BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr
Funded by the JCSDA

4th JEDI Academy - February 2020

Basic facts

- Background error covariance (aka **B** matrix) is a key aspect of variational DA systems.
- In the practical implementation, the **B** matrix itself is not required, only its effect on a state vector.
- **B** must be symmetric and positive, so we generally build its square-root **U**: $\mathbf{B} = \mathbf{U}\mathbf{U}^\mathrm{T}$.
- Usual covariance models are:
  - static **B**: a sequence of parametrized operators
  - ensemble-based **B**: a localized sample covariance matrix
  - hybrid **B**: a linear combination of previous models
- All covariance models require a smoother to spread the innovation information. In most implementations, smoothers are grid-specific (e.g. spectral transform, recursive filters).

BUMP library

- BUMP: "Background error on Unstructured Mesh Package"
- It is designed to work on any grid (gaussian, cubed-sphere, unstructured, limited-area, ocean).
- It can diagnose parameters for all the usual covariance models.
- It implements a generic smoother, NICAS ("Normalized Interpolated Convolution from an Adaptive Subgrid").
- The code is written in Fortran 90, around 25.000 lines.
- It is part of the SABER repository ("System-Agnostic Background Error Representation"), which will include other covariance modeling libraries in the future.
- It is fully interfaced with OOPS, for both diagnostic and application parts.

BUMP credo: subsampling

High-resolution grids are great for models dynamics, but very costly and not really useful for background error covariance modeling.

In general, the size of the background error structures that we can represent is significantly larger than the model grid cell size.

The credo of BUMP is:

You shall cleverly subsample the model grids.

You shall perform costly operations on the subsampled mesh.

You shall interpolate your final results on the model grid.

-The BUMP book

Outline

Introduction

Static **B**

Ensemble/hybrid **B**

The NICAS smoother

BUMP usage

Outline

Static **B** basics

- The static **B** is a robust and well-conditioned model, based on a sequence of parametrized operators.
- The parameters can be defined using ensemble data over a long period, giving a climatological estimate.
- They can also be estimated over a shorter sliding window, and updated at every cycle.
- The most common static **B** model is:

$$\mathbf{B}^s = \mathbf{K}_p \mathbf{\Sigma} \mathbf{C} \mathbf{\Sigma}^{\mathrm{T}} \mathbf{K}_p^{\mathrm{T}}$$

where

  - $\mathbf{C}$ is a correlation matrix
  - $\mathbf{\Sigma}$ is a diagonal matrix of standard deviations
  - $\mathbf{K}_p$ is a multivariate balance operator

Static **B** with BUMP

BUMP can be used for all these operators:

- The correlation length-scales of $\mathbf{C}$ are estimated globally or locally, and used to set up the NICAS smoother, which is exactly normalized ($C_{ii} = 1$).
- The standard-deviations in $\mathbf{\Sigma}$ are estimated locally and potentially filtered, to remove the sampling noise (objective filtering of Ménétrier *et al.*, 2015a,b).
- A vertical balance operator computing regressions between variables can be diagnosed and applied with BUMP (still under development), as part of a more complex $\mathbf{K}_p$.

In the OOPS framework, the static B components can come from BUMP or from your own model, and be combined as you wish.

Outline

Sample covariance

An ensemble of N forecasts $\{\mathbf{x}_p^b\}$ is used to estimate the sample covariance matrix $\widetilde{\mathbf{B}}$:

$$\widetilde{\mathbf{B}} = \frac{1}{N-1} \sum_{p=1}^{N} \delta\mathbf{x}_p^b \delta\mathbf{x}_p^{b\mathrm{T}}$$

where $\delta\mathbf{x}_p^b$ is the p^{th} ensemble perturbation:

$$\delta\mathbf{x}_p^b = \mathbf{x}_p^b - \langle \mathbf{x}^b \rangle \quad \text{and} \quad \langle \mathbf{x}^b \rangle = \frac{1}{N} \sum_{p=1}^{N} \mathbf{x}_p^b$$

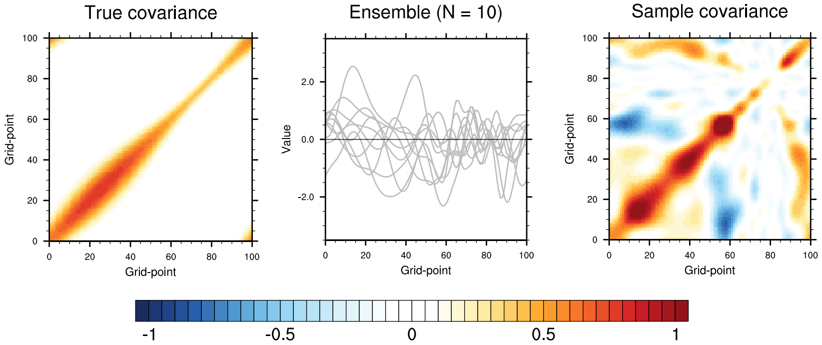
Asymptotic sample covariance: $\mathbf{B} = \lim_{N \to \infty} \widetilde{\mathbf{B}}$

Since the ensemble size $N < \infty$, $\widetilde{\mathbf{B}}$ is affected by sampling noise:

$$\widetilde{\mathbf{B}}^e = \widetilde{\mathbf{B}} - \mathbf{B}$$

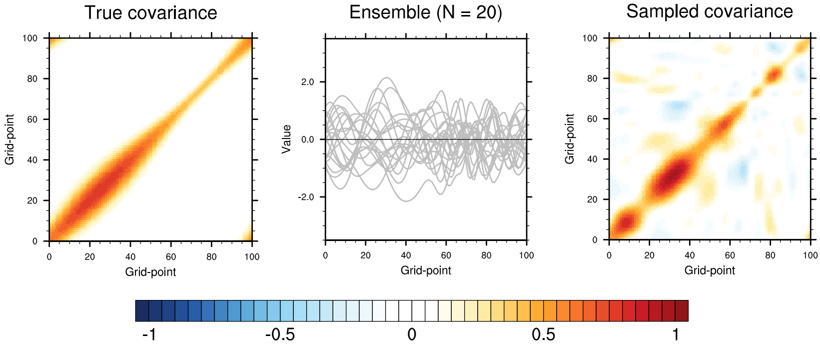
Sample covariance

Sampling noise strongly depends on the ensemble size:



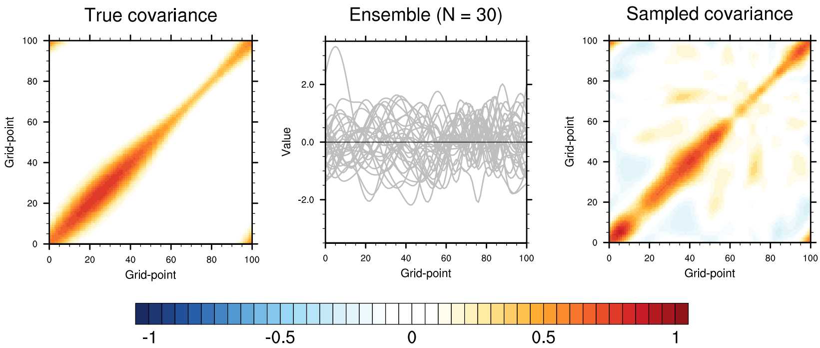
Sample covariance

Sampling noise strongly depends on the ensemble size:



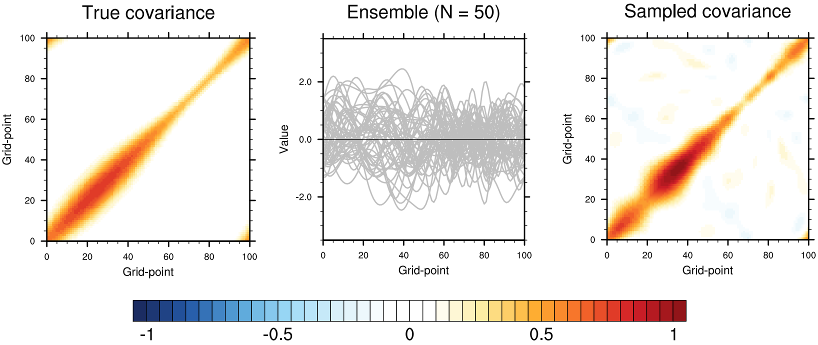
Sample covariance

Sampling noise strongly depends on the ensemble size:



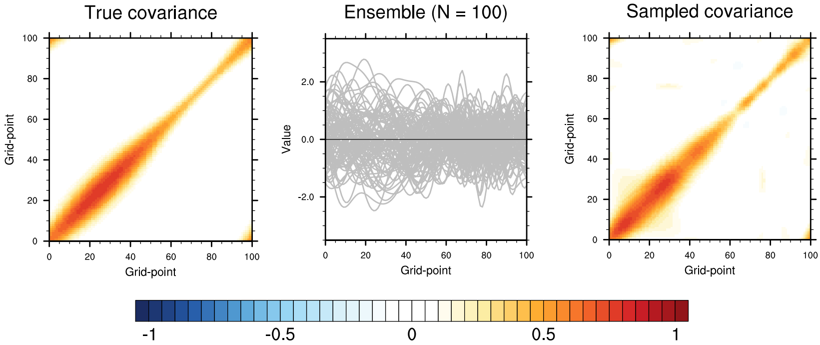
Sample covariance

Sampling noise strongly depends on the ensemble size:



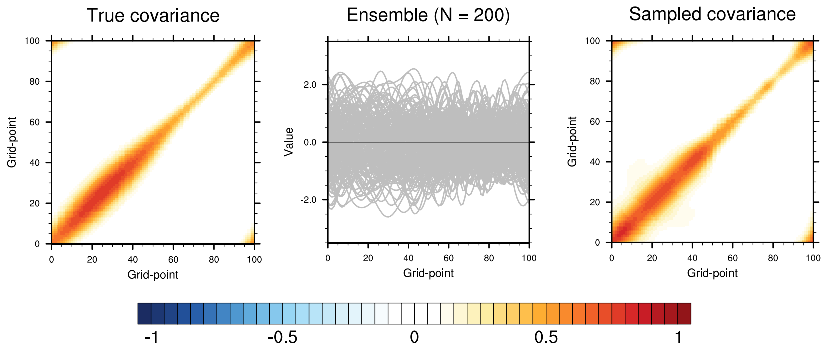
Sample covariance

Sampling noise strongly depends on the ensemble size:



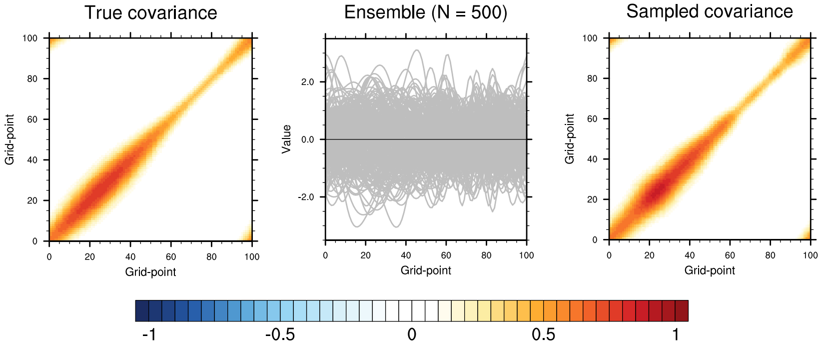
Sample covariance

Sampling noise strongly depends on the ensemble size:



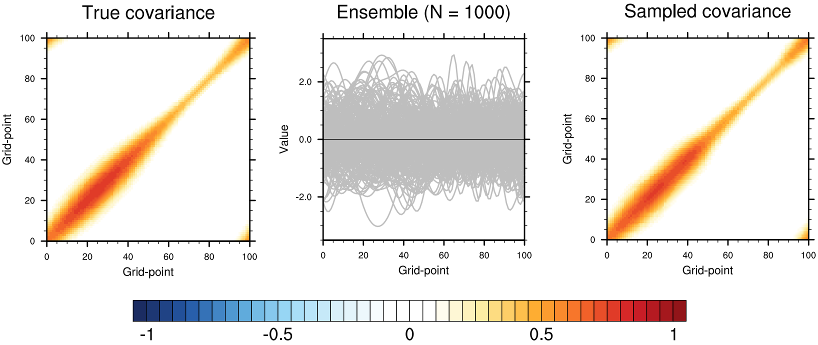
Sample covariance

Sampling noise strongly depends on the ensemble size:



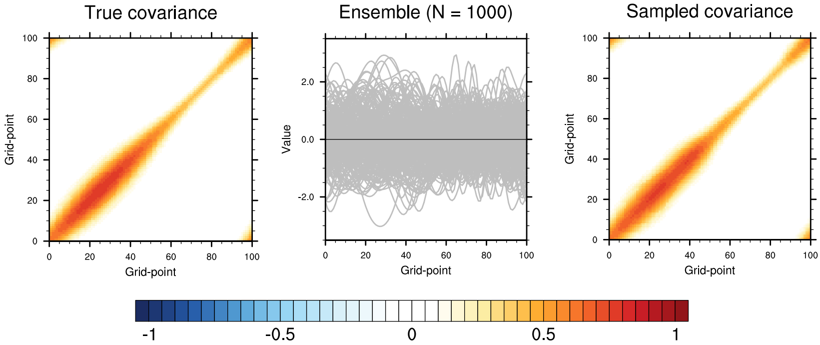
Sample covariance

Sampling noise strongly depends on the ensemble size:



Sample covariance

Sampling noise strongly depends on the ensemble size:



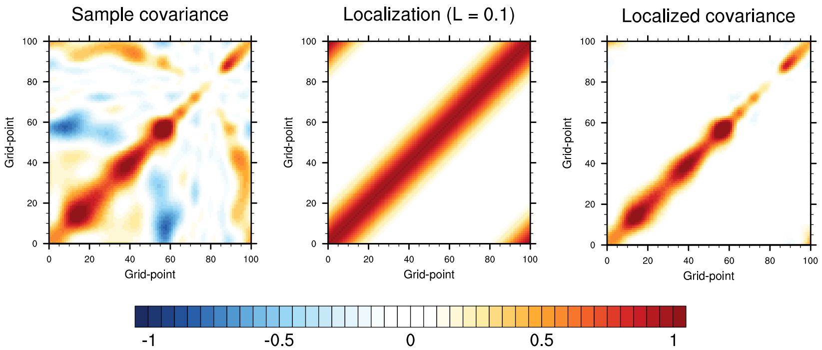
We don't have oil, but we have ideas!

Localized covariance

Sampling noise on $\widetilde{\mathbf{B}}$ can be damped via a Schur product (element-by-element) with a localization matrix $\mathbf{L}$:

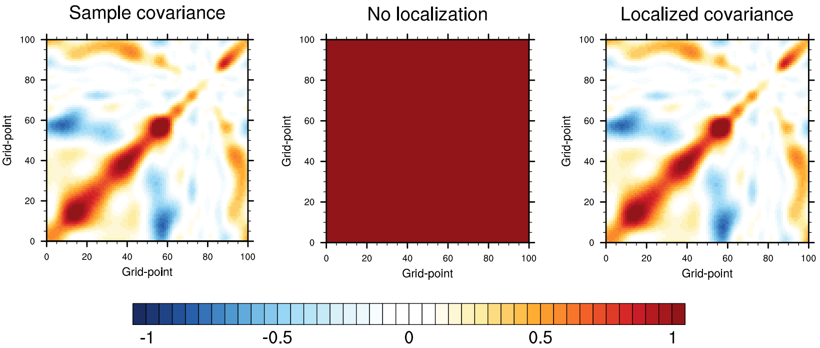
$$\widehat{\mathbf{B}} = \mathbf{L} \circ \widetilde{\mathbf{B}} \quad \Leftrightarrow \quad \widehat{B}_{ij} = L_{ij} \widetilde{B}_{ij}$$

In practice, $\mathbf{L}$ damps the long-distance correlations that are small and more affected by sampling noise (hence the "localization").



Localization: what is the optimal length-scale?

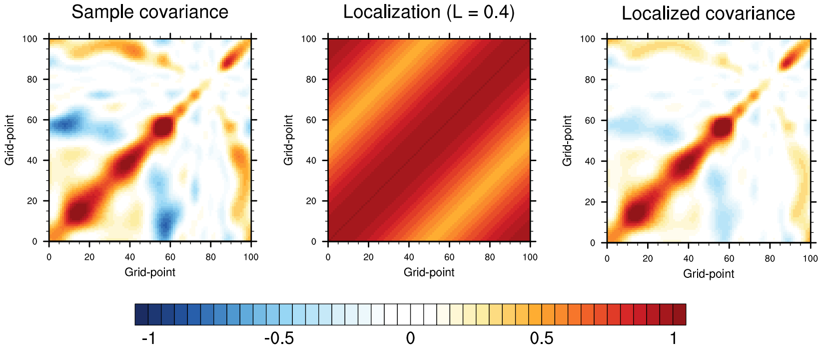
The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



No impact

Localization: what is the optimal length-scale?

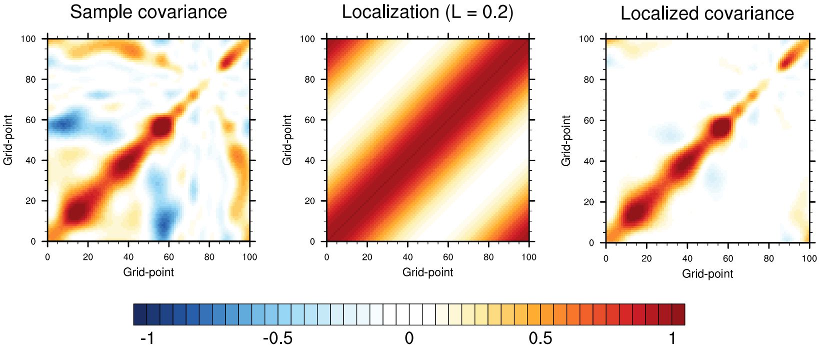
The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



Start reducing the sampling noise...

Localization: what is the optimal length-scale?

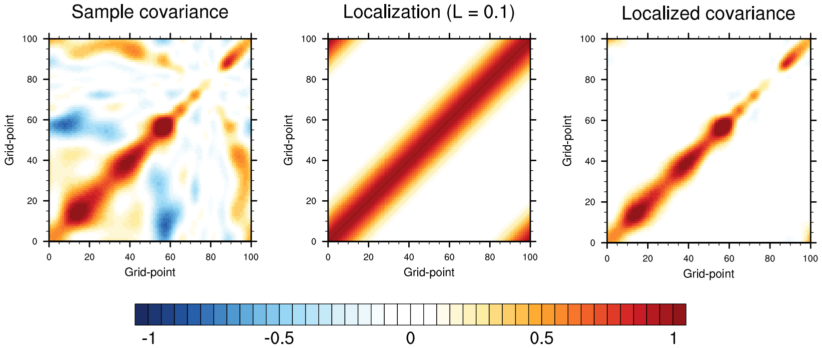
The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



Less and less sampling noise...

Localization: what is the optimal length-scale?

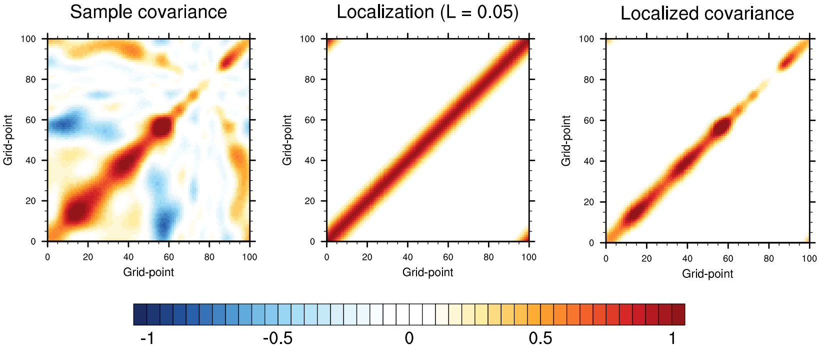
The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



Good ! Almost no sampling noise anymore...

Localization: what is the optimal length-scale?

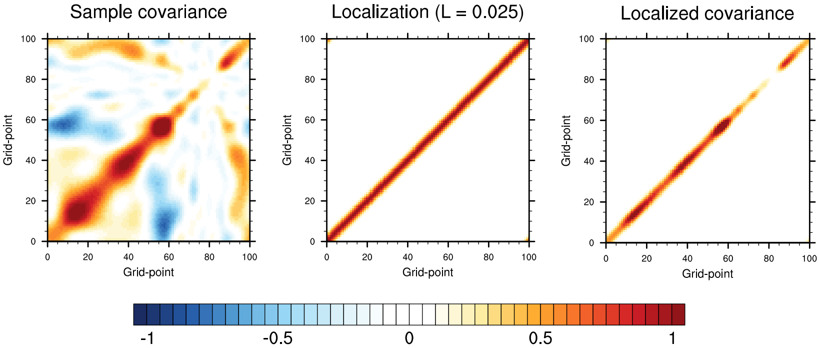
The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



Well, we are loosing some signal now...

Localization: what is the optimal length-scale?

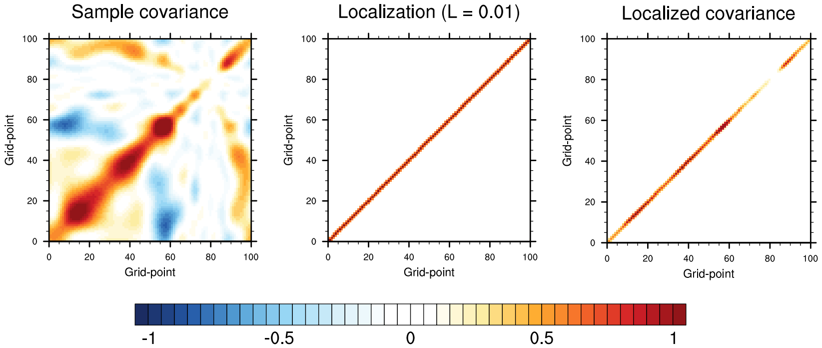
The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



Hey, stop loosing signal !

Localization: what is the optimal length-scale?

The localization length-scale is critical to remove the sampling noise while keeping the relevant covariance signal:



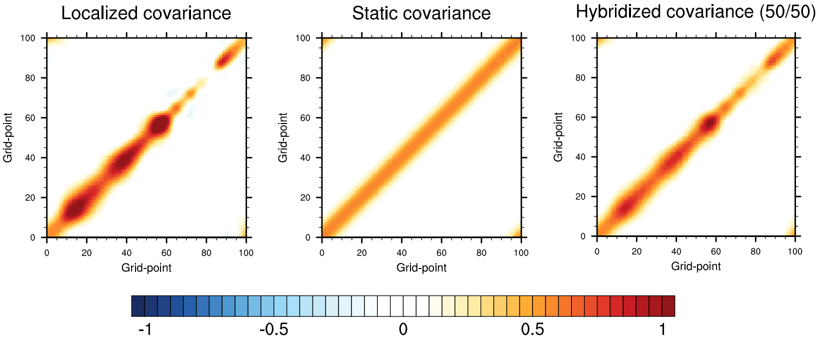
No more signal !

Localized and hybridized covariance

Deficiencies of the localized covariance matrix $\widehat{\mathbf{B}}$ can be corrected via a hybridization with a static covariance matrix:

$$\widehat{\mathbf{B}}^h = \beta^{e2}\ \widehat{\mathbf{B}} + \beta^{s2}\mathbf{B}^s \quad \Leftrightarrow \quad \widehat{B}^h_{ij} = \beta^{e2}\widehat{B}_{ij} + \beta^{s2}B^s_{ij}$$

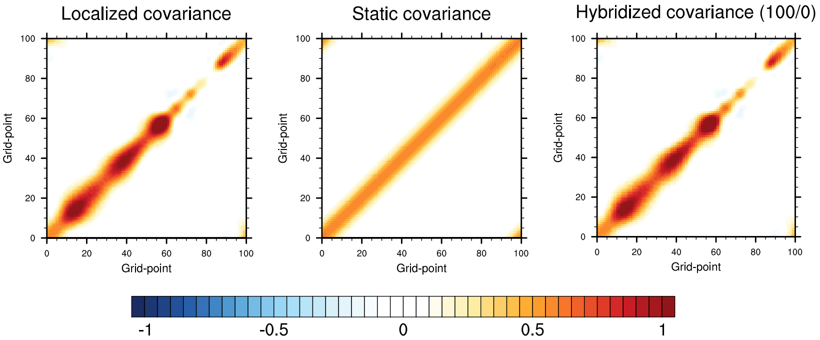
For a homogeneous $\mathbf{B}^s$:



Hybridization: what are the optimal weights

Localization + hybridation:

$$\widehat{\mathbf{B}}^h = \beta^{e2}\ \widehat{\mathbf{B}} + \beta^{s2}\mathbf{B}^s \quad \Leftrightarrow \quad \widehat{B}^h_{ij} = \beta^{e2}\widehat{B}_{ij} + \beta^{s2}B^s_{ij}$$

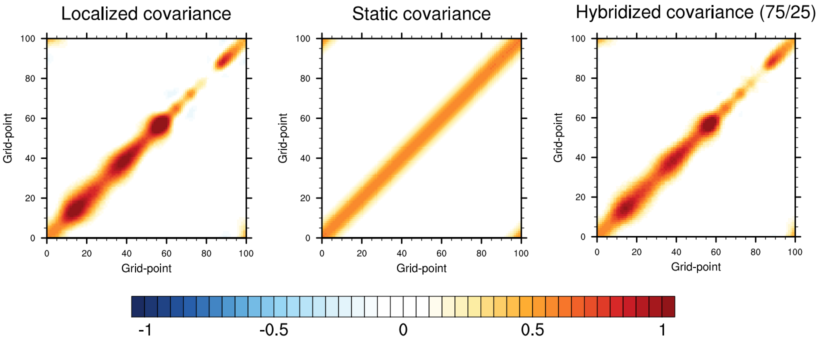


100% on the localized covariance matrix

Hybridization: what are the optimal weights

Localization + hybridation:

$$\widehat{\mathbf{B}}^h = \beta^{e2}\ \widehat{\mathbf{B}} + \beta^{s2}\mathbf{B}^s \quad \Leftrightarrow \quad \widehat{B}^h_{ij} = \beta^{e2}\widehat{B}_{ij} + \beta^{s2}B^s_{ij}$$

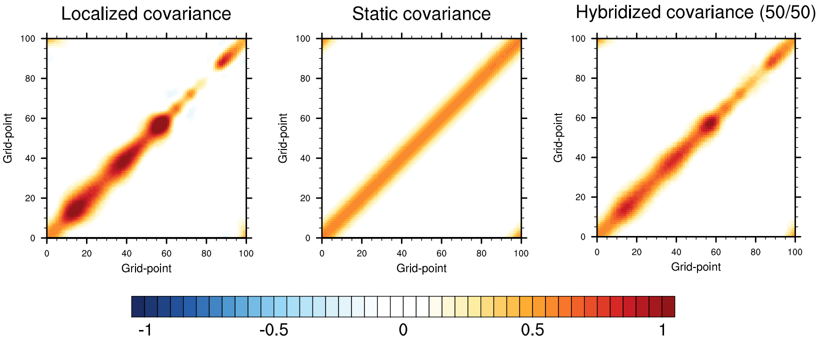


75% on the localized covariance matrix

Hybridization: what are the optimal weights

Localization + hybridation:

$$\widehat{\mathbf{B}}^h = \beta^{e2}\ \widehat{\mathbf{B}} + \beta^{s2}\mathbf{B}^s \quad \Leftrightarrow \quad \widehat{B}^h_{ij} = \beta^{e2}\widehat{B}_{ij} + \beta^{s2}B^s_{ij}$$

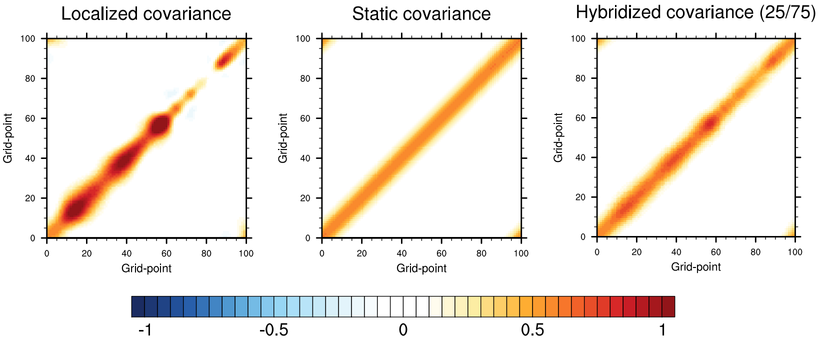


Equal weights on both covariance matrices

Hybridization: what are the optimal weights

Localization + hybridation:

$$\widehat{\mathbf{B}}^h = \beta^{e2}\ \widehat{\mathbf{B}} + \beta^{s2}\mathbf{B}^s \quad \Leftrightarrow \quad \widehat{B}^h_{ij} = \beta^{e2}\widehat{B}_{ij} + \beta^{s2}B^s_{ij}$$

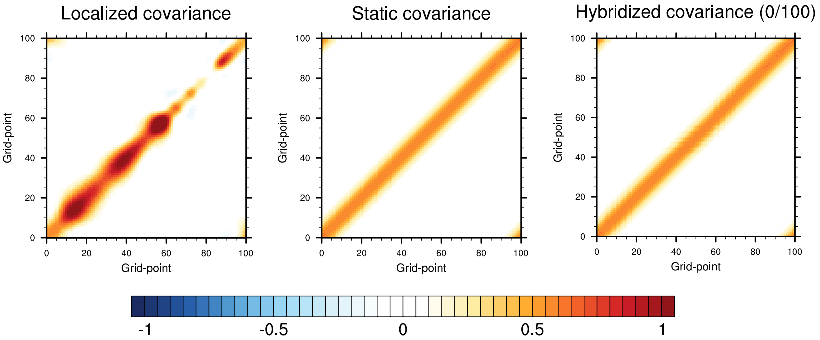


75% on the static covariance matrix

Hybridization: what are the optimal weights

Localization + hybridation:

$$\widehat{\mathbf{B}}^h = \beta^{e2}\ \widehat{\mathbf{B}} + \beta^{s2}\mathbf{B}^s \quad \Leftrightarrow \quad \widehat{B}^h_{ij} = \beta^{e2}\widehat{B}_{ij} + \beta^{s2}B^s_{ij}$$



100% on the static covariance matrix

How to optimize localization and hybridization?

Existing methods are empirical and costly (e.g. OSSE, brute-force optimization). We need a new method that:

- uses only ensemble members,
- is affordable for high-dimensional systems,
- can be generic enough to be run with all kinds of grids.

Principle :

$$\underbrace{\widehat{\mathbf{B}}^h}_{\text{Localized/hybridized covariance}} = \underbrace{\beta^{e2}}_{\text{Ensemble weight}} \underbrace{\mathbf{L}}_{\text{Localization matrix}} \circ \underbrace{\widetilde{\mathbf{B}}}_{\text{Sampled covariance}} + \underbrace{\beta^{s2}}_{\text{Static weight}} \underbrace{\mathbf{B}^s}_{\text{Static covariance}}$$

Localization + hybridization = linear filtering of $\widetilde{\mathbf{B}}$

How to optimize localization and hybridization?

Asymptotic sample covariance: $\mathbf{B} = \lim_{N \to \infty} \widetilde{\mathbf{B}}$

Residual noise (after localization/hybridization): $\widehat{\mathbf{B}}^h - \mathbf{B}$

Objectives:

- Express $\beta^{e2}\mathbf{L}$ and β^{s2} minimizing the error $\mathbb{E}\left[\|\widehat{\mathbf{B}}^h - \mathbf{B}\|^2\right]$.

  $\rightarrow$ Linear filtering theory.

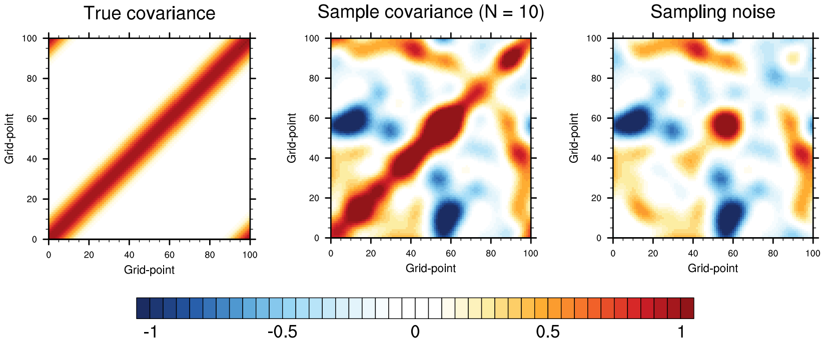
  Some statistics involve the asymptotic sample covariance.

- Express statistics on asymptotic quantities (unknown) with expected sample quantities (knowable).

  $\rightarrow$ Centered moments sampling theory (non-Gaussian case).

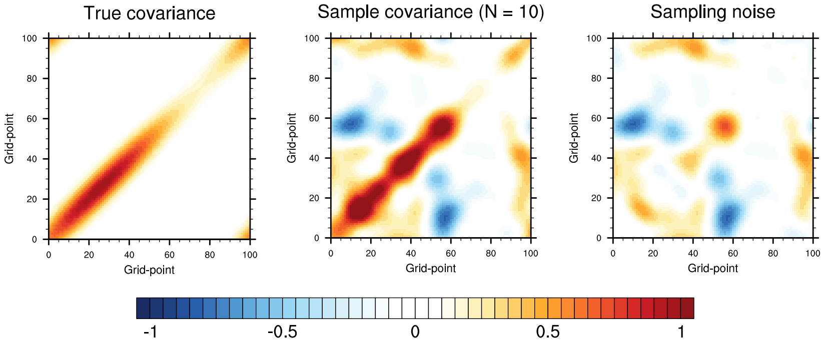
Sampling noise properties

Homogeneous variance / length-scale



Sampling noise properties

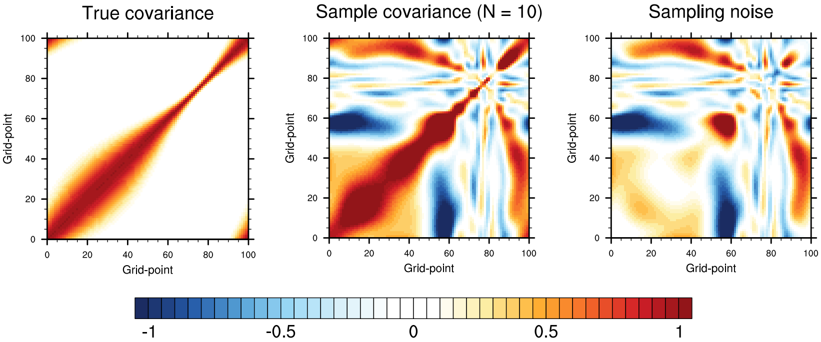
Heterogeneous variance / homogeneous length-scale



Sampling noise amplitude related to the asymptotic variance

Sampling noise properties

Homogeneous variance / heterogeneous length-scale



Sampling noise length-scale related to the asymptotic length-scale

How to optimize localization and hybridization?

Optimal localization alone (without hybridization):

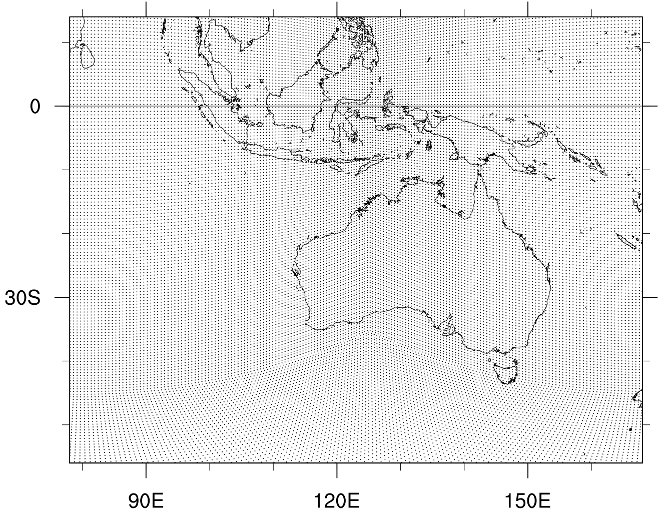
$$L'_{ij} = \frac{\mathbb{E}\left[B_{ij}^2\right]}{\mathbb{E}\left[\widetilde{B}_{ij}^2\right]}$$

$$= \frac{(N-1)^2}{N(N-3)} + \frac{N-1}{N(N-2)(N-3)} \frac{\mathbb{E}\left[\widetilde{B}_{ii}\widetilde{B}_{jj}\right]}{\mathbb{E}\left[\widetilde{B}_{ij}^2\right]} - \frac{N}{(N-2)(N-3)} \frac{\mathbb{E}\left[\widetilde{\Xi}_{ijij}\right]}{\mathbb{E}\left[\widetilde{B}_{ij}^2\right]}$$

where $\widetilde{\Xi}$ is the sample fourth-order centered moment.

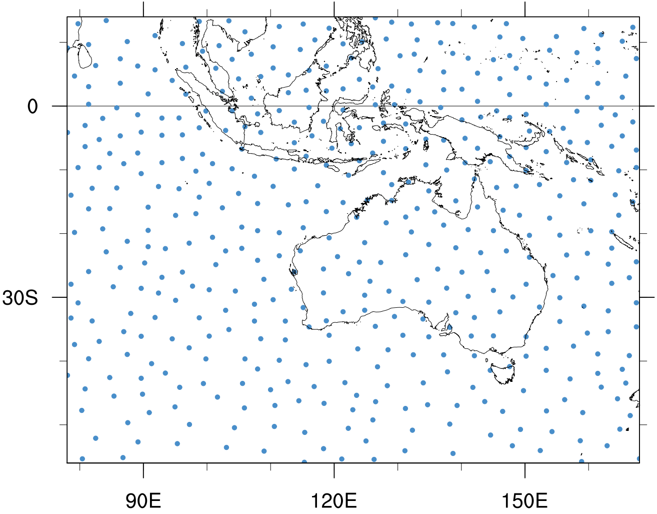
$$\beta^{s2} = \frac{\sum_{ij}\left(1 - L'_{ij}\right)\mathbb{E}\left[\widetilde{B}_{ij}\right]B_{ij}^s}{\sum_{ij}\frac{\mathrm{Var}\left[\widetilde{B}_{ij}\right]}{\mathbb{E}\left[\widetilde{B}_{ij}^2\right]}B_{ij}^{s2}} \quad \text{and} \quad \beta^{e2}L_{ij} = L'_{ij} - \frac{\mathbb{E}\left[\widetilde{B}_{ij}\right]}{\mathbb{E}\left[\widetilde{B}_{ij}^2\right]}\beta^{s2}B_{ij}^s$$

Homogenous and isotropic sampling



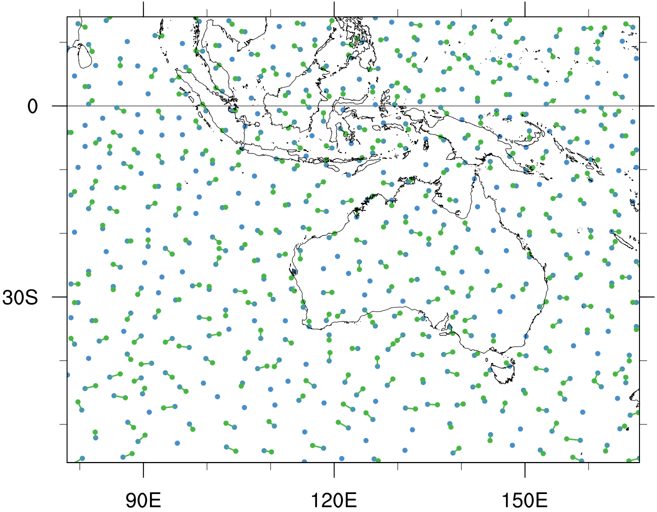
Full model grid (FV3 cubed-sphere grid - C180)

Homogenous and isotropic sampling



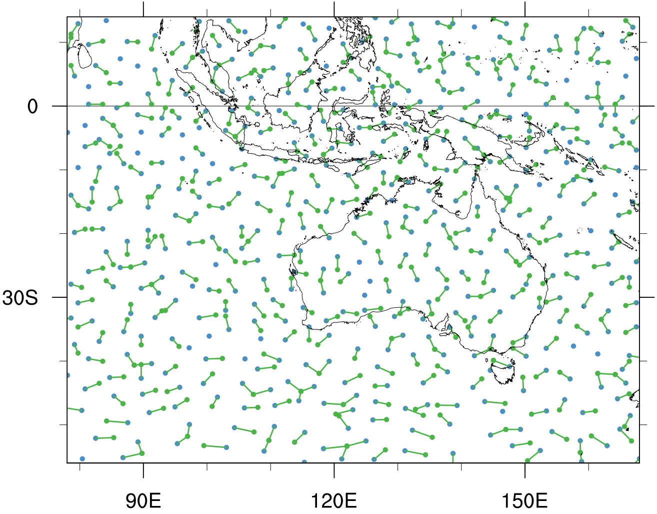
Homogenous subsampling for origin points

Homogenous and isotropic sampling



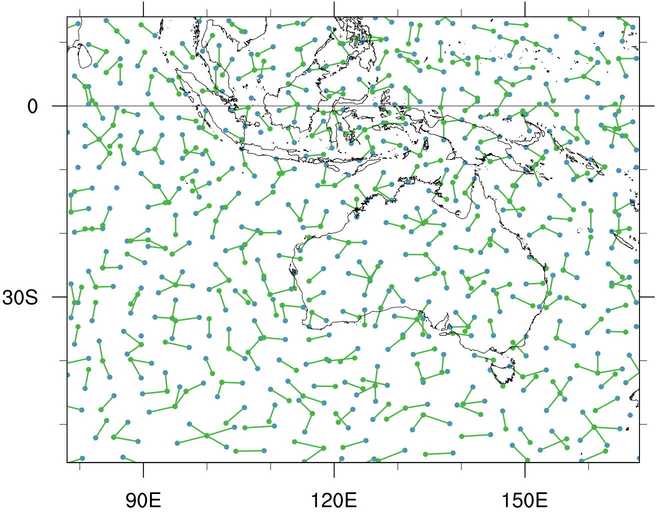
Isotropic subsampling for distant points (increasing distance class)

Homogenous and isotropic sampling



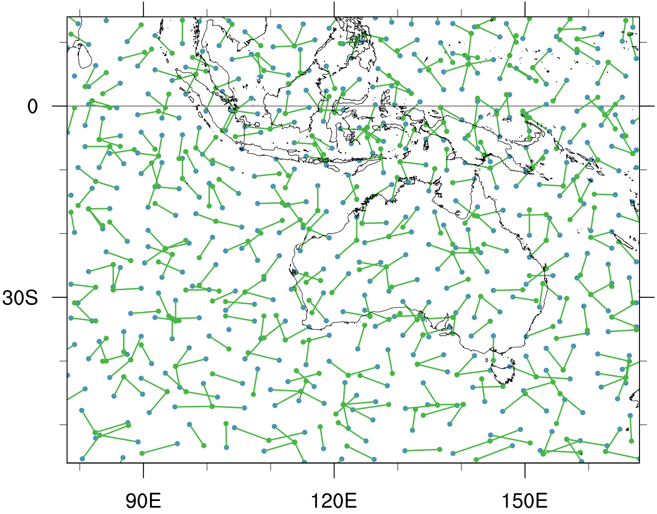
Isotropic subsampling for distant points (increasing distance class)

Homogenous and isotropic sampling



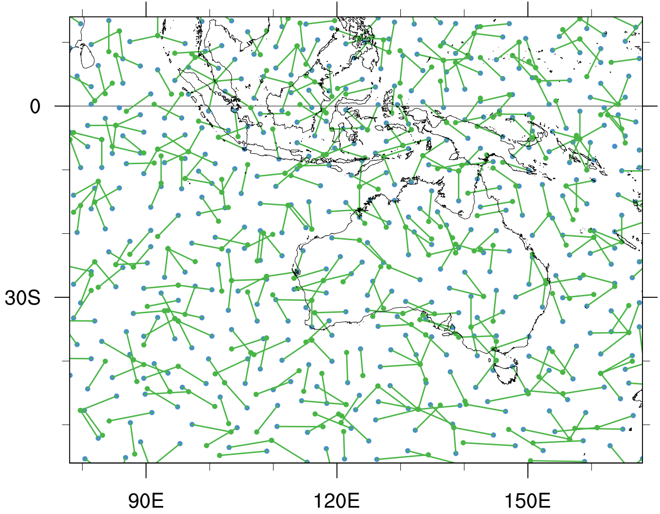
Isotropic subsampling for distant points (increasing distance class)

Homogenous and isotropic sampling



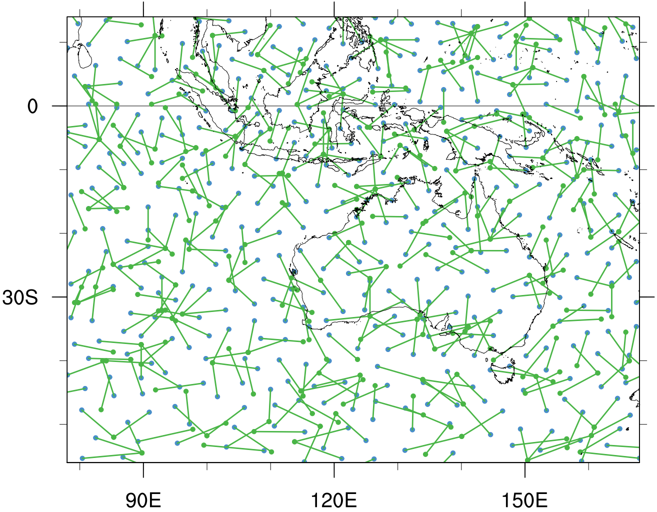
Isotropic subsampling for distant points (increasing distance class)

Homogenous and isotropic sampling



Isotropic subsampling for distant points (increasing distance class)

Homogenous and isotropic sampling



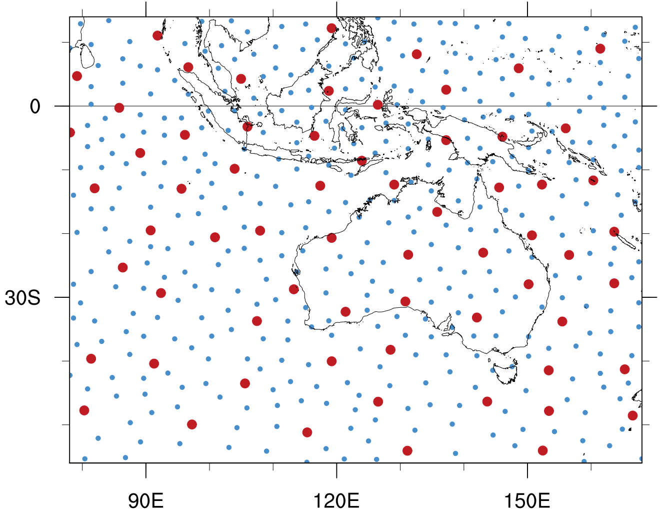
Isotropic subsampling for distant points (increasing distance class)

Ergodicity assumption

For each couple of points, BUMP uses the ensemble to estimate:

- the sample variances $\widetilde{B}_{ii}$ and $\widetilde{B}_{jj}$
- the sample covariance $\widetilde{B}_{ij}$
- the sample fourth-order centered moment $\widetilde{\Xi}_{ijij}$

Local averaging



Local averaging centers (red points)

Ergodicity assumption

For each couple of points, BUMP uses the ensemble to estimate:

- the sample variances $\widetilde{B}_{ii}$ and $\widetilde{B}_{jj}$
- the sample covariance $\widetilde{B}_{ij}$
- the sample fourth-order centered moment $\widetilde{\Xi}_{ijij}$

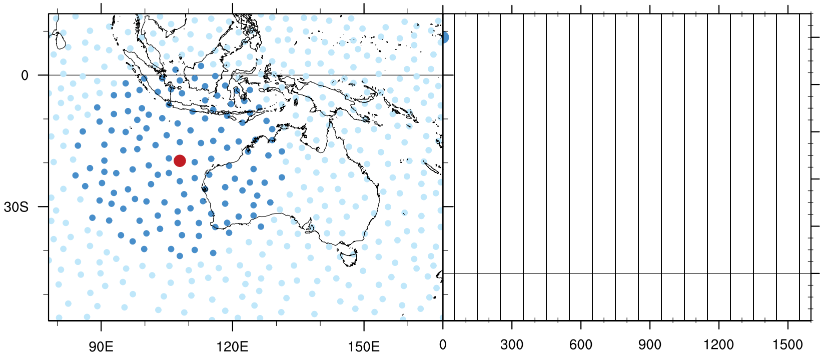
With the spatial and angular ergodicity assumption, these values are averaged locally for each distance class, to get estimations of the following expectations:

$$\mathbb{E}\left[\widetilde{B}_{ii}\widetilde{B}_{jj}\right] \quad , \quad \mathbb{E}\left[\widetilde{B}_{ij}^2\right] \quad \text{and} \quad \mathbb{E}\left[\widetilde{\Xi}_{ijij}\right]$$

These quantities are useful to compute the localization function and hybridization weights with the previous formulas.

Local averaging

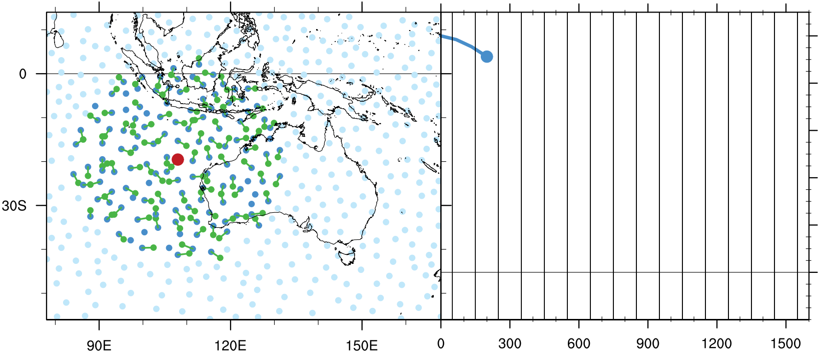
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

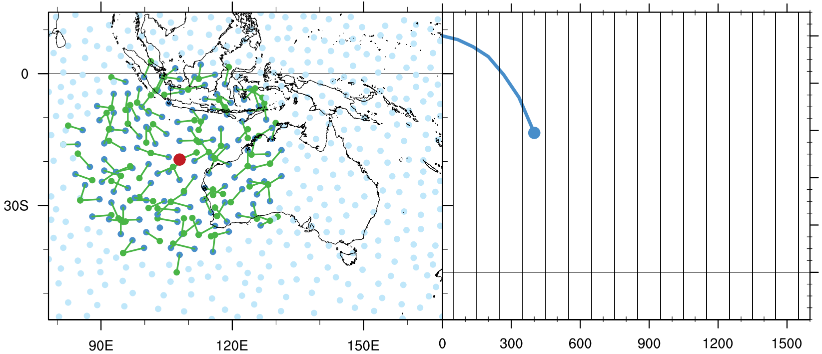
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

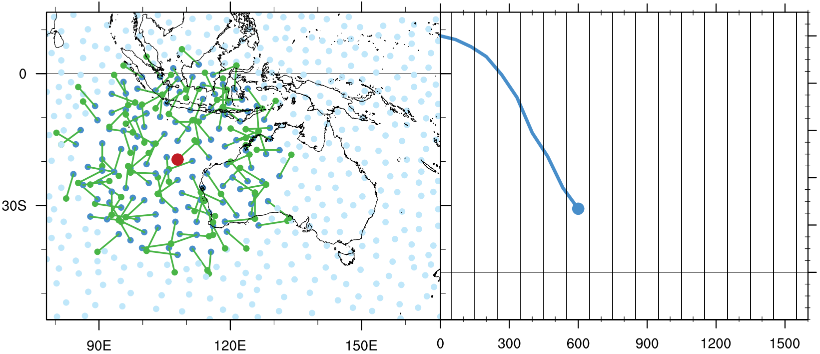
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

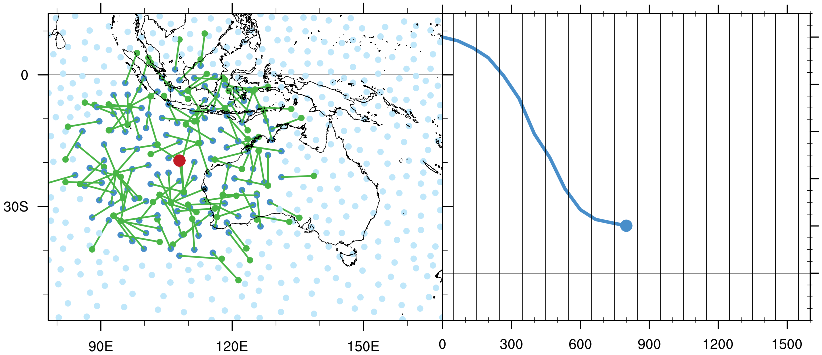
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

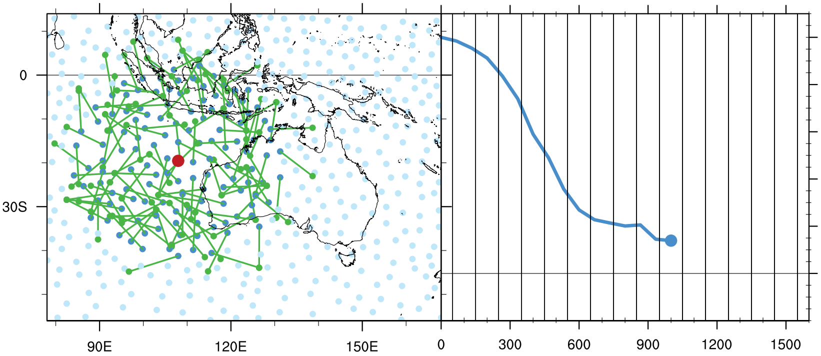
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

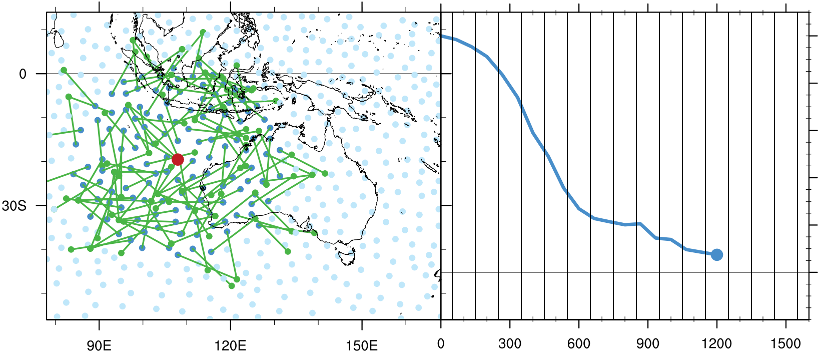
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

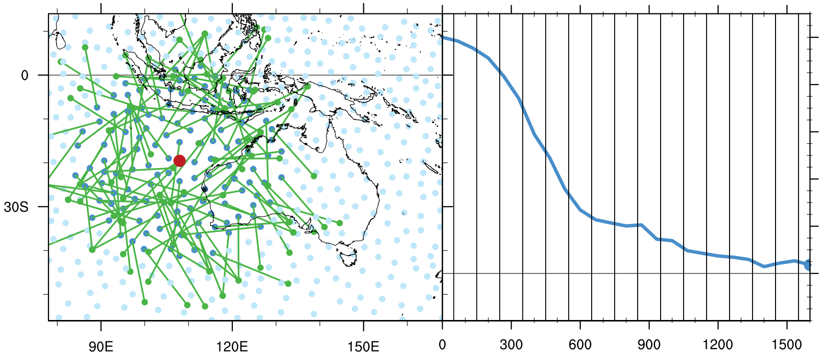
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Local averaging

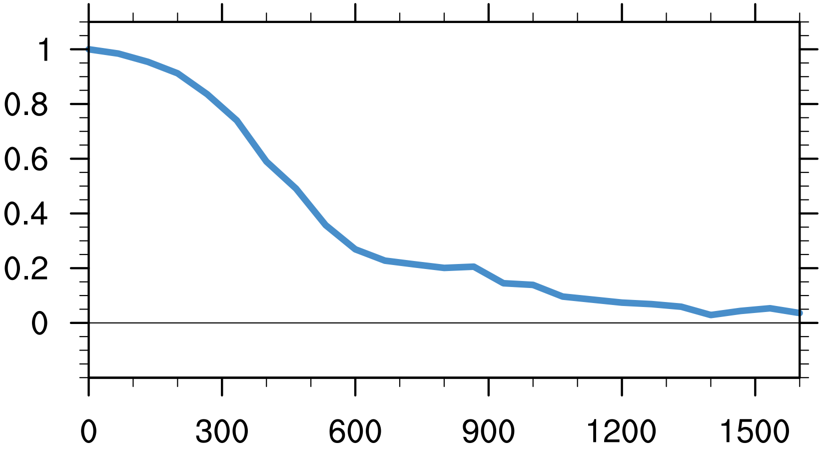
For a given local averaging center:



Averaged couples for a given local averaging center (left)
Diagnosed localization as a function of distance in km (right)

Localization function fit

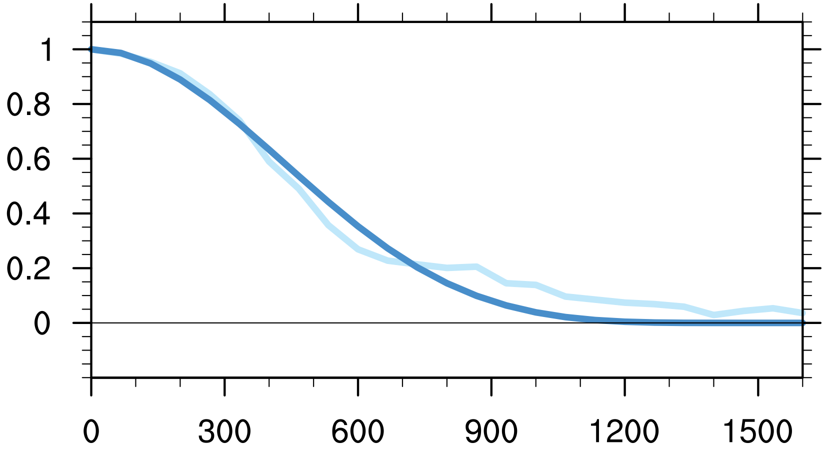
For a given local averaging center:



Raw localization as a function of distance (km)

Localization function fit

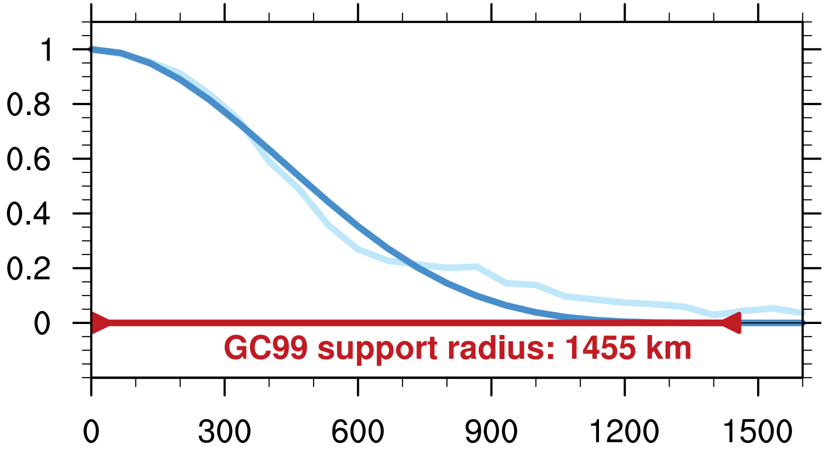
For a given local averaging center:



Fit with a Gaspari-Cohn (1999) function

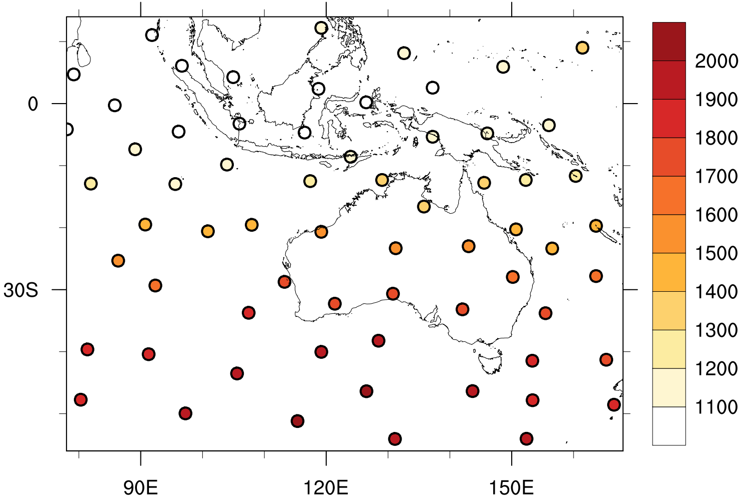
Localization function fit

For a given local averaging center:



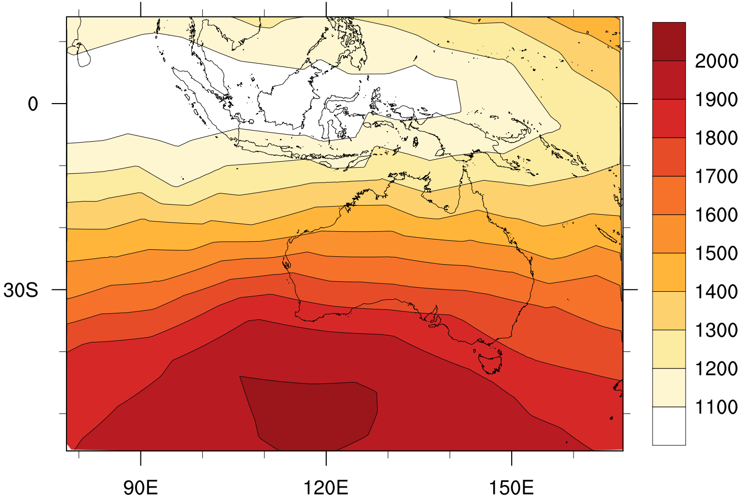
Localization length-scale (support radius actually, km)

Support radius interpolation



Localization support radius (km) at local averaging centers

Support radius interpolation

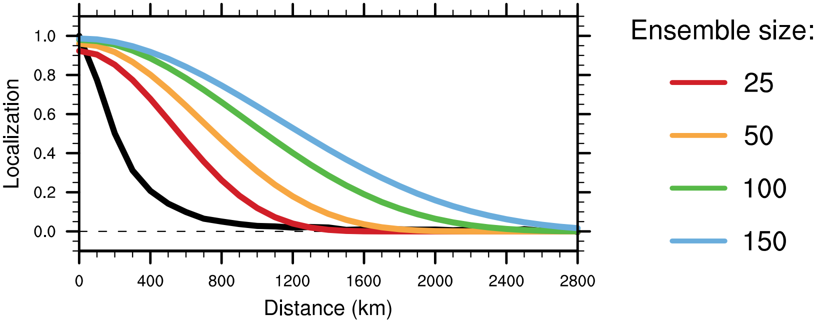


Localization support radius (km) interpolated on the model grid

HDIAG process

1. Define a homogeneous subsampling of the model grid (origin points).
2. For each origin point, find distant points for a series of distance classes.
3. For each local averaging center (a subset of origin points):
   - Average diagnostic values over a given radius.
   - Compute the localization and hybridization weights using the Ménétrier et al. (2015) formula.
   - Fit the raw localization function with a parametrized function (e.g. Gaspari-Cohn, 1999) to get the localization support radius.
4. Interpolate the localization support radius and hybridization weights over the model grid.

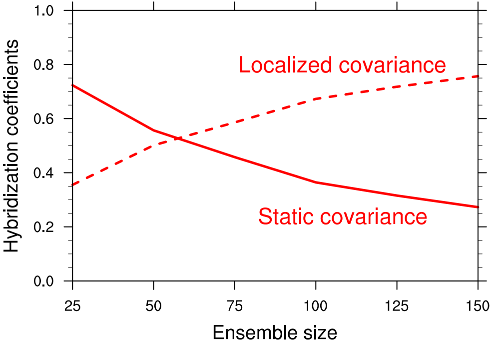
Ensemble size sensitivity



Correlation (black) et localization (colors) for various ensemble sizes

Localization length-scale increases as the ensemble size increases
(less sampling noise to remove)

Ensemble size sensitivity



Ensemble and static weights as a function of the ensemble size

Less weight on the static covariance as the ensemble size increases (less deficiencies to correct).

Summary

- The sample covariance is affected by sampling noise.
- This sampling noise decreases if the ensemble size increases.
- Using a huge ensemble is too costly for operational applications.
- Localization and hybridization can be used to filter out the sampling noise.
- The asymptotic sample covariance is the filtering target.
- We combine the linear filtering theory and the centered moments sampling theory.
- This leads to practicable formulas for optimal localization and hybridization weights.
- We compute robust estimates using ergodicity assumptions.
- These statistics can be computed globally, or locally.
- Thus, we get global, or local estimates of the optimal localization and hybridization weights.

Outline

Introduction

Static **B**

Ensemble/hybrid **B**

The NICAS smoother

BUMP usage

Explicit convolution

Main goal: designing a generic method to apply a normalized convolution operator **on any grid type**.

Standard methods:

- Spectral/wavelet transforms $\rightarrow$ regular grid required
- Recursive filters $\rightarrow$ regular grid required + normalization issue
- Explicit/implicit diffusion $\rightarrow$ potentially high cost + normalization issue

Advantages of an explicit convolution $\mathbf{C}$:

- Work on any grid type
- Exact normalization ($C_{ii} = 1$)

Drawback: the computational cost scales as $O(n^2)$, where n is the size of the model grid...

Explicit convolution

To limit the computational cost, we approximate **C** on a subgrid (subset of n^s points of the model grid):

$$\mathbf{C} \approx \mathbf{S}\mathbf{C}^s\mathbf{S}^\mathrm{T}$$

where

- **S** is an interpolation from the subgrid to the model grid
- $\mathbf{C}^s$ is a convolution matrix on the subgrid

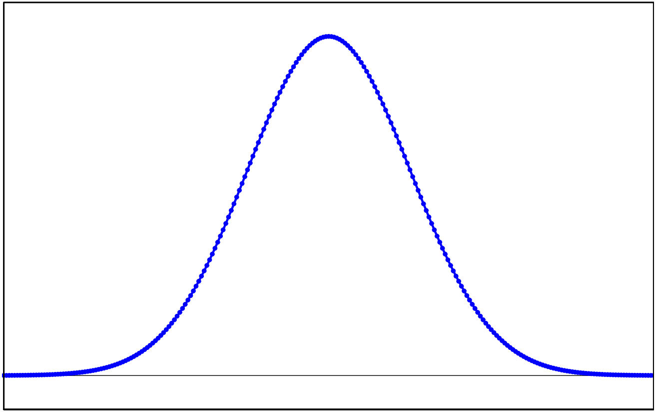
If $n^s \ll n$, then the total cost scales as $O(n)$ (interpolation cost).

Issues with this approach:

- If the subgrid density is too coarse compared to the convolution length-scale, the convolution is distorded.
- Normalization breaks down because of the interpolation: even if $\mathbf{C}^s$ is normalized, $\mathbf{S}\mathbf{C}^s\mathbf{S}^\mathrm{T}$ is not.

Convolution on a subgrid

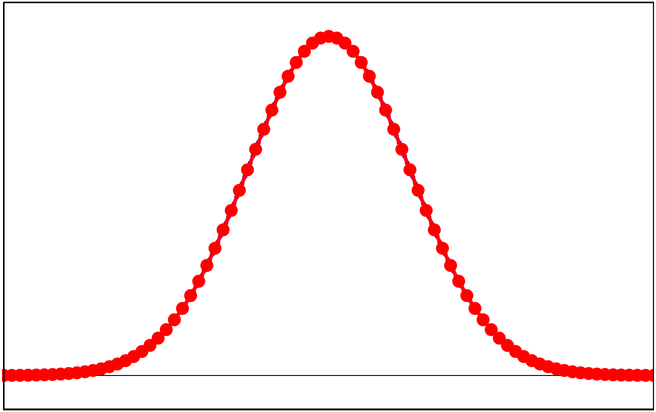
Convolution function on model grid



Model grid (blue)
Large convolution length-scale

Convolution on a subgrid

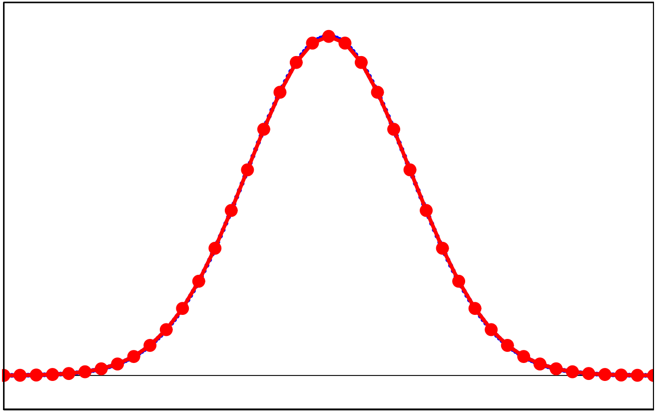
Subsampling: 1 point over 3



Model grid (blue) and subgrid (red)
Large convolution length-scale

Convolution on a subgrid

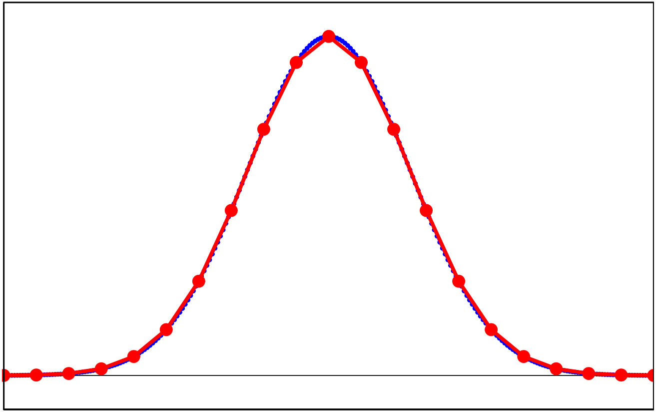
Subsampling: 1 point over 6



Model grid (blue) and subgrid (red)
Large convolution length-scale

Convolution on a subgrid

Subsampling: 1 point over 12

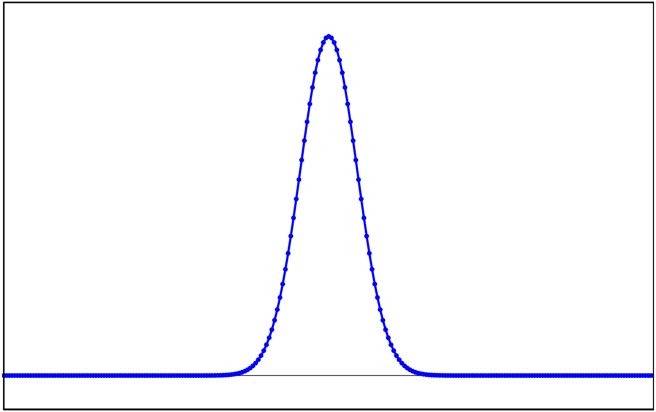


Model grid (blue) and subgrid (red)

Large convolution length-scale

Convolution on a subgrid

Convolution function on model grid

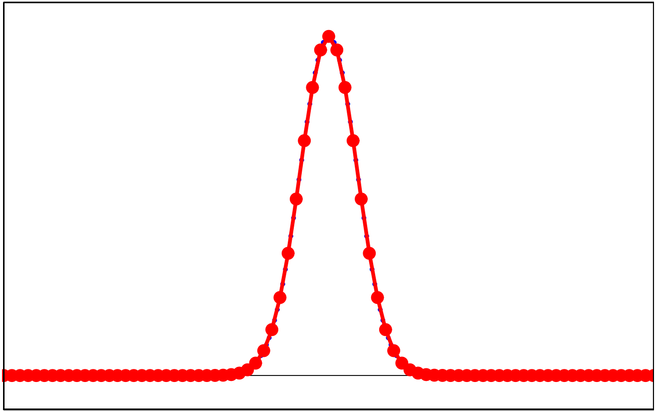


Model grid (blue)

Small convolution length-scale

Convolution on a subgrid

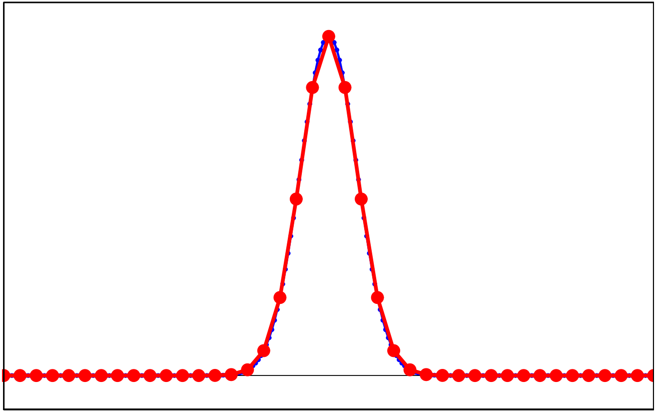
Subsampling: 1 point over 3



Model grid (blue) and subgrid (red)
Small convolution length-scale

Convolution on a subgrid

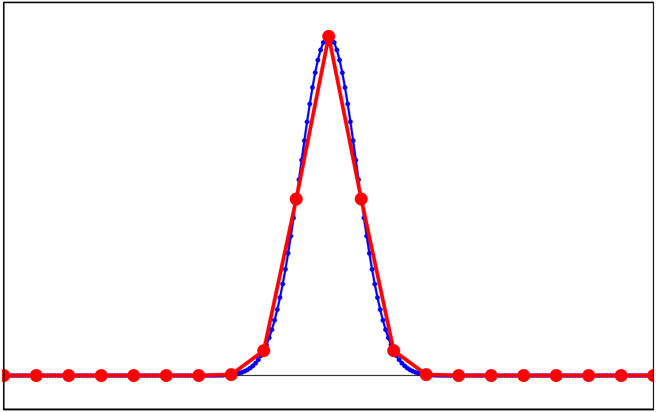
Subsampling: 1 point over 6



Model grid (blue) and subgrid (red)
Small convolution length-scale

Convolution on a subgrid

Subsampling: 1 point over 12



Model grid (blue) and subgrid (red)

Small convolution length-scale

Explicit convolution

The NICAS method (Normalized Interpolated Convolution from an Adaptive Subgrid) is given by:

$$\widetilde{\mathbf{C}} = \mathbf{N}\mathbf{S}\mathbf{C}^s\mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}$$

where

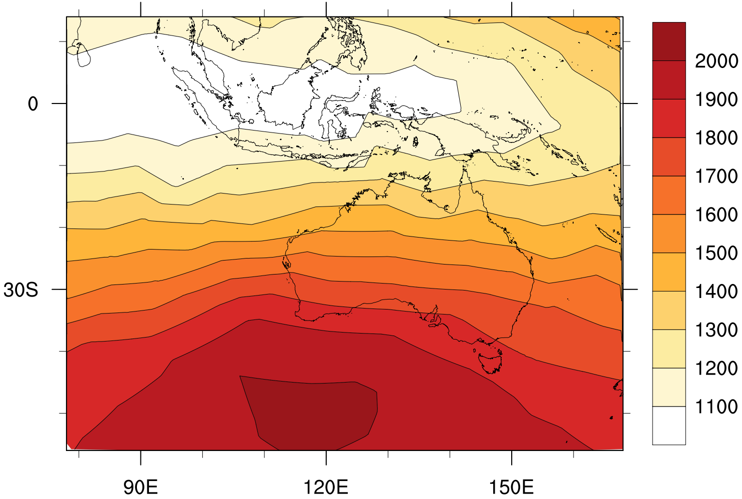
- $\mathbf{N}$ is a diagonal normalization matrix.
- The subgrid is locally adapted to the convolution length-scale.

To illustrate how NICAS works:

- Example of adaptive subgrid.
- Steps of a Dirac test: apply $\widetilde{\mathbf{C}}$ to a vector δ^k where

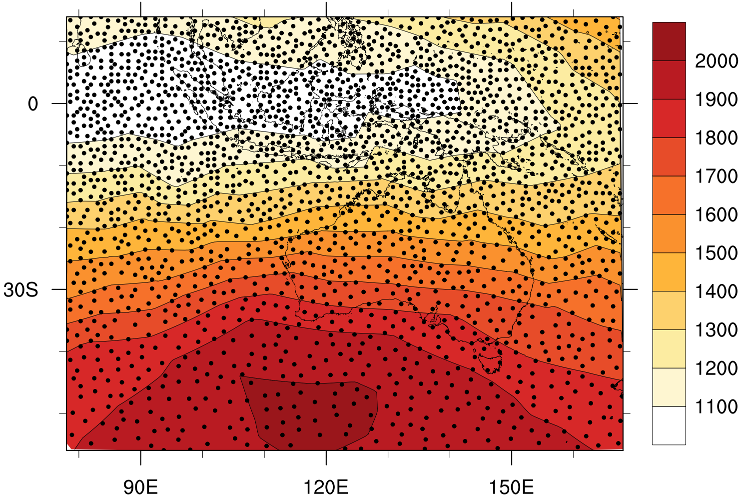
$$\delta_i^k = \begin{cases} 1 & \text{if} \quad i = k \\ 0 & \text{if} \quad i \neq k \end{cases}$$

Adaptive subgrid



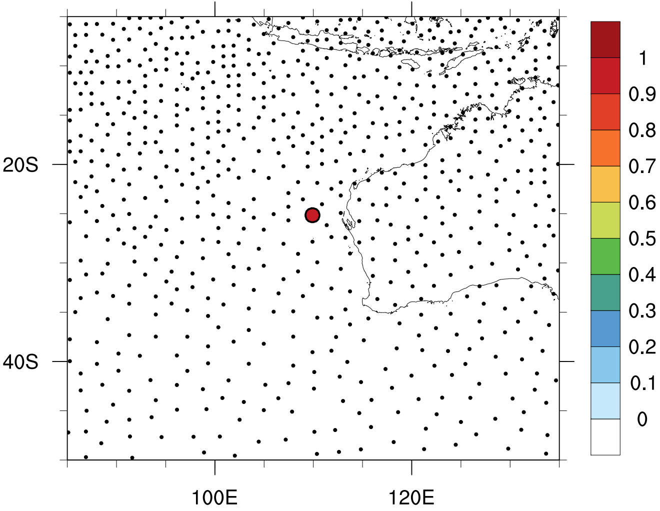
Localization support radius (km) interpolated on the model grid

Adaptive subgrid



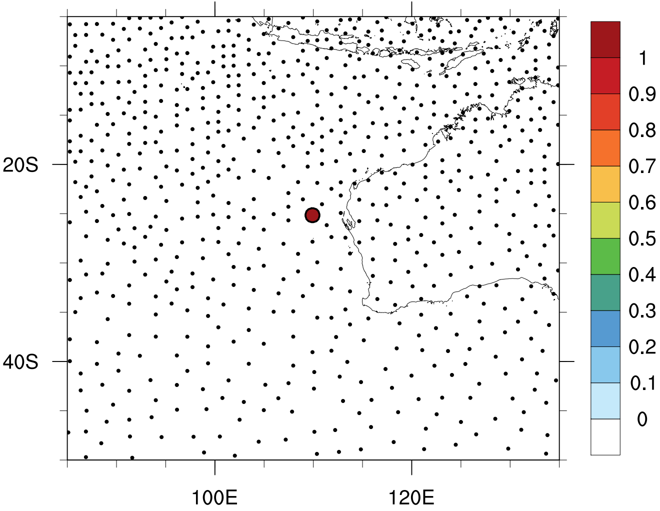
Adaptive support radius-based subgrid

NICAS steps



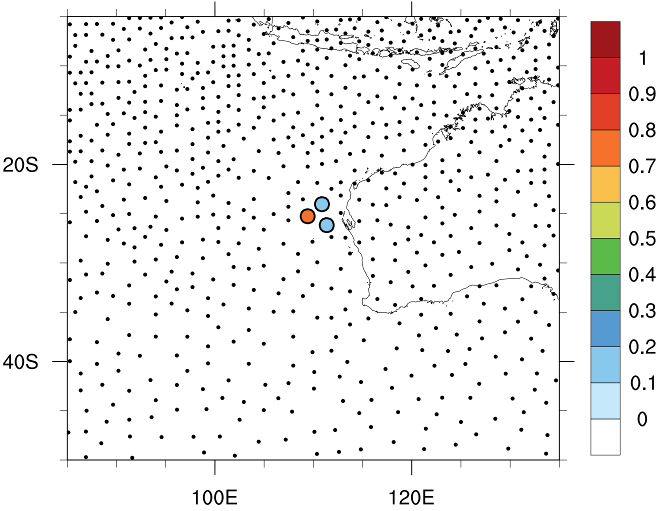
Initial vector: δ^k

NICAS steps



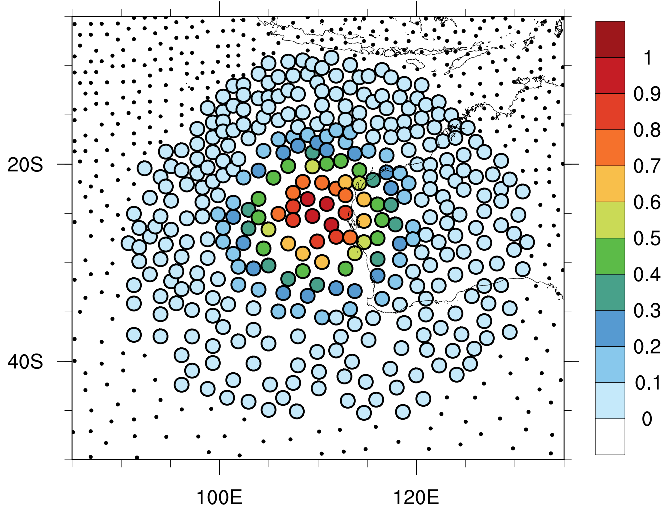
Adjoint normalization: $\mathbf{N}^{\mathrm{T}} \delta^k$

NICAS steps



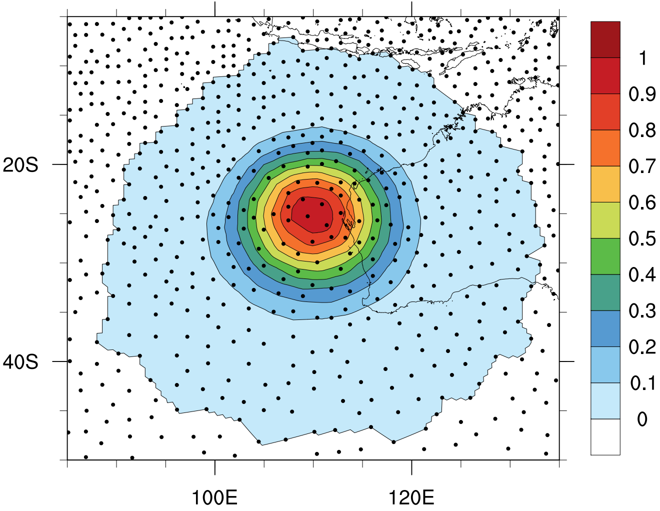
Adjoint interpolation: $\mathbf{S}^{\mathrm{T}}\mathbf{N}^{\mathrm{T}}\delta^{k}$

NICAS steps



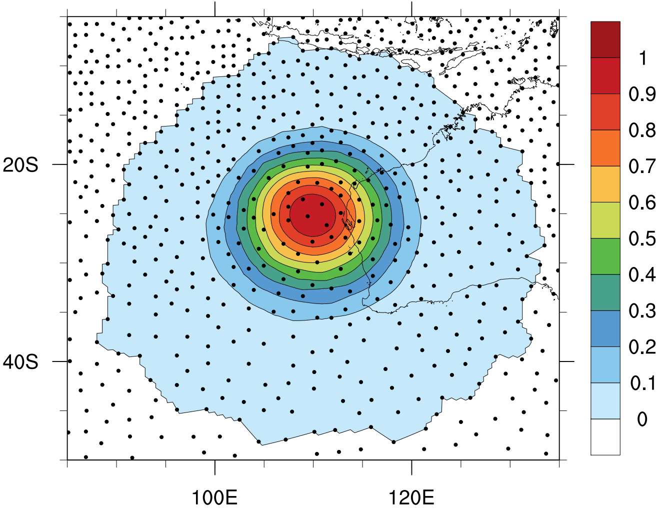
Convolution: $\mathbf{C}^s \mathbf{S}^{\mathrm{T}} \mathbf{N}^{\mathrm{T}} \delta^k$

NICAS steps



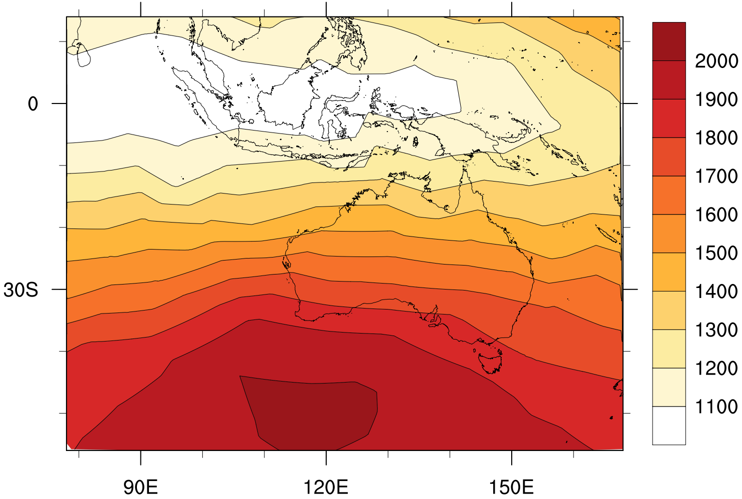
Interpolation: $\mathbf{S}\mathbf{C}^s\mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}\delta^k$

NICAS steps



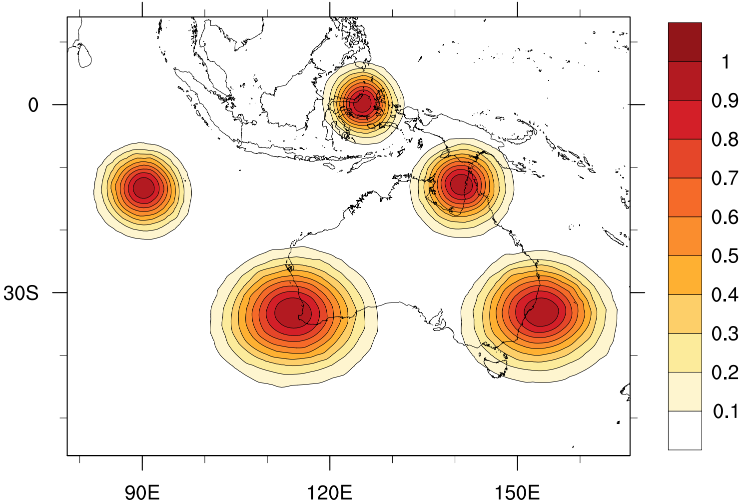
Normalization: $\mathbf{N}\mathbf{S}\mathbf{C}^s\mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}\delta^k$

Result of the Dirac test



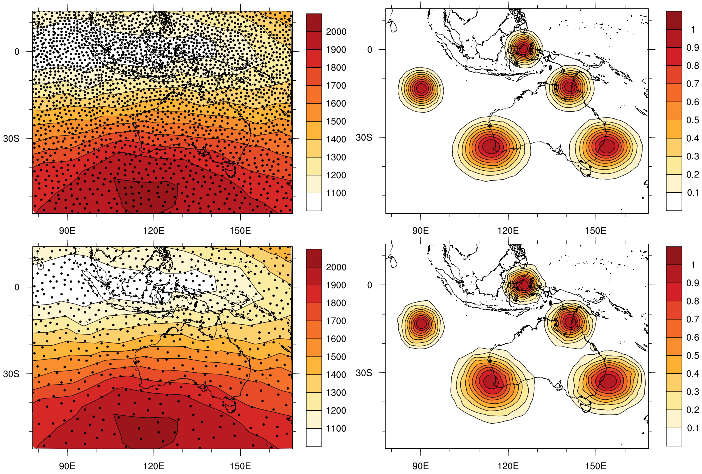
Localization support radius (km)

Result of the Dirac test



Application of NICAS on Dirac functions

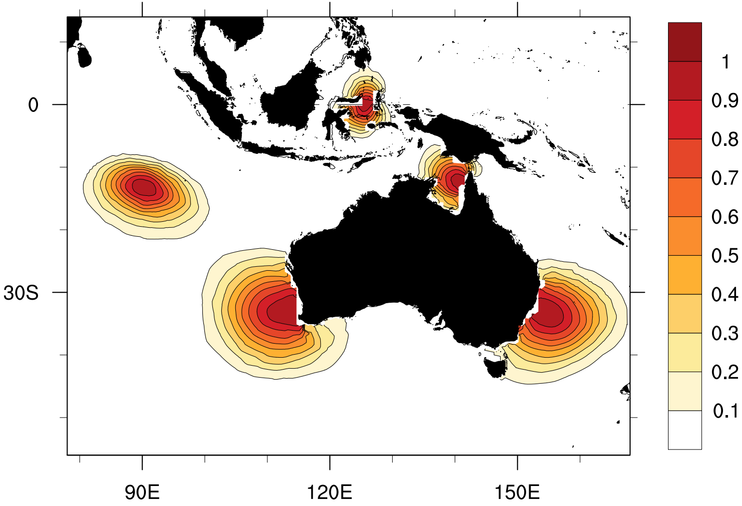
Impact of the subgrid resolution



A scalar parameter controls the subgrid resolution.

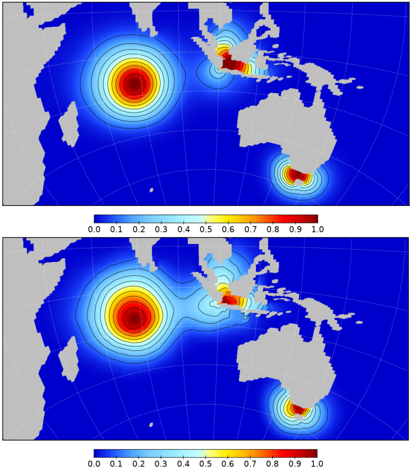
Simple trade-off between cost and accuracy.

Complex boundaries



Application of masked anisotropic NICAS on Dirac functions

Complex boundaries



On the ORCA1 grid of NEMOVAR: implicit diffusion (top) and NICAS (bottom, 50x faster)

Square-root formulation

- Basic NICAS method:

$$\widetilde{\mathbf{C}} = \mathbf{N}\mathbf{S}\mathbf{C}^s\mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}$$

- If $\mathbf{C}^s$ is built as $\mathbf{U}^s\mathbf{U}^{s\mathrm{T}}$, then the square-root of $\widetilde{\mathbf{C}}$ is given by:

$$\widetilde{\mathbf{U}} = \mathbf{N}\mathbf{S}\mathbf{U}^s$$

  which can be useful for square-root preconditioning in variational methods.

- Using the formulation:

$$\widetilde{\mathbf{C}} = \mathbf{N}\mathbf{S}\mathbf{U}^s\mathbf{U}^{s\mathrm{T}}\mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}$$

  also ensures that $\widetilde{\mathbf{C}}$ is positive-semidefinite.

- A good approximation of the Gaspari and Cohn (1999) function square-root can be obtained by multiplying the function length-scale by an empirical scalar factor.

MPI communications

Running NICAS with several MPI tasks:

- Communications are always performed **on the subgrid**, never on the model grid.
- Only **local** communications between halos are required, no global communications.
- NICAS can be applied with 1, 2 or 3 communication steps:

$$\widetilde{\mathbf{C}} = \mathbf{N}\mathbf{S} \boxtimes \mathbf{U}^s\mathbf{U}^{s\mathrm{T}}\mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}$$

$$\widetilde{\mathbf{C}} = \mathbf{N}\mathbf{S} \boxtimes \mathbf{U}^s\mathbf{U}^{s\mathrm{T}} \boxtimes \mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}$$

$$\widetilde{\mathbf{C}} = \mathbf{N}\mathbf{S} \boxtimes \mathbf{U}^s \boxtimes \mathbf{U}^{s\mathrm{T}} \boxtimes \mathbf{S}^\mathrm{T}\mathbf{N}^\mathrm{T}$$

  More communication steps $\Rightarrow$ smaller halos.

- Hybrid parallelization with OpenMP is used to improve efficiency.

Outline

Introduction

Static **B**

Ensemble/hybrid **B**

The NICAS smoother

BUMP usage

BUMP usage in OOPS

- BUMP is fully interfaced with OOPS.
- The BUMP parameters are set from the YAML file.
- Default parameters and short descriptions can be found in bump/type_nam.F90
- Documentation can be found on GitHub page: `https://github.com/JCSDA/saber`
- Support using the GitHub page or my email: `benjamin.menetrier@irit.fr`
- To get started, some examples of BUMP usage for a static B and for an ensemble/hybrid B.

Static B with BUMP: example 1

NICAS smoother (for correlation) with a prescribed support radius

```
bump:
  datadir: bump                       # Bump-specific directory
  forced_radii: 1                     # Forced length-scale
  method: cor                         # Static correlation
  mpicom: 2                           # NICAS communication steps
  new_nicas: 1                        # New NICAS smoother
  ntry: 10                            # Subsampling quality
  prefix: your_experiment             # BUMP files base
  resol: 8.0                          # NICAS Subgrid resolution
  rh: 1000.0e3                        # Horizontal support radius
  rv: 1000.0                          # Vertical support radius
  strategy: specific_univariate       # Multivariate strategy
```

Important keys you might change:

- The horizontal support radius (in m): `rh`
- The vertical support radius (in your vertical unit): `rv`
- The NICAS subgrid resolution: `resol`

Static B with BUMP: example 2

Correlation radius estimation with HDIAG

```
bump:
  datadir:bump                 # Bump-specific directory
  dc: 100.0e3                  # Diagnostic bins size
  method: cor                  # Static correlation
  nc1: 500                     # Diagnostic subsampling size
  nc3: 20                      # Number of diagnostic bins
  ne: 50                       # Ensemble size
  new_hdiag: 1                 # New HDIAG diagnostic
  nl0r: 11                     # Number of diagnostic levels
  ntry: 10                     # Subsampling quality
  prefix: your_experiment      # BUMP files base
  strategy: specific_univariate # Multivariate strategy
```

Important keys you might change:

- The diagnostic horizontal bin size (in m): `dc`
- The diagnostic subsampling size: `nc1`
- The number of diagnostic bins: `nc3`

Static B with BUMP: example 3

Use HDIAG diagnostic in NICAS, 1 step

```
bump:
  datadir:bump                       # Bump-specific directory
  dc: 100.0e3                        # Diagnostic bins size
  method: cor                        # Static correlation
  mpicom: 2                          # NICAS communication steps
  nc1: 500                           # Diagnostic subsampling size
  nc3: 20                            # Number of diagnostic bins
  ne: 50                             # Ensemble size
  new_hdiag: 1                       # New HDIAG diagnostic
  new_nicas: 1                       # New NICAS smoother
  nl0r: 11                           # Number of diagnostic levels
  ntry: 10                           # Subsampling quality
  prefix: your_experiment            # BUMP files base
  resol: 8.0                         # NICAS subgrid resolution
  strategy: specific_univariate # Multivariate strategy
```

In the same run:

- Run HDIAG to estimate correlation radius
- Use it to set up the NICAS smoother

Static B with BUMP: example 4

Use HDIAG diagnostic in NICAS, 2 steps

```
bump:
  datadir: bump                     # Bump-specific directory
  load_cmat: 1                      # Load radius data from HDIAG
  method: cor                       # Static correlation
  mpicom: 2                         # NICAS communication steps
  new_nicas: 1                      # New NICAS smoother
  ntry: 10                          # Subsampling quality
  prefix: your_experiment           # BUMP files base
  resol: 8.0                        # NICAS subgrid resolution
  strategy: specific_univariate # Multivariate strategy
```

In the two runs:

- First run: estimate correlation radius with HDIAG (example 2)
- Second run: load HDIAG data (`load_cmat: 1`) to set up the NICAS smoother (this example)

Warning: the `prefix` key must be the same!

Ensemble/hybrid B with BUMP

All the previous examples can be reused for the localization of an ensemble B, with only two keys to update:

- Method for localization: `method: loc`
- Multivariate strategy: `strategy: common`
  Other multivariate strategies are available in BUMP.

For the hybrid B the method is set at: `method: hyb-rnd` and a randomized pseudo-ensemble must be provided.

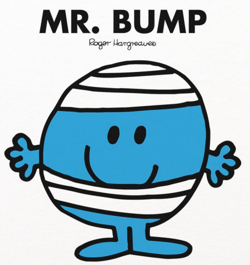
Full YAML files examples are provided with the QG model.

Other advanced BUMP features (anisotropic correlations, 4D localization, etc.) can also be activated from the YAML file.

BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr
Funded by the JCSDA

4^{th} JEDI Academy - February 2020

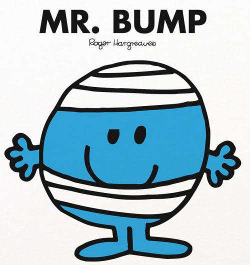


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

4^{th} JEDI Academy - February 2020

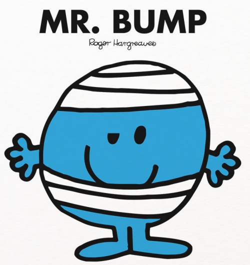


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

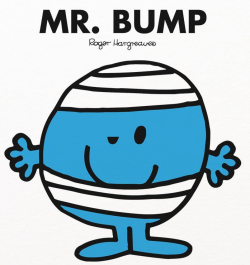
4^{th} JEDI Academy - February 2020



BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr
Funded by the JCSDA

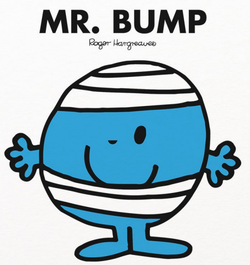
4th JEDI Academy - February 2020



BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr
Funded by the JCSDA

4th JEDI Academy - February 2020

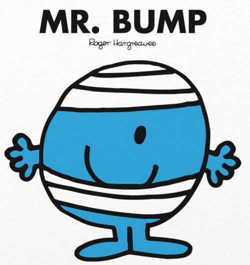


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

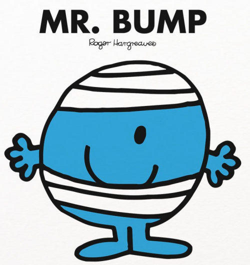
4^{th} JEDI Academy - February 2020



BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr
Funded by the JCSDA

4^{th} JEDI Academy - February 2020

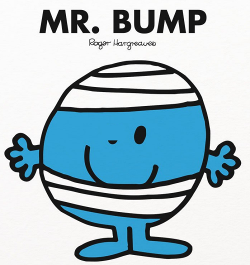


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

4^{th} JEDI Academy - February 2020

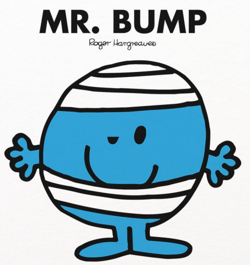


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

4^{th} JEDI Academy - February 2020

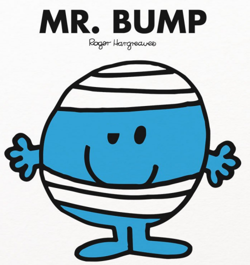


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

4th JEDI Academy - February 2020

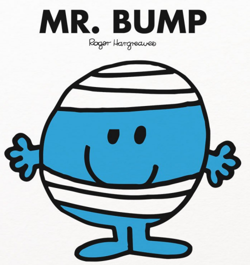


BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

4th JEDI Academy - February 2020



BUMP, a generic tool for background error covariance modeling

Benjamin Menetrier, IRIT, Toulouse, France
benjamin.menetrier@irit.fr

Funded by the JCSDA

4th JEDI Academy - February 2020