Progress in developing a 3D background error correlation model using an implicit diffusion operator

Isabelle Mirouze[1] et Anthony T. Weaver[2]

[1] *CERFACS / CNRS-UMR 5219*
[2] *CERFACS / SUC URA 1875*

VODA: intermediary meeting.
Grenoble 9 and 10 February 2010

Summary

Summary

1. Introduction
2. The 1D diffusion operator
3. Extension to 2D and 3D correlation functions: implementation in NEMOVAR
4. Conclusions

Context: variational assimilation

Goal: Estimate $\mathbf{B}\boldsymbol{\eta}$, with $\mathbf{B}$ the background error covariance matrix and $\boldsymbol{\eta}$ a scalar field.

Practical difficulties:

- Estimate: lack of information
- Multivariate: covariances between variables exist
- Huge size: the matrix cannot be stored nor manipulated
- $\mathbf{B}$ must be symmetric and positive definite: $\mathbf{B} = \mathbf{U}\mathbf{U}^{\mathrm{T}}$

Possibilities:

- Physical insight
- Limit the number of statistically tunable parameters
- Central limit theorem $\Rightarrow$ Assume Gaussian structure

Constraints:

- Need an efficient algorithm
- Access to the "square root"

Different techniques to process different parts

- Multivariate problem
  - Balance operator **K**: transform the multivariate problem in several independent univariate problems
- Spatial covariances for a given variable
  - Diagonal matrix of the variances **D** and correlations matrix **C**
- How to construct **C**?
  - *Spectral space:* diagonal hypothesis for the isotropic case
    - ★ Estimate of **C** with an ensemble method + wavelets filtering or localization, ...
  - *Physical space:* handle more easily complex boundaries
    - ★ Model **C** with operators

      $\Rightarrow$ Evaluate (convolution) integrals

      $\longrightarrow$ Compact support functions

      $\Rightarrow$ Find an alternative method

      $\longrightarrow$ Recursive filter

      $\longrightarrow$ Diffusion equation

      $\longrightarrow$ ...

Method following Weaver and Courtier (2001) and others

1D diffusion equation with initial condition $\eta(z, t_0) = \eta_0(z)$

$$\frac{\partial \eta}{\partial t}(z,t) - \kappa \frac{\partial^2 \eta}{\partial z^2}(z,t) = 0 \qquad \kappa > 0$$

Solution on infinite domain ($z \in \mathbb{R}$) for an integration from t_0 to t_M

$$\eta(z, t_M) = \frac{1}{\sqrt{4\pi\kappa t_M}} \int_{-\infty}^{\infty} e^{-\frac{(z-z')^2}{4\kappa t_M}} \eta_0(z')\, dz'$$

Integrating the constant-coefficient 1D diffusion equation from t_0 to $t_M = M\Delta t$, and normalizing the result (maximum is 1),

$$\eta(z, t_M) = \lambda_M \, \mathcal{L}^M \left[\eta_0(z)\right] = \mathcal{F}_M \left[\eta_0(z)\right]$$

is equivalent to applying a correlation operator

$$\eta(z, t_M) = \int_{-\infty}^{\infty} f_M(z-z')\, \eta_0(z')\, dz'$$

where $f_M(z-z')$ is the kernel (correlation function).

Constructing grid-point correlation models using a diffusion operator

- Why a diffusion operator?
  - Grid-point filters such as diffusion are more convenient than spectral filters for ocean models (easier to handle boundaries).
  - Diffusion operators are self-adjoint and positive definite by construction.
  - Access to the "square root" by dividing the integration time by 2.
  - Analytical solutions are known for the homogeneous and isotropic case.
  - Can be made inhomogeneous and anisotropic.
- What is the main disadvantage?
  - Can be expensive, especially explicit formulations.
    - Many iterations may be required when the local length scale is large relative to the local grid size.
  - Formulations based on an implicit scheme are preferable for general parameter settings
    - A matrix inversion is required.
- 2 main points to focus on:
  - How to solve the diffusion equation for the implicit formulation.
  - How to normalize the diffusion operator to ensure unit amplitude.

Summary

Explicit / Implicit for the constant-coefficient 1D diffusion equation

	Explicit scheme	Implicit scheme
Operator	$\mathcal{L} \equiv 1 + \kappa \Delta t \frac{\partial^2}{\partial z^2}$	$\mathcal{L}^{-1} \equiv 1 - \kappa \Delta t \frac{\partial^2}{\partial z^2}$
Iter.	Stability criterion	Free
Normal.	$\lambda = \sqrt{4\pi\kappa M \Delta t}$	$\lambda_M = \upsilon_M \sqrt{\kappa \Delta t}$
Kernel	$e^{-\frac{(z-z')^2}{2L^2}}$	$\left[1+\sum_{j=1}^{M-1} \beta_j \left(\frac{\lvert z-z'\rvert}{L}\right)^j\right] e^{-\frac{\lvert z-z'\rvert}{L}}$
Scale prm.	$L = \sqrt{2\kappa M \Delta t}$	$L = \sqrt{\kappa \Delta t}$
Length sc.	$L_{\mathrm{g}} = L$	$L_{\mathrm{AR}} = (2M-3)^{\frac{1}{2}} L \quad (M > 1)$

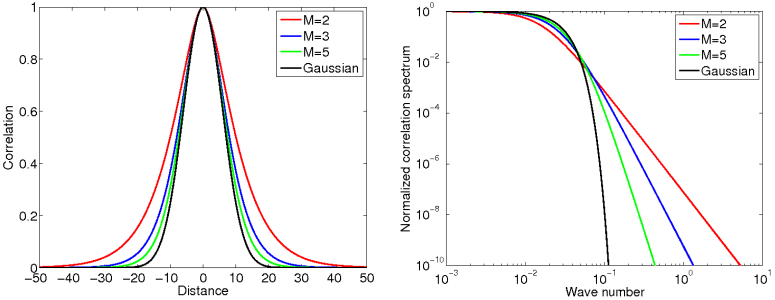
The length scale is defined following Daley (1991) by $\sqrt{-1/f_M^{(2)}(0)}$.

AR functions modeled with a 1D implicit diffusion operator

The Autoregressive functions (AR) are a special case of the Matérn family

$$\frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{r}{L}\right)^{\nu} K_{\nu}\left(\frac{r}{L}\right)$$

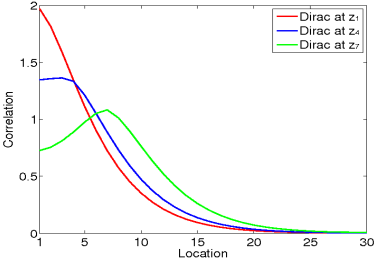
where $r = |z - z'|$



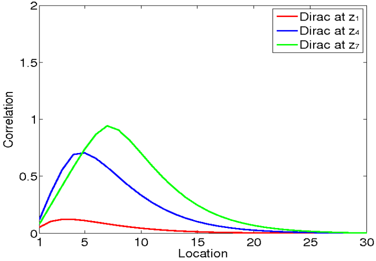
(a) $f_M(r)$ (b) $\hat{f}_M(\hat{r})/\hat{f}_M(0)$

Figure: Variation of the shape and spectrum with M for a fixed length-scale.

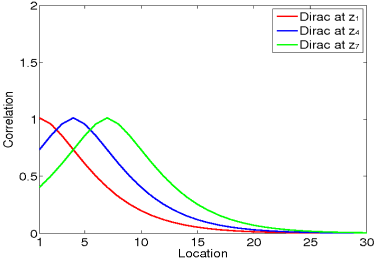
Boundary conditions for grid-point correlation filters



(a) $\lambda_M \mathcal{L}_{\mathrm{N}}^{M}[\,\eta_0(z)\,]$



(b) $\lambda_M \mathcal{L}_{\mathrm{D}}^{M}[\,\eta_0(z)\,]$



(c) $\frac{1}{2}\lambda_M\left(\mathcal{L}_{\mathrm{N}}^{M}[\,\eta_0(z)] + \mathcal{L}_{\mathrm{D}}^{M}[\,\eta_0(z)]\right)$

Boundary conditions for grid-point correlation filters cont.

Define the correlation operator ($\mathcal{F}_M$) as the average of solutions from two diffusion problems with different BCs (Neumann and Dirichlet).

$$\mathcal{F}_M[\eta_0(z)] = \frac{1}{2}\lambda_M \left(\mathcal{L}_{\mathrm{N}}^M[\,\eta_0(z)] + \mathcal{L}_{\mathrm{D}}^M[\,\eta_0(z)] \right)$$

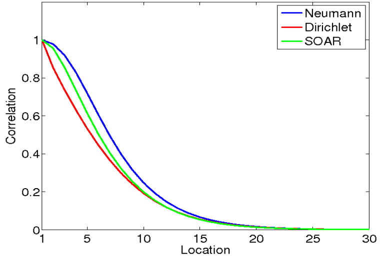
where

$$\begin{aligned} \mathcal{L}_{\mathrm{N}}^M[\cdot] &: \quad M\text{-step diffusion operator with Neumann BC.} \\ \mathcal{L}_{\mathrm{D}}^M[\cdot] &: \quad M\text{-step diffusion operator with Dirichlet BC.} \\ \lambda_M &: \quad \text{normalization constant.} \end{aligned}$$

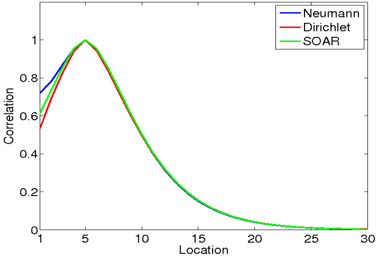
- Simulates transparent BCs under quite general conditions.
- Exact result for the constant-coefficient diffusion equation.
- Works well in more general cases (spatially varying coefficients).
- Less accurate when length scale $\sim\mathrm{O}$(physical domain) or $\sim\mathrm{O}$(grid size).
- Requires an additional application of the diffusion equation.

Exact and randomization methods for normalization

- Advantages:
  - Need only one application of the diffusion equation.
  - Take into account numerical errors.
  - Guarantee the maximum to be 1 (for exact).
- Drawbacks
  - Randomization method is accurate if you have enough members.
  - Require many applications of the diffusion equation.
  - Shift of the length scale near boundaries.



(d) Response to δz_1



(e) Response to δz_5

Location-dependent length scales

To account for spatially varying length scales, we use the inhomogeneous diffusion equation

$$\frac{\partial \eta}{\partial t}(z,t) - \frac{\partial}{\partial z}\left(\kappa(z)\frac{\partial \eta}{\partial z}(z,t)\right) = 0$$

and the correlation operator becomes

$$\mathcal{F}_{\mathrm{M}}[\eta_0(z)] = \sqrt{\widetilde{\lambda}_M(z)}\mathcal{L}^M\left[\sqrt{\widetilde{\lambda}_M(z)}\eta_0(z)\right]$$

A reasonable approximation of the normalization factor is

$$\widetilde{\lambda}_M(z) = \upsilon_{\mathrm{M}} L(z)$$

where $L(z) = \sqrt{\kappa(z)\Delta t}$
and υ_{M} is the coefficient defined for constant length scales.

Location-dependent length scales

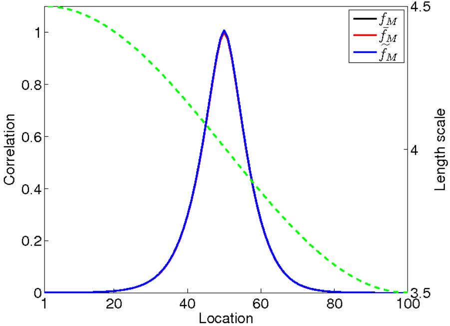
We might expect that the associated kernel has the form

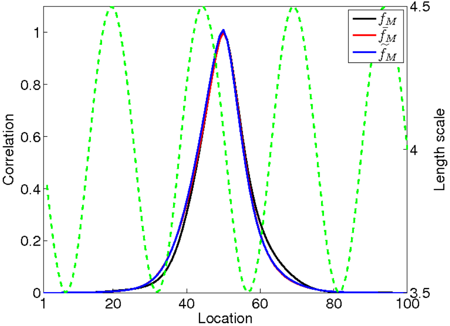
$$f_M(z,z') \approx \frac{\sqrt{L(z)}\sqrt{L(z')}}{\sqrt{\overline{L^2(z,z')}}} \left[1 + \sum_{j=1}^{M-1} \beta_j \left(\frac{|z-z'|}{\sqrt{\overline{L^2(z,z')}}} \right)^j \right] \exp\left(- \frac{|z-z'|}{\sqrt{\overline{L^2(z,z')}}} \right)$$

where $\overline{L^2(z,z')}$ can be defined as

- the geometric mean $\quad \overline{L^2(z,z')} = L(z)L(z')$
  $\rightarrow$ used in some applications but does not guarantee the positive definiteness of $f_M(z,z')$
- or the arithmetic mean $\overline{L^2(z,z')} = \left(L^2(z) + L^2(z')\right)/2$
  $\rightarrow$ does guarantee positive definiteness

Location-dependent length scales





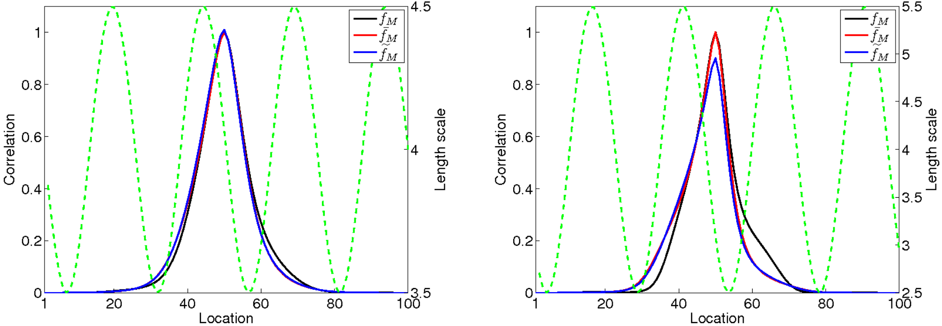
Green: varying length scale

Black: function with arithmetic mean

Blue: diffusion + $\widetilde{\lambda}_M(z) = \upsilon_M L(z)$

Red: diffusion + normalization following Purser *et al.* (2003)

Location-dependent length scales



Green: varying length scale

Black: function with arithmetic mean

Blue: diffusion + $\widetilde{\lambda}_M(z) = \upsilon_M L(z)$

Red: diffusion + normalization following Purser *et al.* (2003)

Application of the 1D diffusion operator

$$\mathbf{B} = \mathbf{U}\mathbf{U}^{\mathrm{T}} = \mathbf{K}\mathbf{D}^{1/2}\mathbf{C}\mathbf{D}^{1/2}\mathbf{K}^{\mathrm{T}}$$

- One diffusion with Neumann or Dirichlet BCs

$$\mathbf{C} = \mathbf{\Lambda}^{1/2}\mathbf{L}^{M/2}\mathbf{W}^{-1}(\mathbf{L}^{M/2})^{\mathrm{T}}\mathbf{\Lambda}^{1/2}$$

- Average of two diffusions with different BCs

$$\begin{aligned}\mathbf{C} &= \frac{1}{2}\mathbf{\Lambda}^{1/2}\left(\mathbf{L}_{\mathrm{N}}^{M/2}\mathbf{W}^{-1}(\mathbf{L}_{\mathrm{N}}^{M/2})^{\mathrm{T}} + \mathbf{L}_{\mathrm{D}}^{M/2}\mathbf{W}^{-1}(\mathbf{L}_{\mathrm{D}}^{M/2})^{\mathrm{T}}\right)\mathbf{\Lambda}^{1/2} \\ &= \sqrt{\frac{1}{2}}\mathbf{\Lambda}^{1/2}\begin{pmatrix}\mathbf{L}_{\mathrm{N}}^{M/2} & \mathbf{L}_{\mathrm{D}}^{M/2}\end{pmatrix}\mathbf{W}^{-1}\begin{pmatrix}\left(\mathbf{L}_{\mathrm{N}}^{M/2}\right)^{\mathrm{T}} \\ \left(\mathbf{L}_{\mathrm{D}}^{M/2}\right)^{\mathrm{T}}\end{pmatrix}\mathbf{\Lambda}^{1/2}\sqrt{\frac{1}{2}}\end{aligned}$$

  - Rectangular matrices are twice the size of the previous **L**.

Λ is the diagonal matrix of normalization factors, **L** the matrix form of the diffusion operator, and **W** the diagonal matrix of grid elements.

Inverting the matrix with a Cholesky decomposition

Applying **L** is equivalent to integrating the diffusion equation on one time step with $\boldsymbol{\eta}^{m-1}$ as the initial condition:

- The problem to solve is given by $\boldsymbol{\eta}^{m-1} = \mathbf{A}\boldsymbol{\eta}^{m}$
  $\rightarrow$ **A** is the triband implicit matrix.

- Multiplying both sides of the equation by **W** to symmetrize the matrix gives $\mathbf{W}\boldsymbol{\eta}^{m-1} = \mathbf{W}\mathbf{A}\boldsymbol{\eta}^{m}$
  $\rightarrow$ **WA** is a symmetric positive definite (diagonally dominant) matrix.

- Cholesky decomposition $\mathbf{W}\mathbf{A} = \mathbf{G}\mathbf{G}^{\mathrm{T}}$ with **G** a lower triangular matrix
  $\rightarrow$ taking advantage of the shape of **WA** $\Rightarrow$ 2 (or 3) bands to store
- Solve
  - Forward elimination: $\boldsymbol{\eta}^{m-1} = \mathbf{G}\boldsymbol{y}$
  - Backward substitution: $\boldsymbol{y} = \mathbf{G}^{\mathrm{T}}\boldsymbol{\eta}^{m}$

Summary of the 1D theory

Mirouze and Weaver (2010), QJRMS in review.
Deliverable D2.1.1

- Diffusion operator:
  - Explicit scheme:
    - Easy to code but may be costly.
  - Implicit scheme
    - Requires a matrix inversion but has no stability criterion.
    - Cholesky decomposition for the 1D problem.
- Boundary conditions
  - Neumann or Dirichlet BCs lead to a corrupted kernel near boundaries.
  - Average of two diffusions with different BCs results in unaffected solution.
    - Requires rectangular matrices.

Summary of the 1D theory

- Normalization
  - Exact or randomization methods:
    - Take into account numerical errors (including BCs corruption) with a shift near boundaries.
    - Need only one diffusion application but can be expensive.
  - Analytical factor:
    - Cheap to compute.
    - Accurate near the boundaries with two diffusion applications.
    - Accurate if the numerical errors are small (length scale with respect of the grid size or physical domain, variation).

Summary

A 3D correlation matrix based on a product of 1D implicit diffusion opts.

$$\mathbf{C} = \mathbf{C}_x \, \mathbf{C}_y \, \mathbf{C}_z$$

where each 1D correlation block is formulated as

$$\mathbf{C}_x = \frac{1}{2}\boldsymbol{\Lambda}_x^{1/2} \left(\mathbf{L}_{\mathrm{N}_x}^{M/2} \mathbf{W}_x^{-1} (\mathbf{L}_{\mathrm{N}_x}^{M/2})^{\mathrm{T}} + \mathbf{L}_{\mathrm{D}_x}^{M/2} \mathbf{W}_x^{-1} (\mathbf{L}_{\mathrm{D}_x}^{M/2})^{\mathrm{T}} \right) \boldsymbol{\Lambda}_x^{1/2} \qquad \text{etc.}$$

Multiply out different terms in $\mathbf{C}$ and interchange "square-root" factors to force symmetry:

$$\begin{aligned}\mathbf{C} \approx \frac{1}{8}\boldsymbol{\Lambda}^{1/2} \ \{ \ & \mathbf{L}_{\mathrm{N}_x}^{M/2} \mathbf{L}_{\mathrm{N}_y}^{M/2} \mathbf{L}_{\mathrm{N}_z}^{M/2} \mathbf{W}^{-1} (\mathbf{L}_{\mathrm{N}_z}^{M/2})^{\mathrm{T}} (\mathbf{L}_{\mathrm{N}_y}^{M/2})^{\mathrm{T}} (\mathbf{L}_{\mathrm{N}_x}^{M/2})^{\mathrm{T}} \\ + \ & \mathbf{L}_{\mathrm{N}_x}^{M/2} \mathbf{L}_{\mathrm{N}_y}^{M/2} \mathbf{L}_{\mathrm{D}_z}^{M/2} \mathbf{W}^{-1} (\mathbf{L}_{\mathrm{D}_z}^{M/2})^{\mathrm{T}} (\mathbf{L}_{\mathrm{N}_y}^{M/2})^{\mathrm{T}} (\mathbf{L}_{\mathrm{N}_x}^{M/2})^{\mathrm{T}} \\ + \ & \ldots \\ + \ & \mathbf{L}_{\mathrm{D}_x}^{M/2} \mathbf{L}_{\mathrm{D}_y}^{M/2} \mathbf{L}_{\mathrm{D}_z}^{M/2} \mathbf{W}^{-1} (\mathbf{L}_{\mathrm{D}_z}^{M/2})^{\mathrm{T}} (\mathbf{L}_{\mathrm{D}_y}^{M/2})^{\mathrm{T}} (\mathbf{L}_{\mathrm{D}_x}^{M/2})^{\mathrm{T}} \} \boldsymbol{\Lambda}^{1/2}\end{aligned}$$

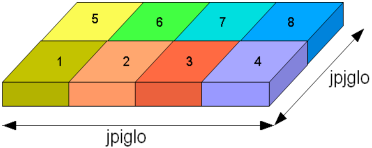
$\mathbf{W} = \mathbf{W}_x \mathbf{W}_y \mathbf{W}_z$ and $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}_x \boldsymbol{\Lambda}_y \boldsymbol{\Lambda}_z$ are diagonal matrices of volume elements and normalization factors.

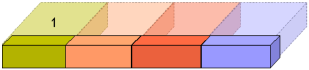
The Cholesky decomposition for the zonal direction: code details

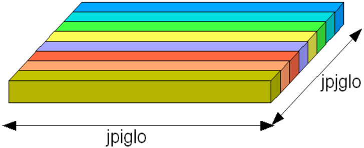
- Lateral boundary conditions:
  - Closed.
  - Cyclic East-West.
- Identification of the 1D problems:
  - *jpjglo* $\times$ *jpk* problems where several domains are bounded by coasts.
  - Cyclic East-West $\rightarrow$ use of periodic boundaries.
  - Use the masks to account for Neumann or Dirichlet BCs.
- Cholesky decomposition:
  - 3 bands of size *jpiglo* $\times$ *jpjglo* $\times$ *jpk* for each diffusion application.
  - Neumann BCs by default, Dirichlet BCs if *key_cor_npd* defined.
- Remapping:
  - Rearrange the processors to have a set of entire lines on each processor.
  - Apply the diffusion operator $M/2$ times.
  - Return to the original processor arrangement.

The Cholesky decomposition for the zonal direction

Example of remapping with $jpni = 4$ and $jpnj = 2$:





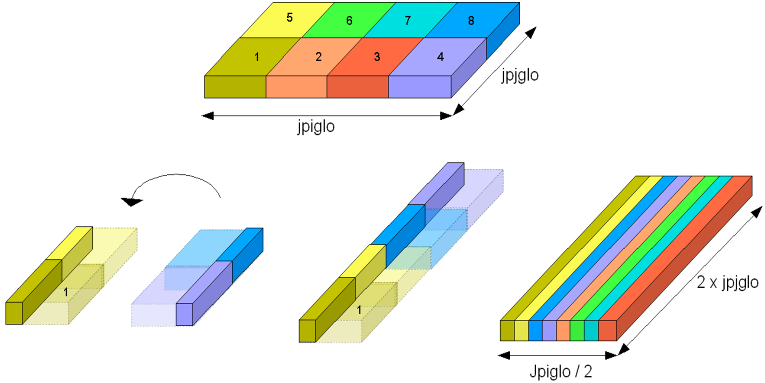


The Cholesky decomposition for the meridional direction: code details

- Lateral boundary conditions:
  - Closed (not yet taken into account).
  - South (not yet taken into account).
  - North fold (even *jpni* with the same *nlci* only, or not efficient $jpnij = 1$).
- Identification of the 1D problems:
  - One line goes from Antarctic to the opposite Antarctic following the North fold.
  - $jpiglo/2 \times jpk$ problems where several domains are bounded by coasts.
  - Use the masks to account for Neumann or Dirichlet BCs.
- Cholesky decomposition:
  - 2 bands of size $jpiglo \times jpjglo \times jpk$ for each diffusion application.
  - Neumann BCs by default, Dirichlet BCs if *key_cor_npd* defined.
- Remapping:
  - Rearrange the processors to have a set of entire lines on each processor.
  - Apply the diffusion operator $M/2$ times.
  - Return to the original processor arrangement.

The Cholesky decomposition for the meridian direction

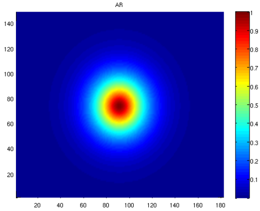
Example of remapping with $jpni = 4$ and $jpnj = 2$:



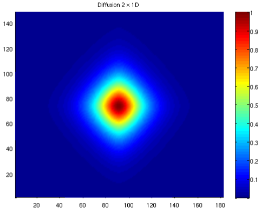
The Cholesky decomposition for the vertical direction

- No lateral boundary conditions.
- Identification of the 1D problems:
  - *jpiglo* × *jpjglo* problems where each domain is bounded by the surface and the bathymetry.
  - Use the masks to account for Neumann or Dirichlet BCs.
- Cholesky decomposition:
  - 2 bands of size *jpiglo* × *jpjglo* × *jpk* for each diffusion application.
  - Neumann BCs by default, Dirichlet BCs if *key_cor_npd* defined.
- No remapping.

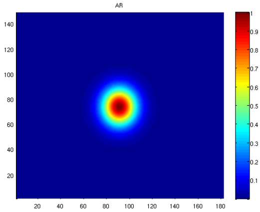
How many iterations should we use?



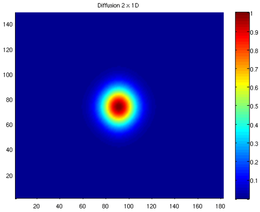
(f) AR M=2



(g) Diffusion M=2



(h) AR M=10



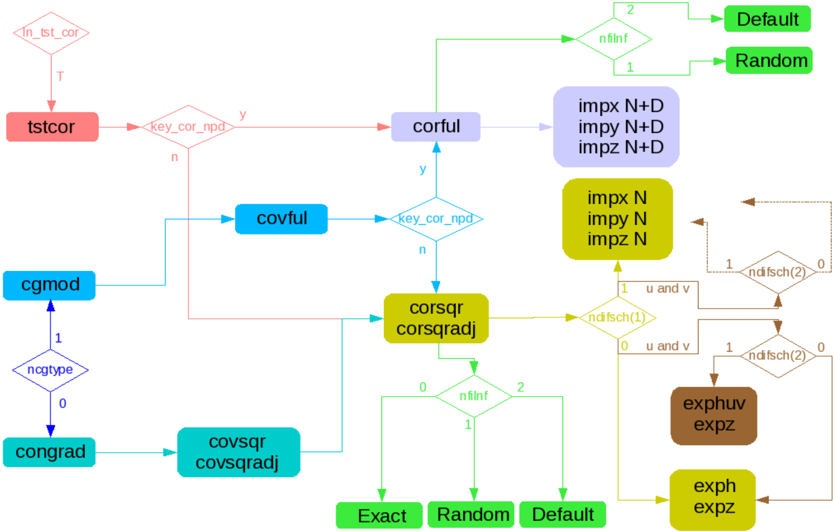
(i) Diffusion M=10

Two minimization algorithms are now available

When the average of two diffusion applications with different BCs are used, we need to handle rectangular matrices.

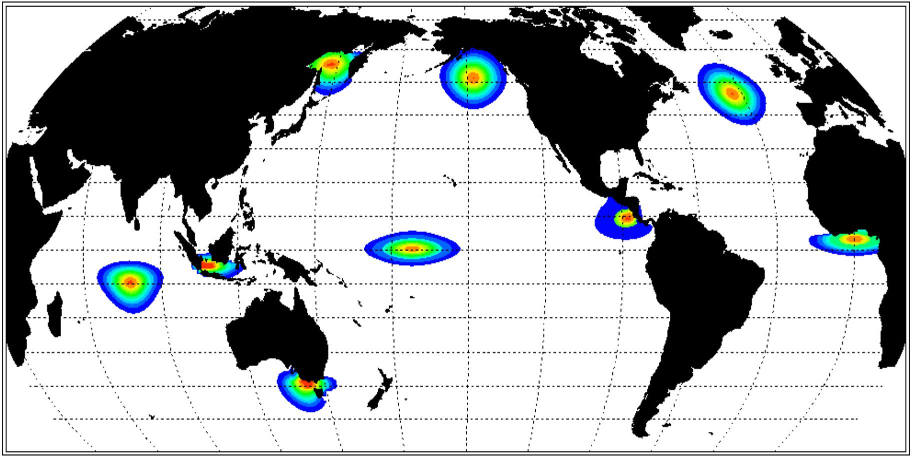
- CONGRAD:
  - Preconditioning by $\mathbf{B}^{1/2}$.
    - ★ The rectangular matrices are needed throughout the minimization.
- CGMOD (see Anthony's presentation):
  - Preconditioning by $\mathbf{B}$.
    - ★ The rectangular matrices are needed punctually when applying $\mathbf{B}$.

The different options



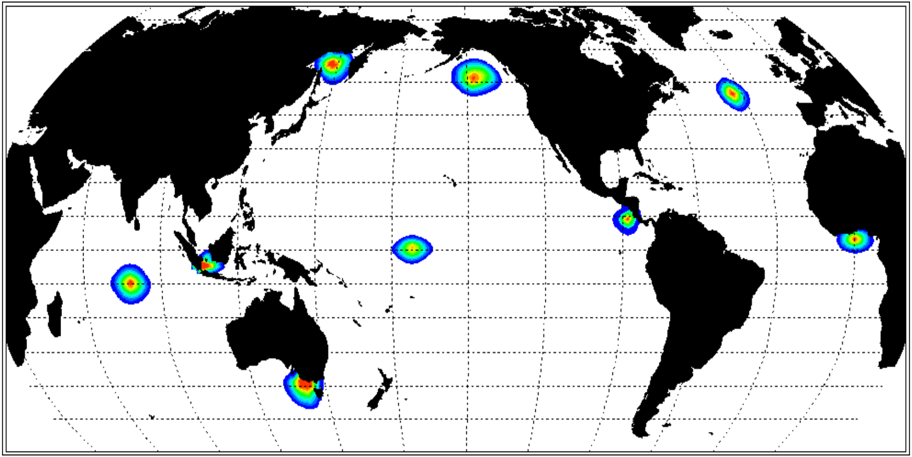
T-T correlations at selected points in ORCA2

Produced using the parameterized length scales of Daget *et al.* (2009), and $M = 10$ implicit iterations (in x, y and z).



T-T correlations at selected points in ORCA2

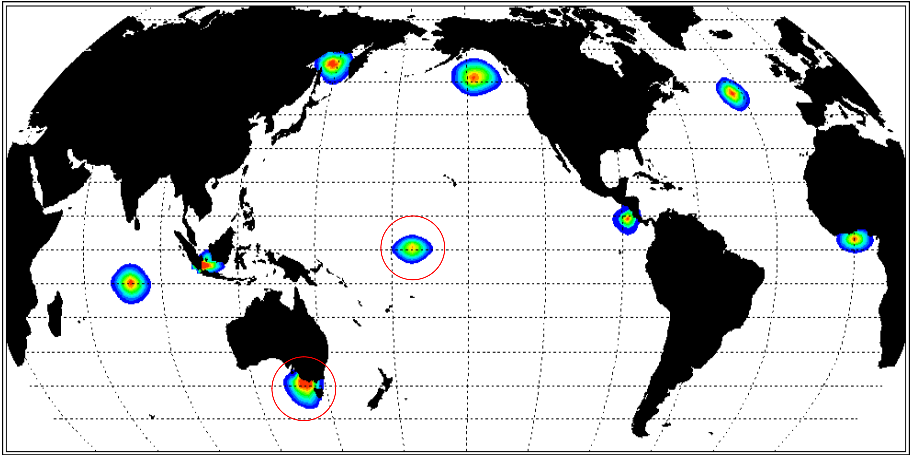
Produced using the ensemble-estimated length scales of Daget (2008), and $M = 10$ implicit iterations (in x, y and z)*.



* ~100× more iterations are required with an explicit scheme.

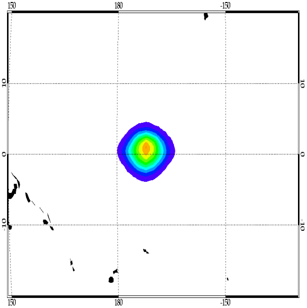
T-T correlations at selected points

Produced using the ensemble-estimated length scales of Daget (2008), and $M = 10$ implicit iterations (in x, y and z)*.

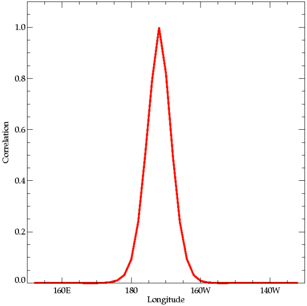


* $\sim$100$\times$ more iterations required with an explicit scheme.

T-T correlations

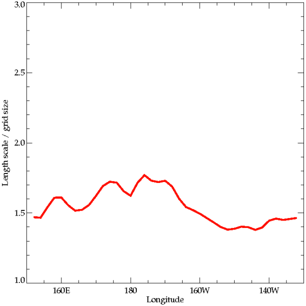


Correlation



Longitude

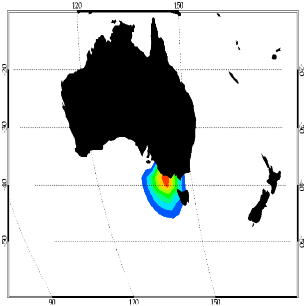
Length scale / grid size



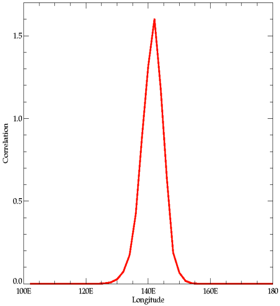
Longitude

- Amplitude at the origin = 0.99
- The analytical estimate of the local normalization constant is very accurate.

T-T correlations

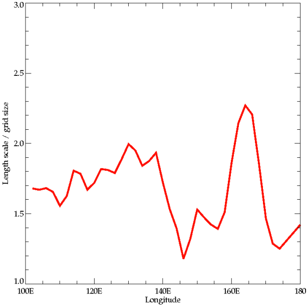


Correlation



Longitude

Length scale / grid size

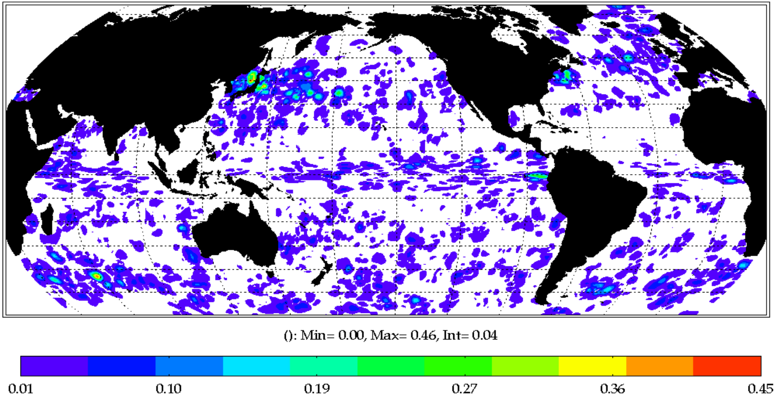


Longitude

- Amplitude at the origin = 1.6
- Rapid spatial variations in L lead to errors in the analytical estimate of the local normalization constant.

How many members for the randomization method?

Differences in the surface T increments between 50 members and 200 members for an explicit scheme.



Comparison between Explicit and Implicit in ORCA1

Different configurations with $jpni = 4$ and $jpnj = 4$ in ORCA1 compared to Explicit where the normalization is a randomized estimate with 200 members.

- CONGRAD 40 iterations

	Elapsed time		Memory	
Explicit_200	283 s		13 724 MB	
Implicit_200	194 s	- 30 %	14 056 MB	+ 300
Implicit_def	137 s	- 50 %	13 924 MB	+ 200

- CGMOD 40 iterations

	Elapsed time		Memory	
Explicit_200	330 s		14 1964 MB	
Implicit_200	208 s	- 35 %	14 602 MB	+ 400
Implicit_def_npd	329 s	0 %	15 089 MB	+ 800
Implicit_def_npd_2	211 s	- 35 %	9459 MB	- 4700

Implicit_def_npd_2: no orthogonalization, 5 more iterations.

Summary

Conclusions

- **2D and 3D implicit schemes available:**
  - Based on the product of 3 x 1D operators.
    - u and v not yet available.
    - Allows for spatial variation of the length scales in the directions of the computational coordinates only.
  - Matrix inversion uses a Cholesky decomposition.
    - MPP version available by remapping (restrictions for the Y axis).
- **Analytical normalization factor available:**
  - Important feature when the scales are flow dependent.
  - Accurate near boundaries only with the average of two diffusion applications with different BCs.
    - Requires rectangular matrices $\rightarrow$ for the minimization, a preconditioning by **B** is more convenient (CGMOD).
  - More work is needed to improve its accuracy in general cases.
  - The randomization algorithm is still an option.
- **Ensemble-estimated length scales can be taken into account.**
  - With restrictions.

Future work

- **Finalize the testing of the 2D and 3D x 1D operator:**
  - Accuracy, cost (time and memory).
  - Long 3D-Var experiment.
- **Improve the estimation of the normalization factors:**
  - Numerical error due to the finite differences.
  - Explore Purser *et al.* (2003) method.
  - ...
- **Take into account flow dependency**
  - Estimate the variances and the length-scales with an ensemble method.
- **Take into account full anisotropy: (Post-VODA)**
  - A non-diagonal diffusion tensor is required for fully anisotropic correlations.
  - This leads to non-trivial extensions to the correlation model.