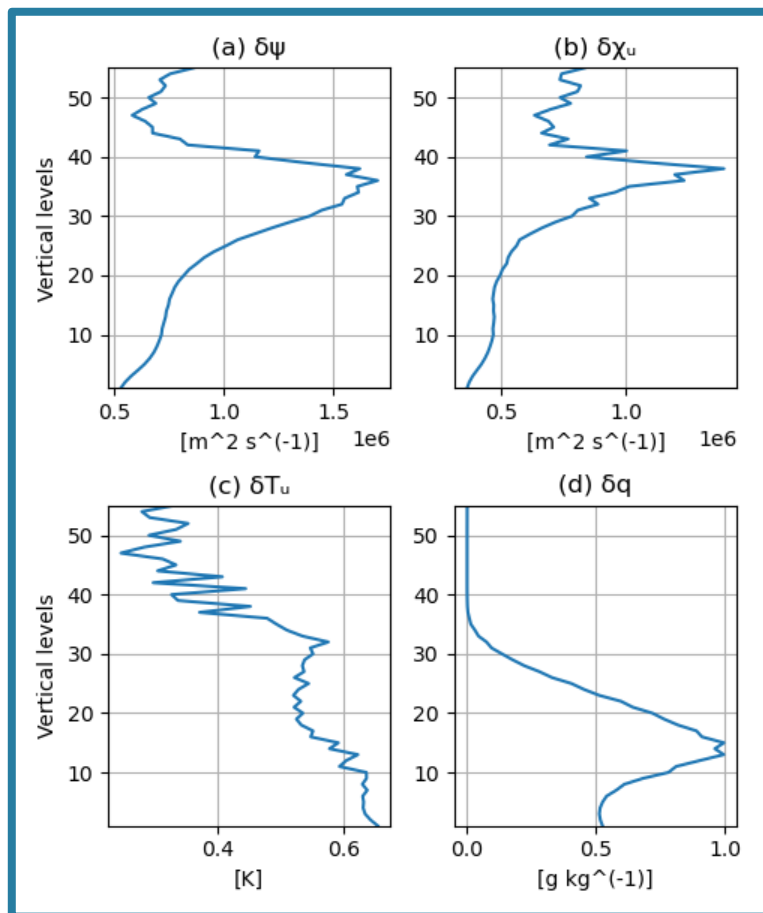
# Ratio of balanced variance to total variance



## Error standard dev. $\Sigma$

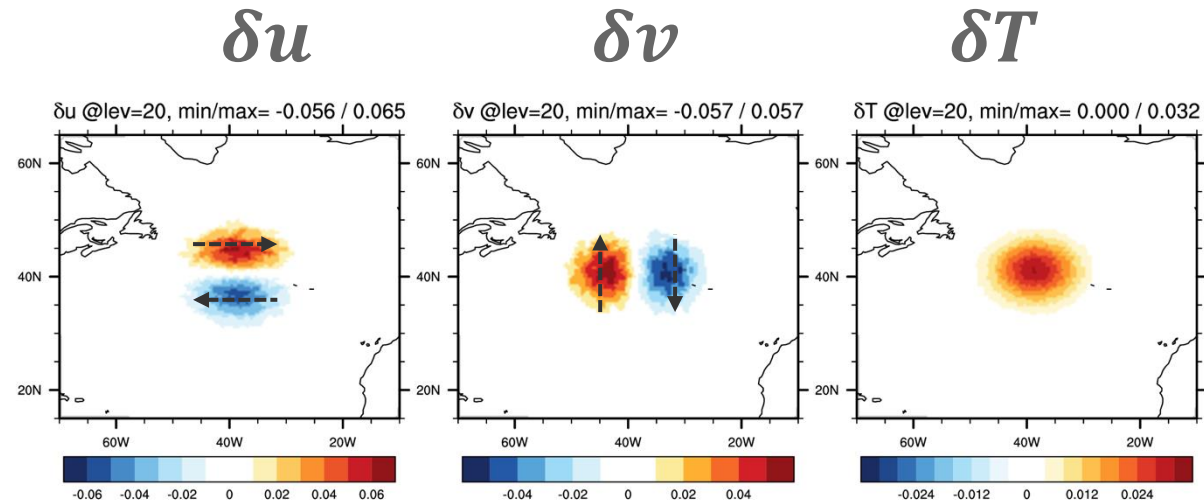
## Horizontal corr. lengths

## Vertical corr. lengths

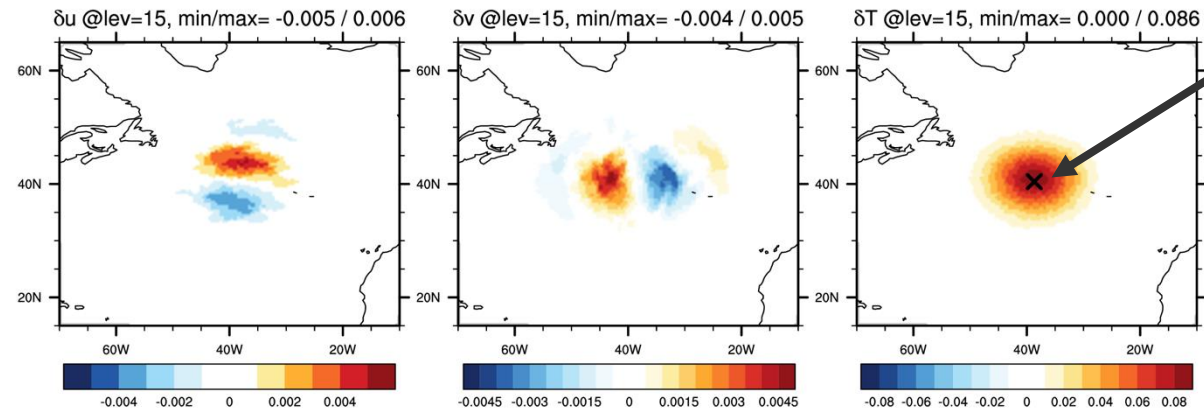


# Single T obs test

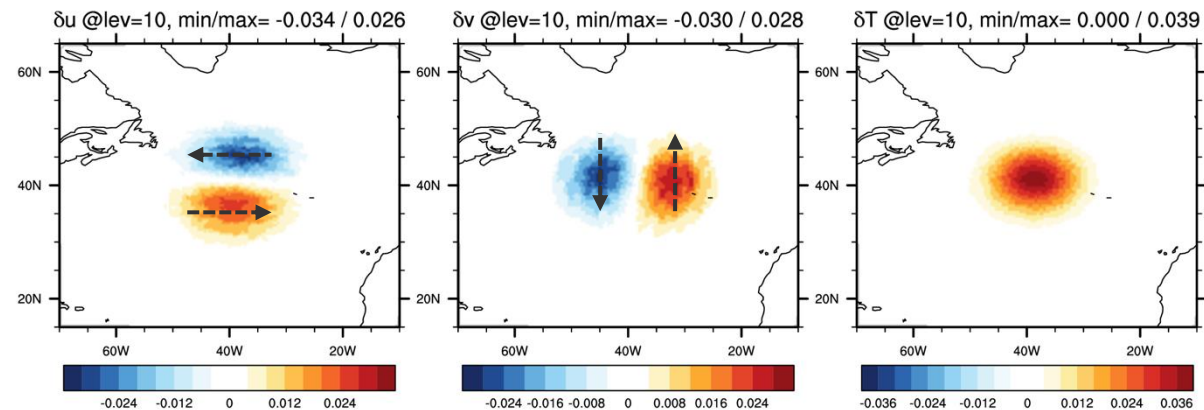
Model level = 20



Model level = 15

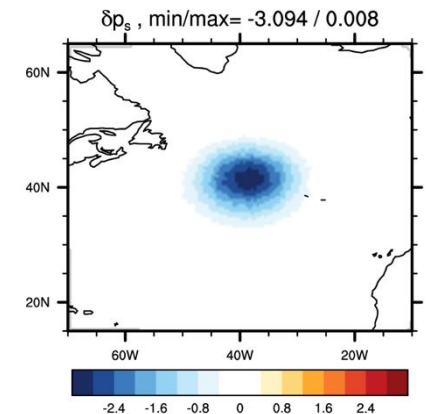


Model level = 10



obs location

$\delta p_s$




$$\mathbf{B} = \mathbf{K}_1 \mathbf{K}_2 \mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T \mathbf{K}_2^T \mathbf{K}_1^T$$

Previous slides describe ‘multivariate’  $\mathbf{B}$ .  
MPAS-JEDI can easily do ‘**univariate**’  $\mathbf{B}$ , in that case:

$$\mathbf{B} = \mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^T$$

i.e., no cross-variable correlation between analysis variables of  $\{\delta u, \delta v, \delta T, \delta Q, \delta p_s\}$

This should work both for **global** and **regional** MPAS meshes.

# Yaml configuration for 3DVar (1/6)

```
cost function:
  cost type: 3D-Var
  window begin: 2018-04-14T21:00:00Z
  window length: PT6H

  analysis variables: &incvars [air_temperature, water_vapor_mixing_ratio_wrt_moist_air,
eastward_wind, northward_wind, air_pressure_at_surface]

background:
  state variables: [air_temperature, water_vapor_mixing_ratio_wrt_moist_air, eastward_wind,
northward_wind, air_pressure_at_surface, air_potential_temperature, dry_air_density, u,
water_vapor_mixing_ratio_wrt_dry_air, air_pressure, landmask, seaice_fraction, snowc,
skin_temperature_at_surface, ivgtyp, isltyp, snowh, vegetation_area_fraction, eastward_wind_at_10m,
northward_wind_at_10m, lai, smois, tslb, pressure_p]
  filename: ./bg.2018-04-15_00.00.00.nc
  date: &analysisDate 2018-04-15T00:00:00Z
```

{ 3D-Var, 4D-Ens-Var, 3D-FGAT }

The variables that are used for the minimization algorithm.

Background state usually includes more than just “analysis variables”

Background state is read from this file.

Analysis time is at the center of assimilation window.

# Yaml configuration for 3DVar (2/6)

cost function:

...

background error:

**covariance model:** SABER

**saber central block:**

**C** `saber block name: BUMP_NICAS`  
`... more config ...`

**saber outer blocks:**

**$\Sigma$**  `- saber block name: StdDev`  
`... more config ...`

**$K_2$**  `- saber block name: BUMP_VerticalBalance`  
`... more config ...`

**$K_1$**  **linear variable change:**  
`linear variable change name: Control2Analysis`  
`... more config ...`

$$B = K_1 K_2 \Sigma C \Sigma^T K_2^T K_1^T$$



*In order of forward (TL) operations*

More generic information on SABER error covariance model and its block chain design can be found [here](#).

# Yaml configuration for 3DVar (3/6)

background error:

covariance model: SABER

saber central block:

**saber block name:** **BUMP\_NICAS**

**active variables:** **&ctlvars** [air\_horizontal\_streamfunction, air\_horizontal\_velocity\_potential, air\_temperature, water\_vapor\_mixing\_ratio\_wrt\_moist\_air, air\_pressure\_at\_surface]

**read:**

**io:**

**data directory:** ./B\_Matrix/bump\_nicas

**files prefix:** mpas

**drivers:**

**multivariate strategy:** univariate

**read local nicas:** true

$$B = K_1 K_2 \Sigma \mathbf{C} \Sigma^T K_2^T K_1^T$$

*“univariate” strategy is used for covariance C.f. “duplicated” or “crossed” strategy is used when NICAS is used for localization*

*Read the pre-generated NICAS files. Here, “local” nicas means one NICAS file per one processor. C.f. “read global nicas: true” reads single NICAS file, which contains the information over all domain.*

*Note that there is NO distinction with “total” and “unbalanced” variable names in the YAMLS or files.*

# Yaml configuration for 3DVar (4/6)

background error:

covariance model: SABER

saber central block:

saber block name: BUMP\_NICAS

... more config ...

saber outer blocks:

- **saber block name: StdDev**

read:

model file:

filename: ./B\_Matrix/stddev/mpas.stddev.nc

date: \*analysisDate

stream name: control

*Read the standard deviation values from MPAS file format.*

$$B = K_1 K_2 \Sigma C \Sigma^T K_2^T K_1^T$$

# Yaml configuration for 3DVar (5/6)

- saber block name: **BUMP\_VerticalBalance**

read:

io:

data directory: ./B\_Matrix/bump\_vertical\_balance

files prefix: mpas

drivers:

read global sampling: true

read vertical balance: true

vertical balance:

vbal:

- balanced variable: air\_horizontal\_velocity\_potential

unbalanced variable: air\_horizontal\_streamfunction

diagonal regression: true

- balanced variable: air\_temperature

unbalanced variable: air\_horizontal\_streamfunction

- balanced variable: air\_pressure\_at\_surface

unbalanced variable: air\_horizontal\_streamfunction

$$B = K_1 K_2 \Sigma C \Sigma^T K_2^T K_1^T$$

Read the pre-generated "global" VBAL & sampling files.

These describe the designed balance (regression) relationship.

# Yaml configuration for 3DVar (6/6)

background error:

covariance model: SABER

saber central block:

saber block name: BUMP\_NICAS

... more config ...

saber outer blocks:

- saber block name: StdDev

... more config ...

- saber block name: BUMP\_VerticalBalance

... more config ...

**linear variable change:**

linear variable change name: **Control2Analysis**

input variables: **\*ctlvars**

output variables: **\*incvars**

$$B = K_1 K_2 \Sigma C \Sigma^T K_2^T K_1^T$$

# Yaml configuration for Hybrid-3DEnVar (1/2)

- 3DVar uses  
cost function:  
background error:  
covariance model: **SABER**  
... more config ...
- 3DEnVar uses  
cost function:  
background error:  
covariance model: **ensemble**  
... more config ...
- We can configure the hybrid covariance as a linear combination of other Bs !

$$\mathbf{B}_{\text{hybrid}} = \alpha \mathbf{B}_{\text{static}} + \beta \mathbf{B}_{\text{ensemble}}$$

*(Hamill and Snyder, 2000)*

# Yaml configuration for Hybrid-3DEnVar (2/2)

- We can configure the hybrid covariance as a linear combination of other Bs !

background error:

covariance model: **hybrid**

components:

- weight:

value: 0.5

covariance:

covariance model: **SABER**

... more config ...

- weight:

value: 0.5

covariance:

covariance model: **ensemble**

... more config ...

$$\mathbf{B}_{\text{hybrid}} = \alpha \mathbf{B}_{\text{static}} + \beta \mathbf{B}_{\text{ensemble}}$$

*The weights can be a simple scalar value or they can be read from a file, which varies spatially.*

# References

- Bannister, R.N. (2008a), A review of forecast error covariance statistics in atmospheric variational data assimilation. I: Characteristics and measurements of forecast error covariances. Q.J.R. Meteorol. Soc., 134: 1951-1970. <https://doi.org/10.1002/qj.339>
- Bannister, R.N. (2008b), A review of forecast error covariance statistics in atmospheric variational data assimilation. II: Modelling the forecast error covariance statistics. Q.J.R. Meteorol. Soc., 134: 1971-1996. <https://doi.org/10.1002/qj.340>
- Derber, J. and Bouttier, F.: A reformulation of the background error covariance in the ECMWF global data assimilation system, Tellus A: Dynamic Meteorology and Oceanography, 51, 195–221, <https://doi.org/10.3402/tellusa.v51i2.12316> , 1999.
- Gaspari, G. and Cohn, S. E.: Construction of correlation functions in two and three dimensions, Quarterly Journal of the Royal Meteorological Society, 125, 723–757, <https://doi.org/https://doi.org/10.1002/qj.49712555417> , 1999.
- Hamill, T. M. and Snyder, C.: A Hybrid Ensemble Kalman Filter–3D Variational Analysis Scheme, Monthly Weather Review, 128, 2905 – 2919, [https://doi.org/10.1175/1520-0493\(2000\)128<2905:AHEKFV>2.0.CO;2](https://doi.org/10.1175/1520-0493(2000)128<2905:AHEKFV>2.0.CO;2) , 2000.
- Jung, B.-J., Ménétrier, B., Snyder, C., Liu, Z., Guerrette, J. J., Ban, J., Baños, I. H., Yu, Y. G., and Skamarock, W. C.: Three-dimensional variational assimilation with a multivariate background error covariance for the Model for Prediction Across Scales–Atmosphere with the Joint Effort for data Assimilation Integration (JEDI-MPAS 2.0.0-beta), Geosci. Model Dev., 17, 3879–3895, <https://doi.org/10.5194/gmd-17-3879-2024> , 2024.
- Ménétrier, B.: The Normalized Interpolated Convolution from an Adaptive Subgrid (NICAS) method, Geosci. Model Dev., 19, 4497–4511, <https://doi.org/10.5194/gmd-19-4497-2026> , 2026.
- Wu, W.-S., Purser, R. J., and Parrish, D. F.: Three-dimensional variational analysis with spatially inhomogeneous covariances, Monthly Weather Review, 130, 2905–2916, [https://doi.org/10.1175/1520-0493\(2002\)130<2905:TDVAWS>2.0.CO;2](https://doi.org/10.1175/1520-0493(2002)130<2905:TDVAWS>2.0.CO;2) , 2002.