Normalized Convolution Interpolated from an Adaptive Subgrid

Benjamin Ménétrier
NCAR - Boulder, Colorado
August 1\textsuperscript{st}, 2017
Outline

Principles

Subgrid definition

Convolution function

MPI communications

Applying NICAS

Conclusions
Outline

Principles

Subgrid definition

Convolution function

MPI communications

Applying NICAS

Conclusions
Explicit convolution

We want to apply a correlation matrix on any grid type.

Advantages of an explicit convolution $\mathbf{C} \in \mathbb{R}^{n \times n}$:

- Explicit choice of the convolution function
- Exact normalization ($C_{ii} = 1$)
- Work on any grid type!

Drawback: cost scales as $O(n^2)$...

To limit the cost, work on a grid subset: $\tilde{\mathbf{C}} = \mathbf{S}\mathbf{C}^s\mathbf{S}^T$ where

- $\mathbf{S} \in \mathbb{R}^{n \times n_s}$ is an interpolation from the subgrid
- $\mathbf{C}^s \in \mathbb{R}^{n_s \times n_s}$ is a convolution matrix on the subgrid

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

Problems with working on a subgrid:

- If $n_s$ is too small, the fonction is distorted,
- Normalization breaks down because of the interpolation: even if $\mathbf{C}^s$ is normalized, $\tilde{\mathbf{C}}$ is not.
Convolution explicite

NICAS method (Normalized Interpolated Convolution from an Adaptive Subgrid) :

\[
\tilde{C} = \text{NSC}^{s}S^{T}N^{T}
\]

where \( \text{N} \) is a diagonal normalization matrix.

Several questions:

- What subgrid? What interpolation?
- What convolution function?
- How to compute the normalization?
- What parallelization method?
- What results/cost/scalability?
- ...
Outline

Principles

Subgrid definition

Convolution function

MPI communications

Applying NICAS

Conclusions
Subgrid choice

Full 3D interpolation: hard to compute and to apply in a geophysical model ⇒ \( S \) split into three operators:

\[
S = S^h S^v S^s
\]

<table>
<thead>
<tr>
<th>Grid</th>
<th>Subset of points (number of points)</th>
<th>Subset of levels (number of levels)</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>( G^s )</td>
<td>( S^c_2 (n^c_2) )</td>
<td>( S^l_1 (n^l_1) )</td>
<td>( S^c_2 ) depending on the level</td>
</tr>
<tr>
<td>( G^v )</td>
<td>( S^c_1 (n^c_1) )</td>
<td>( S^l_1 (n^l_1) )</td>
<td>( n^c_2 \leq n^c_1 )</td>
</tr>
<tr>
<td>( G^h )</td>
<td>( S^c_1 (n^c_1) )</td>
<td>( S^l_0 (n^l_0) )</td>
<td>( n^l_1 \leq n^l_0 )</td>
</tr>
<tr>
<td>( G^f )</td>
<td>( S^c_0 (n^c_0) )</td>
<td>( S^l_0 (n^l_0) )</td>
<td>( n^c_1 \leq n^c_0 )</td>
</tr>
</tbody>
</table>
Choice of the horizontal grid

Black dots: ARPEGE grid at truncation TL149 (set $S^C_0$)
Choice of the horizontal grid

Green dots: basic subset $S^c_1$
Choice of the horizontal grid

Red dots: final subset $S^c_2$ at a level with a short support radius
Choice of the horizontal grid

Red dots: final subset $S^c_2$

at a level with a medium support radius
Choice of the horizontal grid

Red dots: final subset $S_c^2$
at a level with a large support radius
Choice of the vertical grid

Levels sampled depending on the vertical support radius. Example: uniform log(pressure) vertical support radius.

ARPEGE levels (black dots) / subgrid levels (red dots)
Outline

Principles

Subgrid definition

Convolution function

MPI communications

Applying NICAS

Conclusions
Choice of the convolution function

Gaspari and Cohn (1999) function of compact support radius $r$:

$\rightarrow$ homogeneous normalized distance $d_{ij}' = \frac{d_{ij}}{r}$
Choice of the convolution function

Gaspari and Cohn (1999) function of compact support radius $r$

$\rightarrow$ homogeneous normalized distance $d'_{ij} = \frac{d_{ij}}{r}$
Choice of the convolution function

Gaspari and Cohn (1999) function of compact support radius $r$

$\rightarrow$ heterogeneous normalized distance $d'_{ij} = \frac{d_{ij}}{\sqrt{(r_i^2 + r_j^2)/2}}$
Homogeneous / heterogenous support radius

Homogeneous support radius $\rightarrow$ homogenous subgrid:
Homogeneous / heterogeneous support radius

Heterogeneous support radius → heterogeneous subgrid:
Homogeneous / heterogeneous support radius

Convolution with an homogenous support radius
Homogeneous / heterogeneous support radius

Convolution with an heterogeneous support radius
Sharp support radius gradients

Gaspari and Cohn (1999) function of compact support radius \( r \)

\[ d_{ij}' = \frac{d_{ij}}{\sqrt{(r_i^2 + r_j^2)/2}} \]
Sharp support radius gradients

Gaspari and Cohn (1999) function of compact support radius $r$

$\rightarrow$ heterogeneous normalized distance $d'_{ij} = \frac{d_{ij}}{\sqrt{(r_i^2 + r_j^2)/2}}$
Sharp support radius gradients

Gaspari and Cohn (1999) function of compact support radius \( r \)

\[ d'_{ij} = \frac{d_{ij}}{\sqrt{(r_i^2 + r_j^2)/2}} \]
Sharp support radius gradients

Gaspari and Cohn (1999) function of compact support radius $r$

$\rightarrow$ heterogeneous normalized distance $d_{ij}' = \sum_{k=i}^{j-1} d_{k,k+1}'$ (network)
Sharp support radius gradients

Convolution function for a homogeneous support radius $r$
Heterogeneous support radius $r$ and subset $S_2^c$ (black dots)
Sharp support radius gradients

Convolution function for a heterogeneous support radius $r$ with the distance-based approach; zone of lower $r$ (dotted)
Convolution function for a heterogeneous support radius \( r \) with the network-based approach; zone of lower \( r \) (dotted)
Convolution functions with complex boundaries of the NEMO grid:

- for a distance-based approach (left)
- for a network-based approach (right)
Subgrid resolution

For a given support radius, a resolution parameter $\rho$ defines the subgrid density:

$$\rho = 8 \text{ (2827 points)}$$
Subgrid resolution

For a given support radius, a resolution parameter $\rho$ defines the subgrid density:

$$\rho = 6 \text{ (1590 points)}$$
For a given support radius, a resolution parameter $\rho$ defines the subgrid density:

$$\rho = 4 \ (706 \text{ points})$$
Subgrid resolution

For a given support radius, a resolution parameter $\rho$ defines the subgrid density:

Horizontal convolution function for a spectral method and for the NICAS method with a decreasing resolution ($\rho = 8, 6$ and $4$)
Subgrid resolution

For a given support radius, a resolution parameter $\rho$ defines the subgrid density:

Vertical convolution function for a spectral method and for the NICAS method with a decreasing resolution ($\rho = 8, 6$ and $4$)
Normalization computation

Prohibitive cost of applying the non-normalized method to a Dirac at every gridpoint ($O(n^2)$). However, very limited number of nodes involved in the computation of $N_{ii}$. Number of non-zero nodes:

- 1 node of $G^f$ in $\delta_i$,
- up to 3 nodes of $G^h$ in $S^{hT}\delta_i$ (bilinear interp.)
- up to 6 nodes of $G^v$ in $S^{vT}S^{hT}\delta_i$ (linear interp.)
- up to 18 nodes of $G^s$ in $S^{sT}S^{vT}S^{hT}\delta_i$ (bilinear interp.)

Affordable procedure:
1. computing the 18 non-zero values of $S^{sT}S^{vT}S^{hT}\delta_i$ ($\rightarrow \delta'$)
2. applying the relevant block of the subgrid convolution matrix $C^s$ to these coefficients ($\rightarrow \delta''$)
3. computing the normalization coefficient as the inverse square-root of a scalar product ($N_{ii} = \langle \delta', \delta'' \rangle^{-1/2}$)
Outline

Principles

Subgrid definition

Convolution function

MPI communications

Applying NICAS

Conclusions
MPI communications

Four different halo zones:

- **[F]** zone: full grid $G^f$ nodes on a given task (usually provided by the model itself)
- **[A]** zone: subgrid $G^s$ nodes on a given task (define by the subsampling on [F])
- **[B]** zone: subgrid $G^s$ nodes involved in the interpolation
- **[C]** zone: subgrid $G^s$ nodes involved in the convolution

$$\tilde{C} = [F] \ N \ [F] \ S \ [B] \leftarrow [A] \leftarrow [C] \ C^s \ [C] \leftarrow [B] \ S^T \ [F] \ N^T \ [F]$$
MPI communications

Four different halo zones:

- **[F]** zone: full grid $\mathcal{G}^f$ nodes on a given task (usually provided by the model itself)
- **[A]** zone: subgrid $\mathcal{G}^s$ nodes on a given task (define by the subsampling on [F])
- **[B]** zone: subgrid $\mathcal{G}^s$ nodes involved in the interpolation
- **[C]** zone: subgrid $\mathcal{G}^s$ nodes involved in the convolution

\[
\tilde{C} = \begin{bmatrix} \mathbf{F} & \mathbf{N} \end{bmatrix} \begin{bmatrix} \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{B} \leftarrow \mathbf{A} \leftarrow \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{C}^s \end{bmatrix} \begin{bmatrix} \mathbf{S}^T \end{bmatrix} \begin{bmatrix} \mathbf{S}^T & \mathbf{F} & \mathbf{N}^T \end{bmatrix} \begin{bmatrix} \mathbf{F} \end{bmatrix}
\]
Outline

Principles

Subgrid definition

Convolution function

MPI communications

Applying NICAS

Conclusions
Offline computations

- Namelist
  - Model grid

NICAS

- Compute parameters and normalization
- Model grid splitting

- Support radii (if heterogeneous)

Compute MPI distribution

- All parameters
- Parameters for proc 1
- Parameters for proc 2
- Parameters for proc p

Optional tests (e.g. adjointness, dirac application)

- Dirac test results

Restarts
Inline computations

Generic implementation pattern for all linear operations (horizontal and vertical interpolations, convolution):

\[
\text{fld}_{\text{out}} = 0.0 \\
d\text{o } i_s=1,n_s \\
\quad \text{fld}_{\text{out}}(\text{row}(i_s)) = \text{fld}_{\text{out}}(\text{row}(i_s)) \\
\quad \quad + S(i_s) \times \text{fld}_{\text{in}}(\text{col}(i_s)) \\
\end{do}

where \text{fld}_{\text{in}} is the input field, \text{fld}_{\text{out}} the output field, \text{n}_s the number of operations involving \text{row} and \text{col} as indices and \text{S} as coefficients. Easy to code the adjoint operator:

\[
\text{fld}_{\text{out}} = 0.0 \\
d\text{o } i_s=1,n_s \\
\quad \text{fld}_{\text{out}}(\text{col}(i_s)) = \text{fld}_{\text{out}}(\text{col}(i_s)) \\
\quad \quad + S(i_s) \times \text{fld}_{\text{in}}(\text{row}(i_s)) \\
\end{do}
Subset $S^c_2$ for a heterogeneous support radius $r$, for the regional model AROME:
Generic implementation

Convolution functions for a heterogeneous support radius \( r \), for the regional model AROME:
Scalability

Elapsed time for one application of the correlation with ARPEGE on a TL149 grid:

Scaling:
- Spectral
- NICAS - 1 com. - $\rho = 4$
- NICAS - 1 com. - $\rho = 6$
- NICAS - 1 com. - $\rho = 8$
- NICAS - 2 com. - $\rho = 4$
- NICAS - 2 com. - $\rho = 6$
- NICAS - 2 com. - $\rho = 8$
Scalability

Elapsed time for one application of the correlation with ARPEGE on a TL399 grid:

Scaling:
- Spectral
- NICAS - 1 com. - $\rho = 4$
- NICAS - 1 com. - $\rho = 6$
- NICAS - 1 com. - $\rho = 8$
- NICAS - 2 com. - $\rho = 4$
- NICAS - 2 com. - $\rho = 6$
- NICAS - 2 com. - $\rho = 8$
Conclusions

Already coded:

- Generic Fortran/C++ code for the NICAS method (offline computations):
  - using ESMF for horizontal interpolation
  - convolution with a Gaspari and Cohn (1999) function, possibly heterogeneous and with complex boundaries
  - exact normalization
  - approximate square-root formulation
  - hybrid MPI/OpenMP parallelization
  - basic tests (adjoint, positive-definiteness, Diracs)

- Inline computations efficiency tested in ARPEGE and AROME models → satisfying scalability

To be done:

- Further testing, validation, documentation
- Inline implementation in a flexible framework...