

ESMF v7.1.0r remapping between two structured grids and between two unstructured grids and one structured grid

Coquart L. , Valcke S.

CECI, Université de Toulouse, CNRS, CERFACS, Toulouse,
France – TR-CMGC-19-43



The work reported in this document has been done in the framework of the IS-ENES3 project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084.

1 Table des matières

2	Introduction	3
3	Creation of the grids at the ESMF format	4
3.1	<i>Creation of the structured grids bggd and nogt</i>	4
3.2	<i>Creation of the icosahedral grid icos</i>	4
3.3	<i>Creation of the unstructured gaussian reduced grid ssea</i>	5
4	Creation of the ESMFv7.1.0r remapping weights	7
5	ESMF v7.1.0r remapping results between bggd and nogt	8
5.1	<i>Results for the non-conservative remapping between bggd and nogt</i>	9
5.1.1	<i>Results from nogt to bggd</i>	9
5.1.2	<i>Results from bggd to nogt</i>	9
5.2	<i>Results for the conservative remapping between bggd and nogt</i>	10
5.2.1	<i>Results from nogt to bggd</i>	10
5.2.2	<i>Results from bggd to nogt</i>	11
6	ESMF v7.1.0r remapping results between bggd and icos	12
6.1	<i>Results for non-conservative remapping between bggd and icos</i>	12
6.1.1	<i>Results from bggd to icos</i>	12
6.1.2	<i>Results from icos to bggd</i>	12
6.2	<i>Results for conservative remapping between bggd and icos</i>	13
6.2.1	<i>Results from bggd to icos</i>	13
6.2.2	<i>Results from icos to bggd</i>	14
7	ESMF v7.1.0r remapping results between bggd and ssea	15
7.1	<i>Results for non-conservative remapping between bggd and ssea</i>	15
7.1.1	<i>Results from bggd to ssea</i>	15
7.1.2	<i>Results from ssea to bggd</i>	15
7.2	<i>Results for conservative remapping between bggd and ssea</i>	16
7.2.1	<i>Results from bggd to ssea</i>	16
7.2.2	<i>Results from ssea to bggd</i>	17
8	Conclusion and perspective	18
9	Bibliography	19
10	Appendix A: Compiling and installing ESMF v7.1.0r on Kraken (Cerfacs Lenovo computer)	20
11	Appendix B: Grids format for ESMFv7.1.0r	21
11.1	<i>Structured grids</i>	21
11.2	<i>Unstructured grids</i>	22
12	Appendix C : ESMF v7.1.0r remapping files	23
12.1	<i>Creation of ESMF v7.1.0r remapping files from bggd to nogt</i>	23
12.2	<i>Creation of ESMF v7.1.0r remapping files between bggd and icos</i>	23
12.3	<i>Creation of ESMF v7.1.0r remapping files between bggd and ssea</i>	24

2 Introduction

The OASIS3-MCT coupler (2-Valcke, S. et al. (2018) ; 6-Craig, A. et al. (2017)), developed at Cerfacs in collaboration with the CNRS and the Argonne National Laboratory, is widely used in the climate community. It includes the SCRIP library routines (13-Jones, P.W. 1998) to offer to the users the possibility to calculate the weight and address files associated to the interpolations between the grids of their models, in spherical coordinates. The remapping files can now be calculated using an OpenMP+MPI hybrid parallelization of the SCRIP, leading to a large decrease of the time spent to create these files at high and very high resolution (3-Piacentini, A. et al (2018), 1-Valcke, S. et al. (2018)).

Since 2015, we are interested in testing other remapping libraries than SCRIP to offer the OASIS users the possibility to choose a remapping method best adapted to their grids. We also observed for a long time some problems at the North pole when using the SCRIP conservative remapping (1-Valcke, S. et al. (2018)). The Earth System Modeling Framework library (9-Theurich, G. et al. (2016)) is one of them. This library (designed as the **ESMF library** in the document below) could then be an alternative to the SCRIP library to calculate the remapping files, in particular for the conservative interpolation.

In 2016, we were able to compare the results obtained with SCRIP and ESMF v6.3.0rp1 libraries between two structured grids for neareststod, bilinear, patch and conserve interpolations (8-Senhaji, H. (2016)). There was no nearest neighbor option implemented in ESMF v6.3.0rp1 so a NCL environment was created to perform ESMF remapping with nearest neighbor option. The unmapped unmasked target points could be replaced by the value of the neareststod remapping. The comparison of the results showed that the remapped fields of the different interpolations were very similar in general in terms of errors. Yet it was observed that ESMF v6.3.0rp1 used in our NCL environment was more strict than SCRIP concerning the nearest neighbor option. It was more “strict” in the sense that when at least one of the original neighbours of a target grid point is masked, the original scheme cannot be applied whereas SCRIP uses all remaining non-masked neighbours to perform a n-nearest neighbor remapping (1-Valcke, S. et al. (2018)).

A new version of ESMF, ESMF v7.1.0r is available since March of 2018. It is now possible to perform second order conservative remapping (conserve2nd) and an extrapolation option for unmapped unmasked destination points was also added, except for conservative remapping. We then decided to do new tests for neareststod, bilinear, patch, conserve and conserve2nd interpolations with ESMF v7.1.0r between two structured grids and between unstructured grids and one structured grid. No NCL environment was created. Only the ESMF v7.1.0r ESMF_RegridWeightGen executable, that creates remapping files between two grids, and that is obtained when compiling and installing the ESMF library (see *Appendix A*), was used on kraken Cerfacs Lenovo computer.

The first part of the document describes the creation of the different grids at the ESMF format (see *Appendix B*). Then it presents the calculation of the ESMF v7.1.0r remapping files between two structured grids (see *Appendix C for other remapping files*). The results obtained between two structured grids and then between two unstructured grid and one structured grid, for the neareststod, bilinear, patch, conserve and conserve2nd ESMF remapping, are discussed. Finally, the last part presents some conclusions about this work and some perspectives.

3 Creation of the grids at the ESMF format

An environment **ESMFv7.1.0r_weights_using_exe** was created on kraken (Cerfacs Lenovo computer) to construct the different grids at the ESMF format and also to calculate the different ESMF remapping weights.

Four grids, coming from the toy INTERPOL of OASIS3-MCT, were tested:

- bggd: 144x143x1 points, IPLS/LMD structured logically rectangular grid
- nogt: 294x362x1 points, IPSL/LOCEAN NEMO structured logically rectangular grid
- icos: 15212x1 points, IPSL/LMD unstructured icosahedral grid
- ssea: 24572x1 points, Météo-France/CNRM unstructured gaussian reduced grid

3.1 Creation of the structured grids bggd and nogt

To create the two structured grids bggd and nogt at the ESMF SCRIP format (see *Appendix B* and *4-ESMF Reference Manual*), the NCL routine **curvilinear_to_SCRIP** was used (5- *NCL, The NCAR Command Language*).

The NCL function is based on the longitudes, the latitudes and the masks of the grids read in the files grids_INTERPOL.nc and masks_INTERPOL.nc defined at the OASIS3-MCT SCRIP format (2- *Valcke et al. (2018)*). The mask in ESMF is the inverse of the one used in OASIS3-MCT (one on the ocean and zero on the land).

The new files bggd_SCRIP_NCL_BR_INTERPOL.nc and nogt_SCRIP_NCL_MR_INTERPOL.nc created at the ESMF format, contain the longitudes and latitudes of the centers and of the corners of the grids, as well as the mask (see *Appendix B*). To perform conservative remapping between bggd and ssea, it was necessary to add afterwards the area of bggd in the ESMF SCRIP format structured grid bggd_SCRIP_NCL_BR_INTERPOL.nc using the file areas_INTERPOL.nc of OASIS3-MCT (see section 3.3, section 7.2 and *Appendix C*).

3.2 Creation of the icosahedral grid icos

The icosahedral grid icos is described in (12- *Boumediene, Y. et al. (2015)*) and is plotted figure 1.

To create the unstructured grid icos at the ESMF unstructured format (see *Appendix B* and *4-ESMF Reference Manual*), a SCRIP file was first created using the longitudes, the latitudes and the mask of the grid contained in the files grids_INTERPOL.nc and masks_INTERPOL.nc at the OASIS3-MCT SCRIP format (2- *Valcke et al. (2018)*). The mask in ESMF is the inverse of the one used in OASIS3-MCT.

Then the executable **ESMF_Scrip2Unstruct**, created when installing and compiling the ESMF library on kraken (see *Appendix A*), was used to obtain the ESMF unstructured grid for icos, using the command:

- `./ESMF_Scrip2Unstruct grid_SCRIP_icos.nc icos_ESMF_unstruct_six_corners_INTERPOL.nc 0`

The file icos_ESMF_unstruct_six_corners_INTERPOL.nc contains the centers of the cells, the corners of the cells, the mask for the cells and the connectivity. The connectivity describes how the nodes are connected together to form each element (see *Appendix B*).

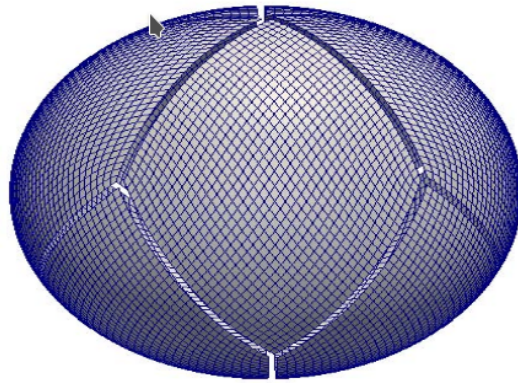


Figure 1: Icosahedral grid icos

3.3 Creation of the unstructured gaussian reduced grid ssea

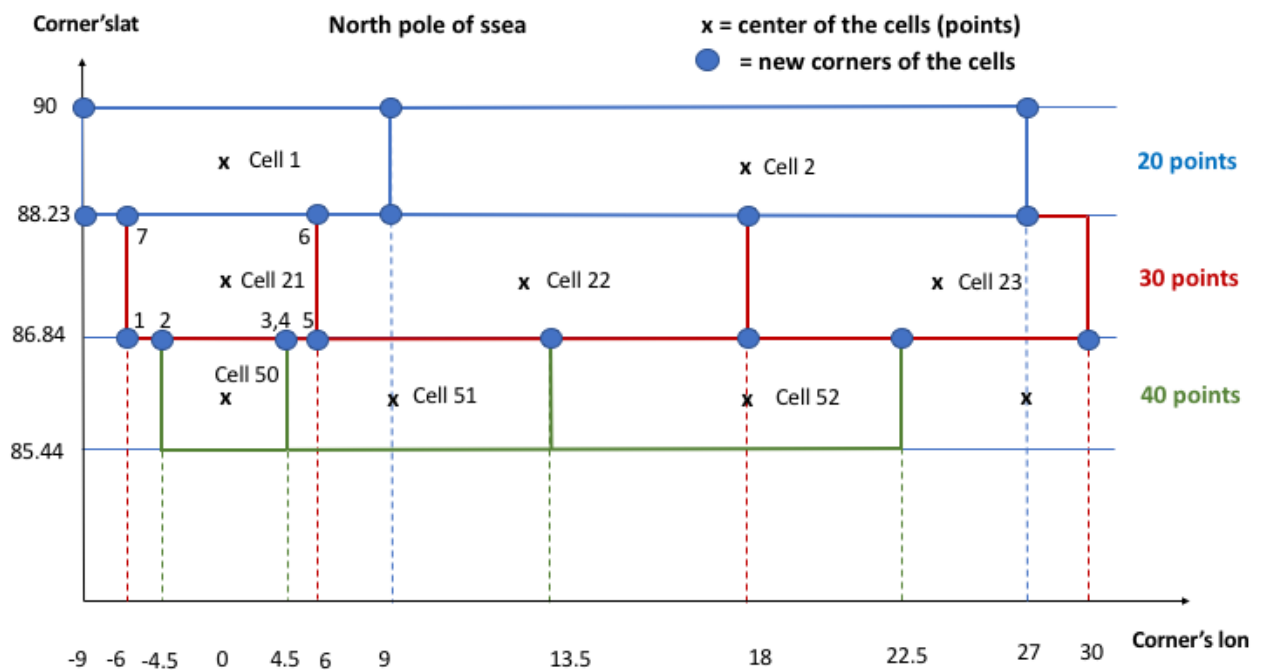


Figure 2: ssea gaussian reduced grid structure at the North pole

The original grid has 4 corners defined for each cell. But as the number of cells is not the same for each latitude, except at the equator, the corners are discontinuous in space. This is a problem when performing conserve remapping using ssea (8- *Senhaji (2016)*).

This is why a new grid was constructed taking into account, at a given latitude, the corners of the cells below and the corners of the cells above to create the new corners of each cell, as shown on figure 2.

For each cell at a given latitude four arrays are first constructed. Two of them store the longitudes and the latitudes of all the corners of the cells above the current cell contained in the current cell, as well as the corners of the current cell, and two of them contain the

longitudes and latitudes of all the corners of the cells below the current cell contained in the current cell, as well as the corners of the current cell.

The cells of the first line (latitude = +88.92°) and the last line (latitude = -88.92°) of ssea are special as they do not have cells above or below. The upper corners of the cells of the first line are the third and fourth corners of the initial grid, while the lower corners of the cells of the last line are the first and second corners of the initial grid.

The result of the algorithm is that there are maximum 12 corners in one cell for ssea.

For example, cell 21, located at lon=0° and lat=87.54°, will now have five corners below and two corners above. The third corner of the cell 50 is counted twice as it is also the fourth corner of cell 51 and it is contained between the initial corners of cell 21.

Then the array of longitudes of the corners above and below are merged, as well as the array of latitudes of the corners above and below so that the new corners of the grid are in counterclockwise order.

To create the unstructured gaussian reduced grid ssea at the ESMF format, a SCRIP file was first created. To be able to construct the new ssea grid at the SCRIP format, it was necessary that all the cells have the same number of corners (see *Appendix B*). So the last corner of each cell was duplicated if necessary, so to have twelve corners for each cell for the new ssea grid. Finally the executable **ESMF_Scrip2Unstruct** was used to obtain the ESMF unstructured grid for ssea using the command:

- `./ESMF_Scrip2Unstruct grid_SCRIP_ssea.nc
ssea_ESMF_unstruct_12_corners_INTERPOL.nc 0`

It was not possible to calculate the area of the grid during the ESMF conservative remapping. The error returned by ESMF is that it was not possible to compute the area of a polygon containing fewer than 3 corners. It is certainly linked to the structure of the new ssea grid. So the area of the sse grid was added to the unstructured grid file using the file `areas_INTERPOL.nc` of OASIS3-MCT, and the option `-user_areas` was used for remapping (see section 7.2 and *Appendix C*).

4 Creation of the ESMFv7.1.0r remapping weights

As the grids tested in this study are not high or very high resolution grids, the ESMF remapping weights were calculated using only one process of kraken. Nevertheless, it is possible to calculate the ESMF weights in parallel. We only present here the commands used to calculate the **ESMF remapping weights from nogt to bggd**.

The commands used to calculate the ESMF remapping weight files from bggd to nogt, between bggd and icos and between bggd and ssea are given in *Appendix C*.

To calculate the remapping files between structured grids, it is not necessary to specify which part of the grid is used for regridding for non-conservative remapping. It is always the centers of the grids that are used. For bilinear and patch remapping the following command is used to calculate the ESMF remapping weight file from nogt to bggd:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
 - **-source** nogt_SCRIP_NCL_MR_INTERPOL.nc \
 - **-destination** bggd_SCRIP_NCL_BR_INTERPOL.nc \
 - **-method** remap_method - **-weight** weights-file-name \
 - **-extrap_method** neareststod - **-ignore_degenerate**

It is necessary to use the option `- -ignore_degenerate` because the source grid nogt contains a cell that has corners close enough that the cell collapse to a line or point (ESMF remapping error message). For neareststod the option `- -ignore_degenerate` can be removed.

For conservative remapping (`conserve` or `conserve2nd`), there is no extrapolation method for unmapped unmasked destination points. It is necessary to replace the option `-extrap_method` by `-ignore_unmapped` to avoid an error. It is also necessary to use the normalization option `fracarea` (for which the destination field is divided by the corresponding destination fraction) to get reasonable values for cells which are only partially covered by the source grid. If the default option `destarea` is used, there are some unreasonable values (in the physical sense even if the conservation is locally ensured) of the remapping field near the cost. The command used is:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
 - **-source** nogt_SCRIP_NCL_MR_INTERPOL.nc \
 - **-destination** bggd_SCRIP_NCL_BR_INTERPOL.nc \
 - **-method** remap_method - **-weight** weights-file-name \
 - **-ignore_unmapped** - **-norm_type** fracarea

The commands used to create the ESMF remapping files from bggd to nogt, between bggd and icos and between bggd and ssea are described in *Appendix C*.

To calculate the interpolated fields with OASIS3-MCT using the ESMF remapping weight files, the weight files must be adapted to the OASIS3-MCT format, so the names of the variables in the files were modified using a shell script after their creation.

5 ESMF v7.1.0r remapping results between bggd and nogt

A new environment, **test_rmp_esmf_with_oasis**, was created from the toy test_interpolation of OASIS3-MCT, to calculate the remapping fields using the ESMF remapping weights. Test_rmp_esmf_with_oasis is a toy model coupling model1 and model2 defined on two different grids. An analytical coupling field, see figure 3, is sent at t=0 by model 1 through OASIS3-MCT to model2. The field is remapped from the source grid to the target grid using the ESMF remapping weight file.

As for ESMFv6.3.0rp1 (8- *Senhaji (2016)*) the received field and the error are post-processed in model2. The interpolated field is set to 10000. on masked target points and is set to 1.E20 on unmasked target points that do not receive any value.

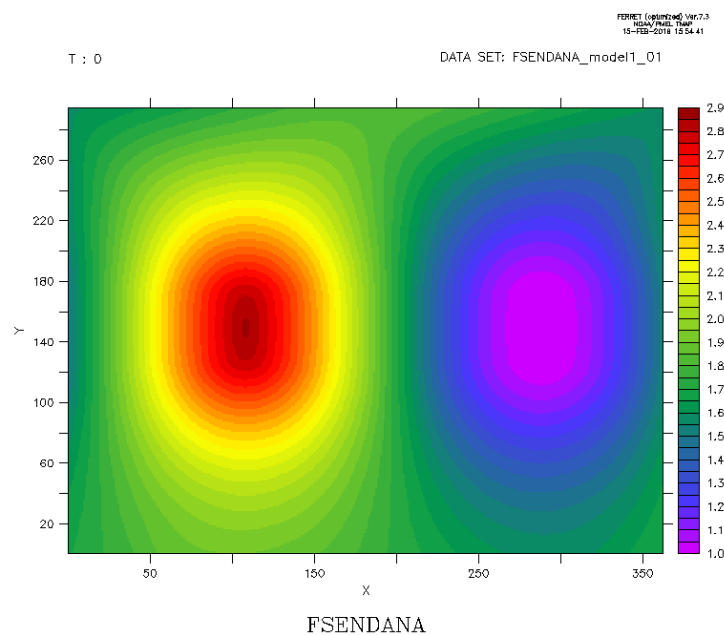


Figure 3: Analytical coupling field sent by model1 to model2

The error of interpolation is calculated on the target grid from the remapping field defined on unmasked target points that receive a value by:

- **Error = ABS(((field_analytical – field_received)/field_received))*100**

On masked points the error is set to -10000. and on unmasked target points that do not receive any value the error is set to -1.E20 .

The post-processed remapping field and error are one dimensional arrays. To be plotted using Ferret, they are projected on the latitudes and the longitudes of the different target grids (see the figures below).

The results of remapping between the structured grids nogt and bggd are very similar to the ones obtained with ESMF v6.3.0rp1 (8- *Senhaji (2016)*).

As there is an extrapolation method for non-conservative remapping, neither the interpolated field nor the error have some points at 1.E20 or -1.E20. On the opposite, as there is no extrapolation option for unmasked target points that do not receive any value for first and second order conservative remapping, there are some values of the error at -1.E20.

5.1 Results for the non-conservative remapping between bggd and nogt

5.1.1 Results from nogt to bggd

For neareststod, bilinear and patch remapping from nogt to bggd, the maximum of the error is of order of 10% and is located in the Hudson bay and the Caspian sea, as in (8- *Senhaji (2016)*). Figure 4 represents the received field and the error for neareststod remapping from nogt to bggd.

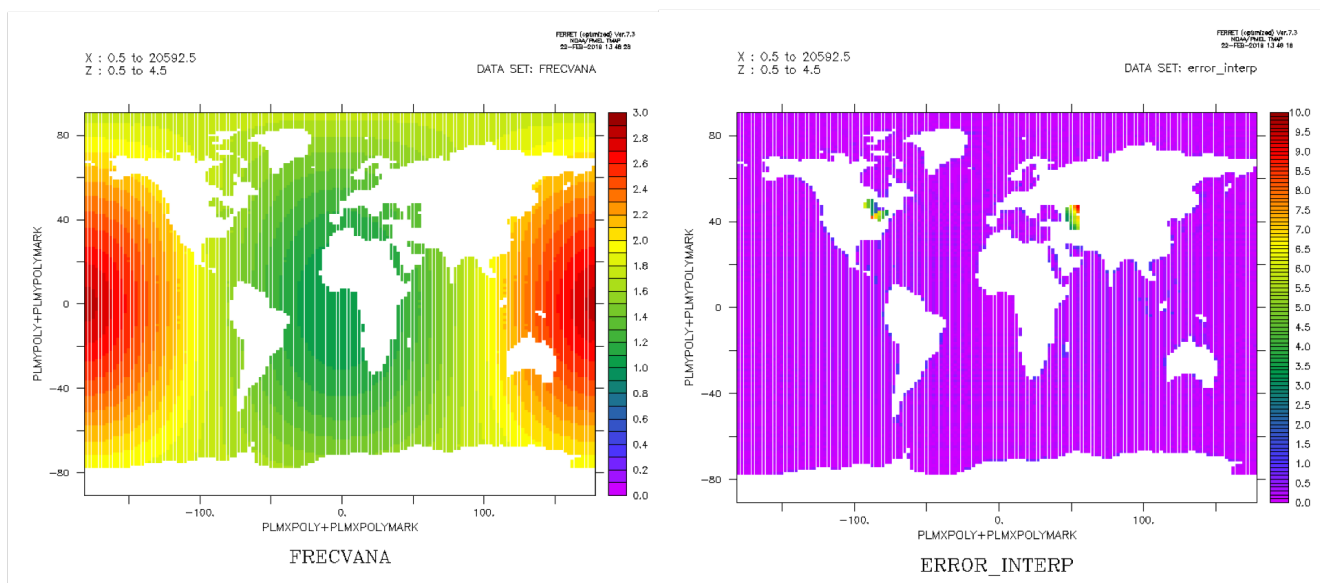


Figure 4: Neareststod remapping from nogt to bggd using ESMF

5.1.2 Results from bggd to nogt

For neareststod, bilinear and patch remapping from bggd to nogt, the error is of order of 1% as in (8- *Senhaji (2016)*). Figure 5 represents the received field and the error for neareststod remapping from bggd to nogt.

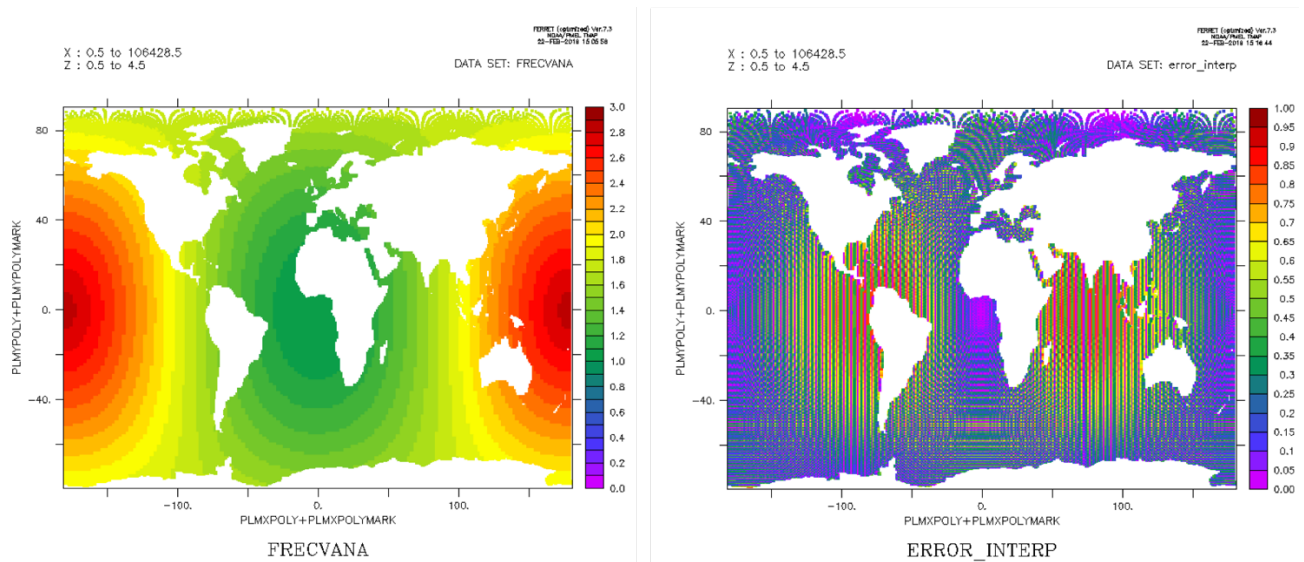


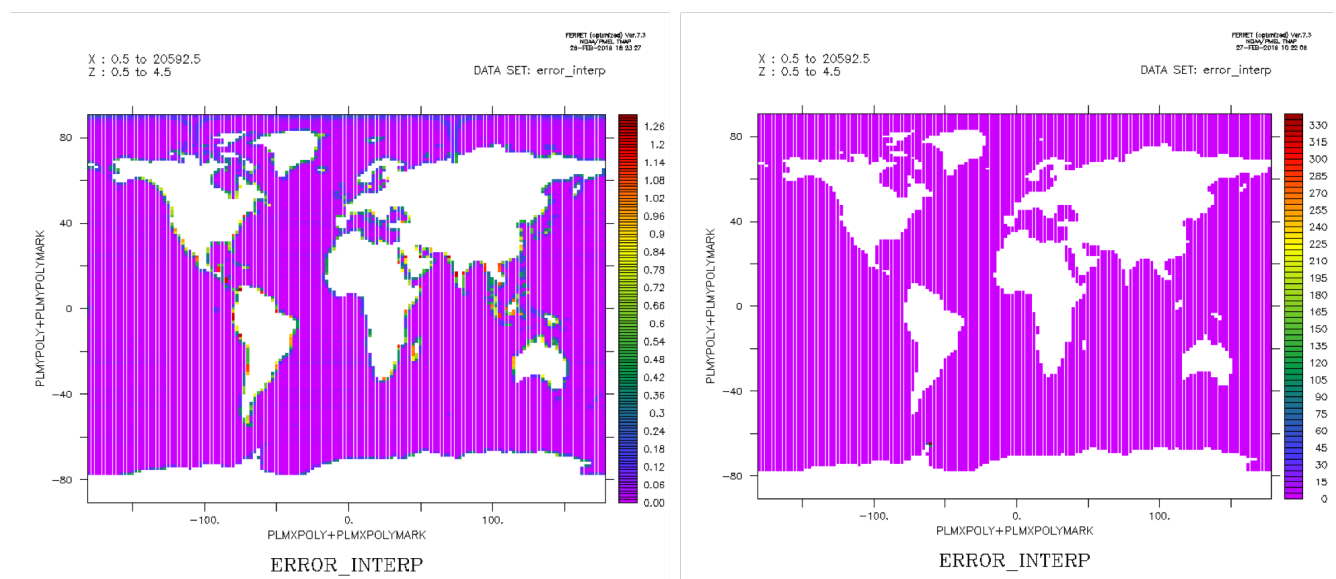
Figure 5: Neareststod remapping from bggd to nogt using ESM

5.2 Results for the conservative remapping between bggd and nogt

The remapped fields look like the one of figure 5. Only the error calculated on unmasked target points that receive a value is plotted below for first and second order conservative remapping.

5.2.1 Results from nogt to bggd

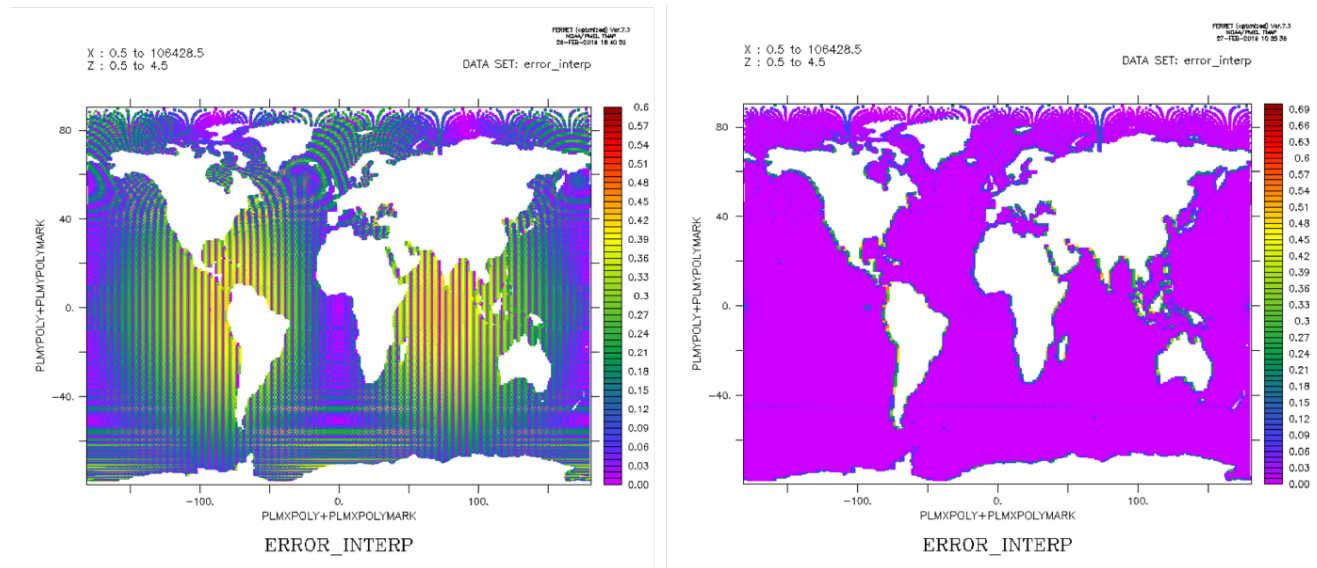
For first order conservative remapping from nogt to bggd, the maximum of the error calculated on unmasked target points that receive a value is of order of 1.3%. The error is plotted on figure 6. The pattern is very similar to what was observed in (8- *Senhaji (2016)*). For second order conservative remapping from nogt to bggd, the error calculated on unmasked target points that receive a value is of order of 300% because some unmasked target points have wrong values at the South.



**Figure 6: Error for conservative remapping from nogt to bggd using ESMF:
left-first order ; right-second order**

5.2.2 Results from bggd to nogt

For first and second order conservative remapping from bggd to nogt the maximum of the error calculated on unmasked target points that receive a value is of order of 0.6% for both methods. The pattern of the error of first order conservative remapping is different from what was observed in (8- *Senhaji (2016)*). There is no more high error at the North pole.



**Figure 7: Error for conservative remapping from bggd to nogt using ESM:
left-first order ; right-second order**

6 ESMF v7.1.0r remapping results between bggd and icos

As for section 5, the test_rmp_esmf_with_oasis environment was used to calculate the remapping fields with the ESMF weights. As there is an extrapolation method for non-conservative remapping, neither the interpolated field nor the error have some points at 1.E20 or -1.E20. On the opposite, as there is no extrapolation option for unmasked target points that do not receive any value for first and second order conservative remapping, there are some values of the error at -1.E20.

6.1 Results for non-conservative remapping between bggd and icos

6.1.1 Results from bggd to icos

For neareststod, bilinear and patch remapping from bggd to icos, the error is of order of 15%. The received field and the error are plotted on figure 8 for patch remapping from bggd to icos.

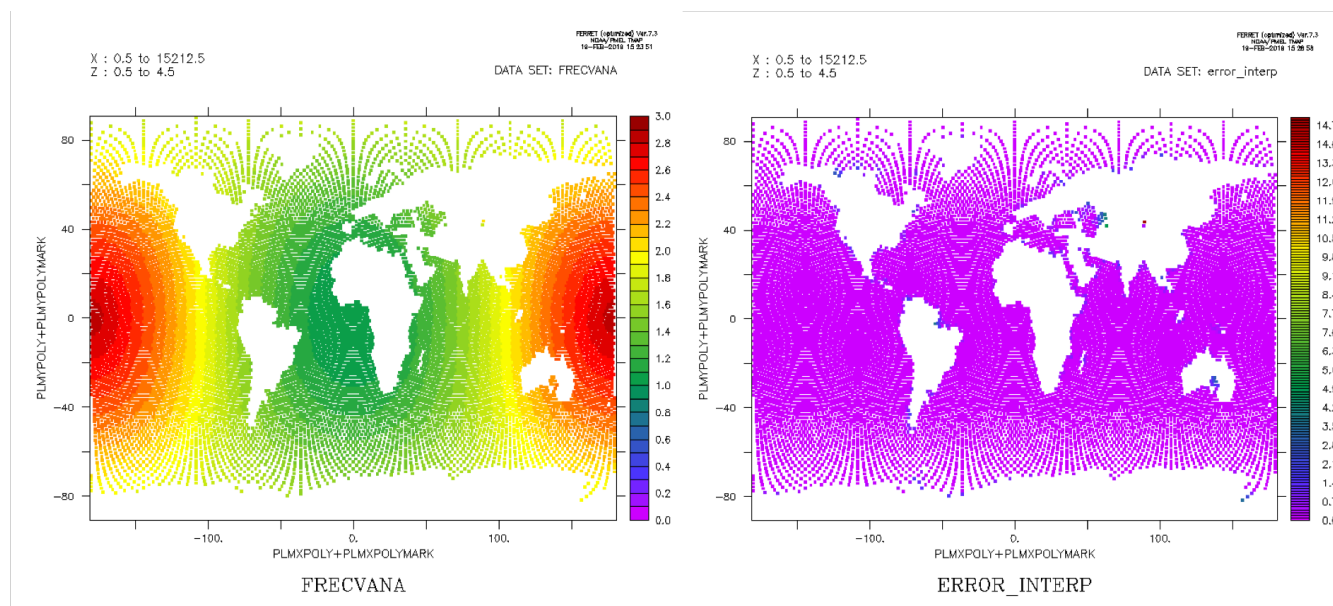


Figure 8: Patch remapping from bggd to icos using ESMF

6.1.2 Results from icos to bggd

For neareststod, bilinear and patch remapping from icos to bggd, the error is of order of 7%. The received field and the error are plotted on figure 9 for patch remapping from icos to bggd.

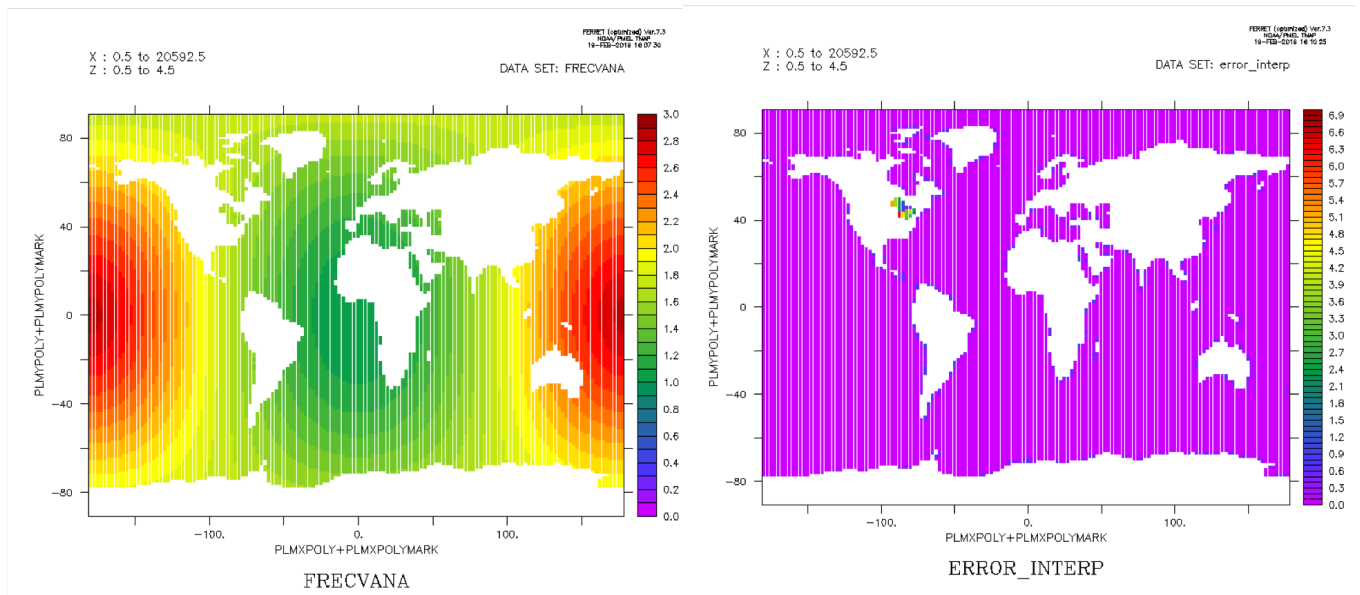


Figure 9: Patch remapping from icos to bgd using ESMF

6.2 Results for conservative remapping between bgd and icos

The remapped fields look like the one of figure 9. Only the error calculated on unmasked target points that receive a value is plotted below for first and second order conservative remapping.

6.2.1 Results from bgd to icos

For first order conservative remapping, the maximum of the error calculated on unmasked target points that receive a value is of order of 1.6%. For second order conservative remapping, the maximum of the error calculated on unmasked target points that receive a value is of order of 36% because there is one point that receives a wrong value. This will have to be investigated.

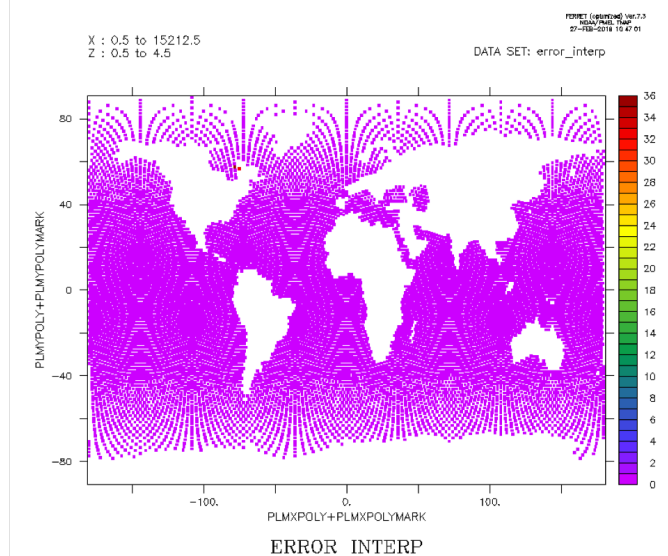
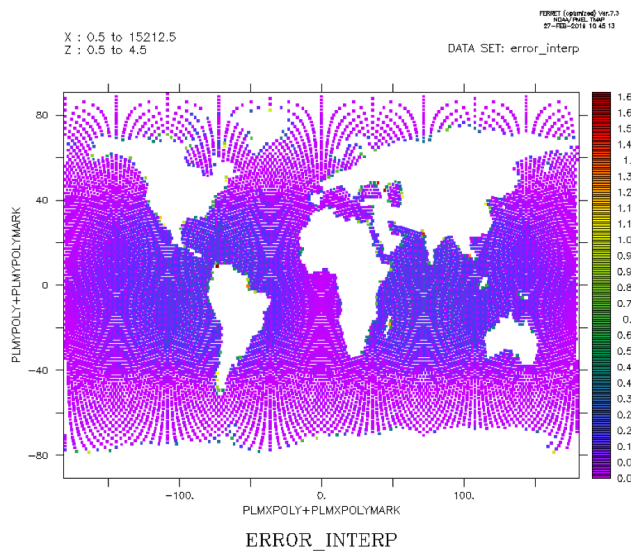


Figure 10: Error for conservative remapping from bggd to icos using ESMF: left-first order ; right-second order

6.2.2 Results from icos to bggd

For first order conservative remapping the error calculated on unmasked target points that receive a value is of order of 1.3%. For second order conservative remapping the error calculated on unmasked target points that receive a value is of order of 1.05%.

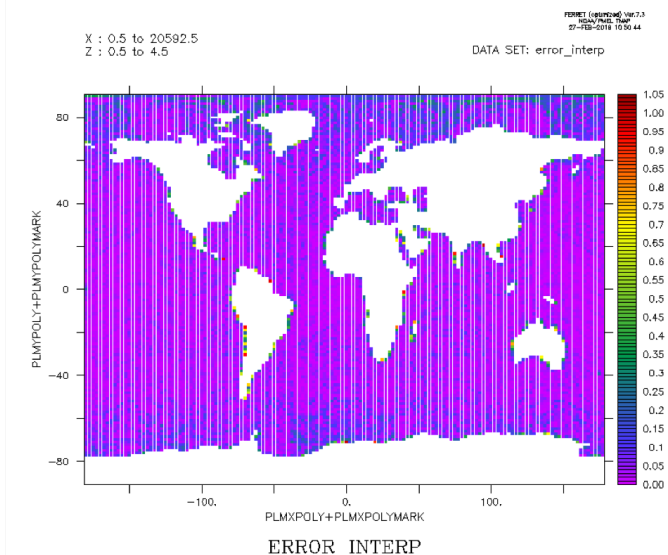
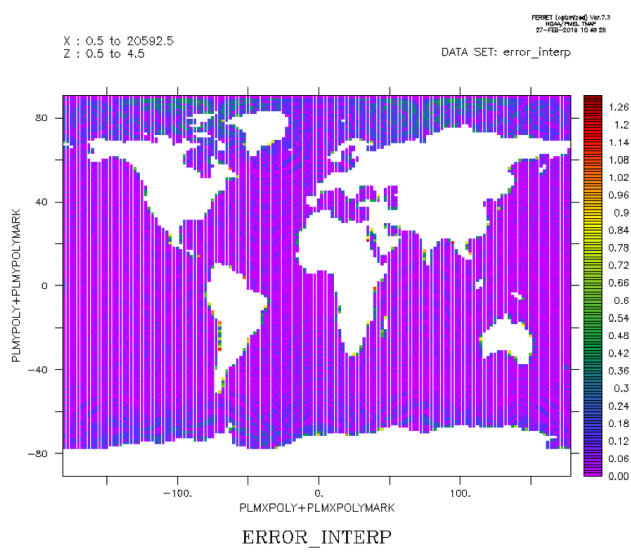


Figure 11: Error for conservative remapping from icos to bggd using ESMF: left-first order ; right-second order

7 ESMF v7.1.0r remapping results between bggd and ssea

As for section 5 and 6, the test_rmp_esmf_with_oasis environment was used to calculate the remapping fields with the ESMF weights. As there is an extrapolation method for non-conservative remapping, neither the interpolated field nor the error have some points at 1.E20 or -1.E20. On the opposite, as there is no extrapolation option for unmasked target points that do not receive any value for first and second order conservative remapping, there are some values of the error at -1.E20.

7.1 Results for non-conservative remapping between bggd and ssea

7.1.1 Results from bggd to ssea

For neareststod, bilinear and patch remapping from bggd to ssea, the error is of order of 3%. The received field and the error are plotted on figure 12 for bilinear remapping from bggd to ssea.

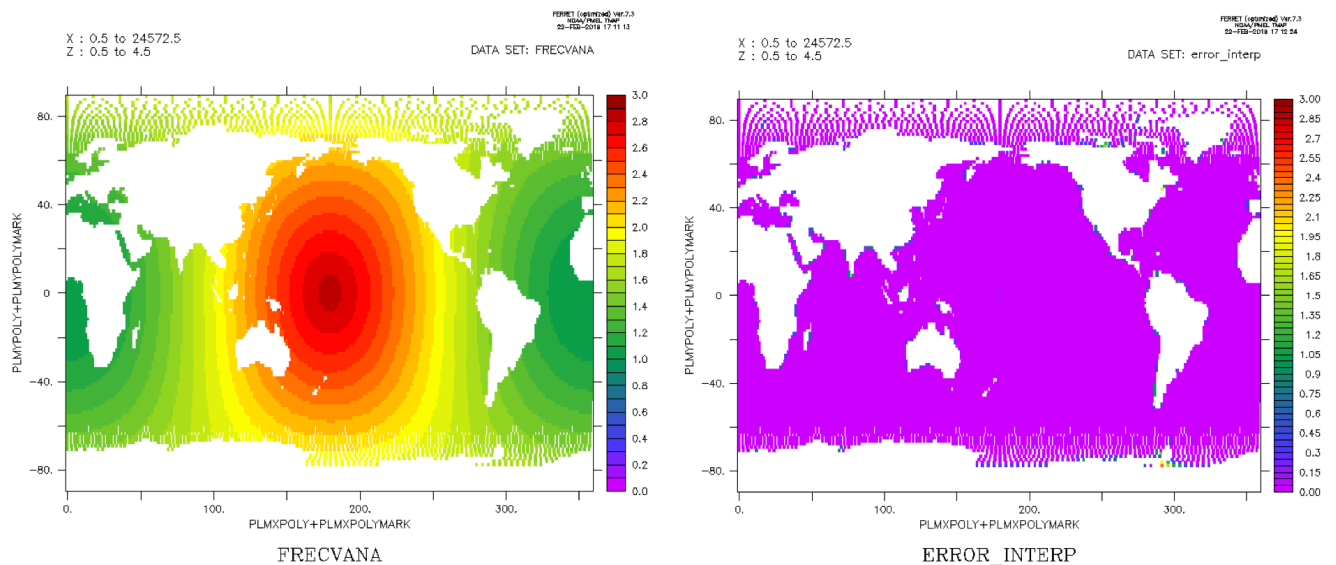


Figure 12: Bilinear remapping from bggd to ssea using ESMF

7.1.2 Results from ssea to bggd

For neareststod, bilinear and patch remapping from ssea to bggd, the error is of order of 10%. The received field and the error are plotted on figure 13 for bilinear remapping from ssea to bggd.

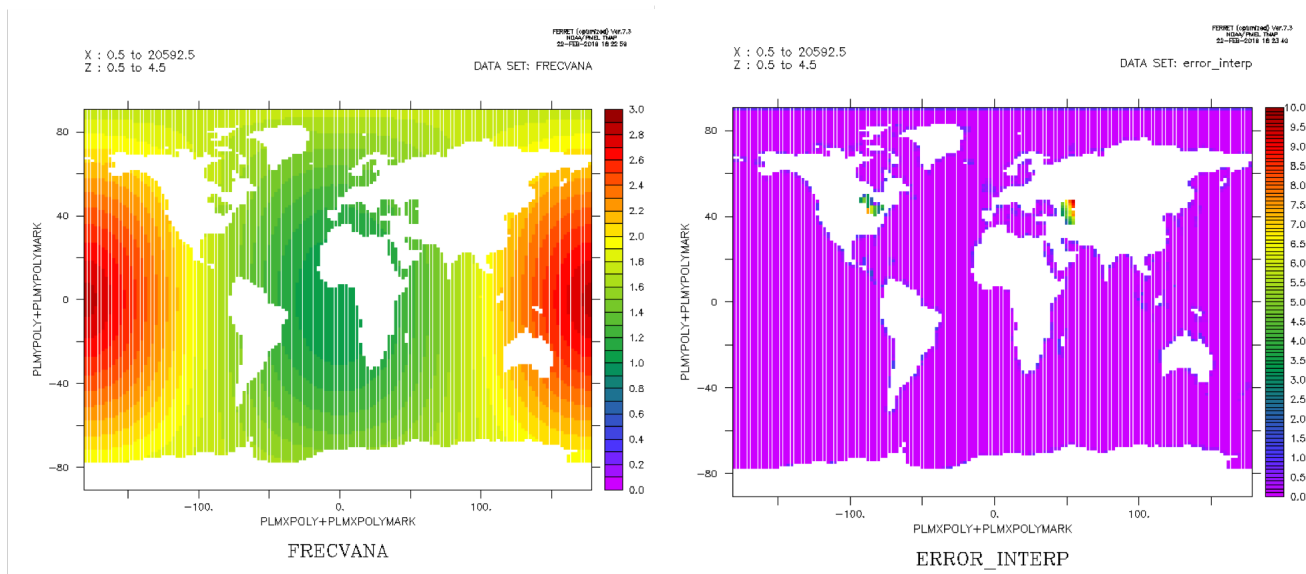


Figure 13: Bilinear remapping from ssea to bggd using ESMF

7.2 Results for conservative remapping between bggd and ssea

The remapped fields look like the one of figure 13. Only the error calculated on unmasked target points that receive a value is plotted below for first and second order conservative remapping.

7.2.1 Results from bggd to ssea

For first order conservative remapping the error calculated on unmasked target points that receive a value is of order of 1%. For second order conservative remapping the error calculated on unmasked target points that receive a value is of order of 0.6%.

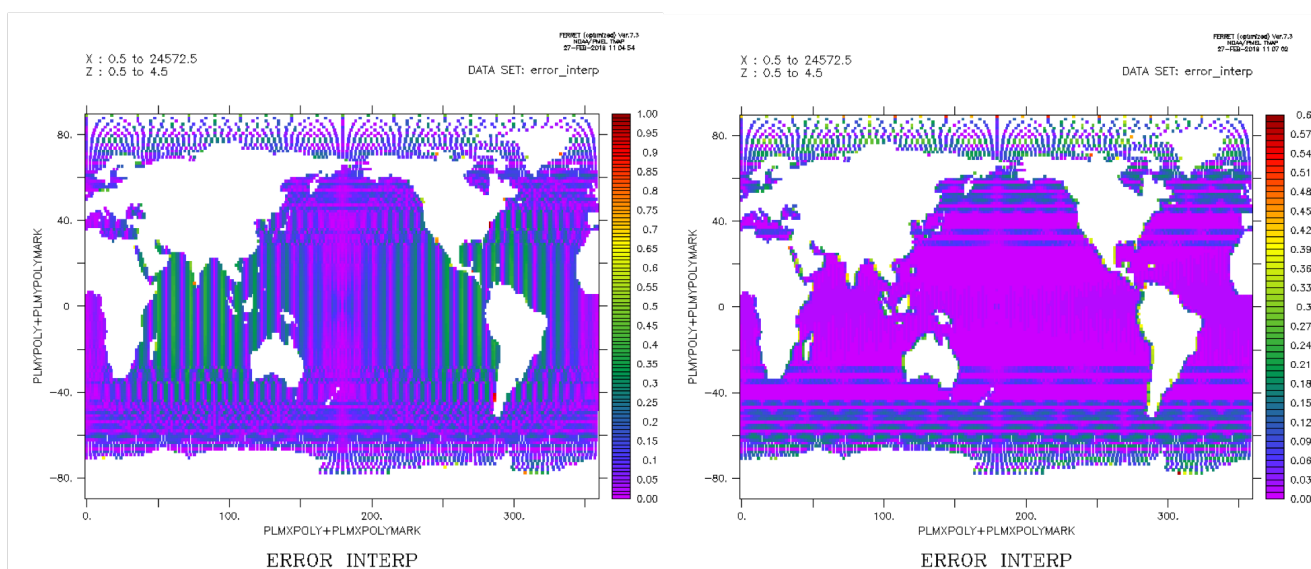


Figure 14: Error for conservative remapping from bggd to ssea using ESMF: left-first order ; right-second order

7.2.2 Results from ssea to bggd

For first order conservative remapping the maximum of the error calculated on unmasked target points that receive a value is of order of 1.5%. For second order conservative remapping it was not possible to create the ESMF remapping weight file. The error given by ESMF will have to be investigated in more details in the future in collaboration with the ESMF support but it may be linked to the fact that it is not possible to calculate the gradients of the field on ssea, as for OASIS3-MCT.

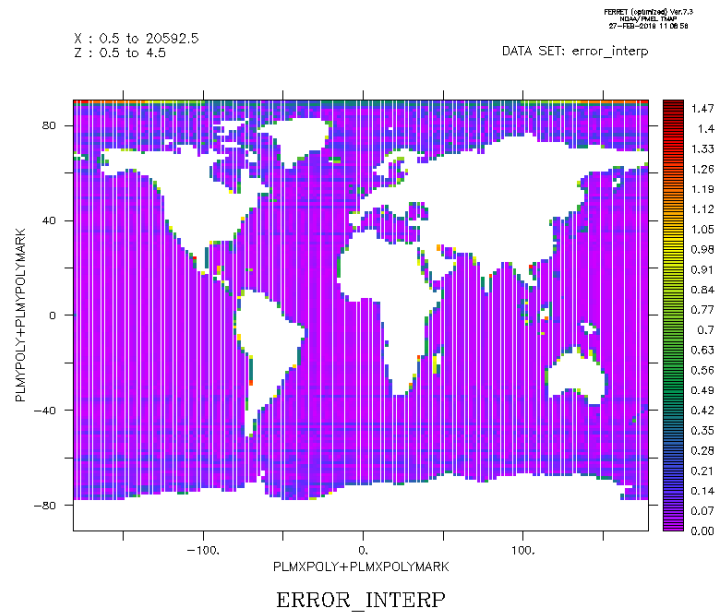


Figure 15: Error for first order conservative remapping from ssea to bggd using ESMF

8 Conclusion and perspective

The first part of the document describes the creation of the structured and unstructured grids tested in this study, at the ESMF format. Then the commands used to create the ESMF remapping weights are presented.

The results obtained are discussed. Only one interpolation results for non-conservative between the different grids are presented. The error calculated on unmasked target points that receive a value are plotted for conservative remapping between all grids. For remapping between structured grids, the results are very similar to the ones analyzed in (8- *Senhaji (2016)*). There is a problem with the second order remapping from nogt to bggd as some destination points do not receive a correct value. For remapping results between unstructured and structured grids, the error is of the same order of the one for remapping between structured grids, except for the second order remapping from bggd to icos where some destination points also receive wrong values.

Compare to what was done in (8- *Senhaji (2016)*), an unstructured gaussian grid ssea was successfully constructed for conservative remapping. It was possible to do remapping tests between two unstructured grids (icos and ssea) and a structured grid (bggd).

This work will be followed by other studies. It would be interesting to compare in details the results obtained using the SCRIP library of OASIS3-MCT and using ESMF when doing remapping tests with unstructured grids.

Other libraries, like XIOS (7-*Kritsikis E. et al. (2017)*), or YAC (10-*Hanke, M. et al (2016)*), or CWIPI (11-*Duchaine, F. et al (2015)*) are also planned to be tested and compared to the SCRIP library of OASIS3-MCT.

Acknowledgements

We would like to thanks Robert Oehmke (ESMF, University of Colorado) and Mary Haley (NCL, NCAR) for their help since 2015 on ESMF regrdding.

We would like to thanks Isabelle D'Ast from Cerfacs who helped us to create the submission scripts for the tests on Cerfacs computer Kraken_Lenovo.

This research was supported by the ESiWACE H2020 European project, grant agreement no. 675191 (www.esiwace.eu), the IS-ENES2 FP7 European project, contract number 312979 and the CONVERGENCE project funded by the French National Research Agency: ANR-13-MONU-0008.

9 Bibliography

- 1- Valcke, S., Coquart, L. and Jonville, G. (2018) 9, MS2.4: High quality interpolation weight and parallel coupling available to IPSL-CM and CNRM-CM, CECI, Université de Toulouse, CNRS, CERFACS - TR-CMGC-18-91, Toulouse, France, Technical report
- 2- Valcke, S., Craig, A. and Coquart, L. (2018), OASIS3-MCT User Guide, OASIS3-MCT4.0, CECI, Université de Toulouse, CNRS, CERFACS - TR-CMGC-18-77, Toulouse, France, Technical report
- 3- Piacentini, A., Maisonnave, E., Jonville, G., Coquart, L. and Valcke, S. (2018), A parallel SCRIP interpolation library for OASIS, CECI, Université de Toulouse, CNRS, CERFACS - WN-CMGC-18-34, Toulouse, France, Working note
- 4- ESMF Reference Manual for Fortran, Version 7.1.0r (2018):
http://www.earthsystemmodeling.org/esmf_releases/public/ESMF_7_1_0r/ESMF_refdoc/
- 5- NCL, The NCAR Command Language (Version 6.5.0) [Software]. (2018).
Boulder, Colorado: UCAR/NCAR/CISL/TDD. <http://dx.doi.org/10.5065/D6WD3XH5>
- 6- Craig, A., Valcke, S. and Coquart, L. (2017) Development and performance of a new version of the OASIS coupler, OASIS3-MCT 3.0, Geoscientific Model Development, 10, pp. 3297-3308, doi:10.5194/gmd-10-3297-2017
- 7- Kritsikis, E., Aechtner, M., Meurdesoif, Y., and Dubos, T. (2017), Conservative interpolation between general spherical meshes, Geosci. Model Dev., 10, 425-431, <https://doi.org/10.5194/gmd-10-425-2017>
- 8- Senhaji, H. (2016), Evaluation et comparaison des interpolations numériques des bibliothèques SCRIP et ESMF, UMR 5318 CECI, CERFACS-CNRS, WN-CMGC-16-227, Toulouse, France, working note
- 9- Theurich, G., Deluca, C., Campbell, T., Liu, F., Saint, K., Vertenstein, M., Chen, J et al. (2016): The Earth System Prediction Suite: Toward a Coordinated U.S. Modeling Capability, B. Am. Meteor. Soc., 97, pp. 1229–1247, <https://doi.org/10.1175/BAMS-D-14-00164.1>
- 10- Hanke, M., Redler, R., Holfeld, T., and Yastremsky, M. (2016), YAC 1.2.0: new aspects for coupling software in Earth system modelling, Geosci. Model Dev., [9, 2755-2769](https://doi.org/10.5194/gmd-9-2755-2016), <https://doi.org/10.5194/gmd-9-2755-2016>
- 11- Duchaine, F., Jaure, S., Poitou, D., Quemerais, E., Staffelbach, G., Morel, T., and Gicquel, L. (2015), Analysis of High Performance Conjugate Heat Transfer with the OpenPALM Coupler. [Journal of Computational Science and Discovery. 8:015003, 2015.](https://doi.org/10.1145/275003)
- 12- Boumediene, Y., Valcke, S. and Coquart, L. (2015), Les performances des coupleurs OASIS3-MCT et OpenPALM pour les grilles icosaèdres, URA SUC 1875, CERFACS-CNRS, TR-CMGC-15-29024, Toulouse, France, technical report
- 13- Jones PW (1998) A user's guide for SCRIP: a spherical coordinate remapping and interpolation package, Los Alamos National Laboratory.

10 Appendix A: Compiling and installing ESMF v7.1.0r on Kraken (Cerfacs Lenovo computer)

The web address to download ESMF v7.1.0r is:

https://www.earthsystemcog.org/projects/esmf/download_710r

To be able to compile and install correctly ESMF on Kraken, the following ESMF environment variables must be defined in the file *\$HOME/.bashrc* :

```
export ESMF_DIR=/scratch/globc/coquart/esmf

export ESMF_COMPILER=intel

export ESMF_INSTALL_PREFIX=/scratch/globc/coquart/esmf/install_dir

export ESMF_INSTALL_HEADERDIR=include

export ESMF_INSTALL_MOODIR=mod/modg/Linux.intel.64.mpiuni.default

export ESMF_INSTALL_LIBDIR=lib/libg/Linux.intel.64.mpiuni.default

export ESMF_INSTALL_BINDIR=bin/bing/Linux.intel.64.mpiuni.default

export ESMF_INSTALL_DOCDIR=doc

# Definition of ESMF_NETCDF (needed to read the grid files and create the remapping files)

export ESMF_NETCDF=split

export ESMF_NETCDF_LIBPATH="/softs/local_intel/netcdf-fortran/4.4.4_phdf5/lib -
L/softs/local_intel/netcdf-c/4.6.1_phdf5/lib"

export ESMF_NETCDF_INCLUDE="/softs/local_intel/netcdf-fortran/4.4.4_phdf5/include -
I/softs/local_intel/netcdf-c/4.6.1_phdf5/include"
```

To compile esmf on kraken : cd \${ESMF_DIR} ; gmake clean ; gmake lib

To install esmf after compilation on kraken : cd \${ESMF_DIR} ; gmake install

To see the ESMF variables environment : cd \${ESMF_DIR} ; gmake info

11 Appendix B: Grids format for ESMFv7.1.0r

There are two kind of grid file format for ESMF:

- The SCRIP format to describe structured grids.
- The ESMF unstructured grid file format to describe the unstructured grids. This format is indeed more compatible than the SCRIP format with the methods used to create an ESMF Mesh object. The ESMF format is also more efficient than SCRIP for these grids when used with ESMF.

11.1 Structured grids

Below is the header of the NetCDF grid file for the structured bggd grid:

```
netcdf bggd_SCRIP_NCL_BR_INTERPOL {  
  dimensions:  
    grid_size = 20592 ;  
    grid_corners = 4 ;  
    grid_rank = 2 ;  
  variables:  
    int grid_dims(grid_rank) ;  
    double grid_center_lat(grid_size) ;  
    grid_center_lat:units = "degrees" ;  
    double grid_center_lon(grid_size) ;  
    grid_center_lon:units = "degrees" ;  
    int grid_imask(grid_size) ;  
    grid_imask:units = "unitless" ;  
    double grid_corner_lat(grid_size, grid_corners) ;  
    grid_corner_lat:units = "degrees" ;  
    double grid_corner_lon(grid_size, grid_corners) ;  
    grid_corner_lon:units = "degrees" ;  
  // global attributes:  
  :date_created = "Tue Jan 15 16:19:59 CET 2019" ;  
  :Createdby = "ESMF_regridding.ncl" ;  
  :Conventions = "SCRIP" ;  
  :title = "curvilinear_to_SCRIP (143,144)" ;  
}
```

The `grid_size` dimension is the total number of cells in the grid; `grid_rank` refers to the number of dimensions. In this case `grid_rank` is 2 for a 2D logically rectangular grid. The integer array `grid_dims` gives the number of grid cells along each dimension. The number of corners (vertices) in each grid cell is given by `grid_corners`. The grid corner coordinates need to be listed in an order such that the corners are in counterclockwise order. Also, note that if your grid has a variable number of corners on grid cells, then you should set `grid_corners` to be the highest value and use redundant points on cells with fewer corners.

The integer array `grid_imask` is used to mask out grid cells which should not participate in the regridding. The array values should be zero for any points that do not participate in the regridding and one for all other points. Coordinate arrays provide the latitudes and longitudes

of cell centers and cell corners. The unit of the coordinates can be either "radians" or "degrees".

11.2 Unstructured grids

Below is the header of the NetCDF grid file for the unstructured icos grid:

```
netcdf icos_ESMF_unstruct_six_corners_INTERPOL {  
  dimensions:  
    nodeCount = 30420 ;  
    elementCount = 15212 ;  
    maxNodePElement = 6 ;  
    coordDim = 2 ;  
  variables:  
    double nodeCoords(nodeCount, coordDim) ;  
    nodeCoords:units = "degrees" ;  
    int elementConn(elementCount, maxNodePElement) ;  
    elementConn:long_name = "Node indices that define the element connectivity" ;  
    elementConn:_FillValue = -1 ;  
    byte numElementConn(elementCount) ;  
    numElementConn:long_name = "Number of nodes per element" ;  
    double centerCoords(elementCount, coordDim) ;  
    centerCoords:units = "degrees" ;  
    int elementMask(elementCount) ;  
  // global attributes:  
  :gridType = "unstructured mesh" ;  
  :version = "0.9" ;  
  :inputFile = "grid_SCRIP_icos.nc" ;  
  :timeGenerated = "We" ;  
}
```

The `nodeCount` dimension is the number of nodes in the mesh. The `elementCount` dimension is the number of elements in the mesh. The `maxNodePElement` dimension is the maximum number of nodes in any element in the mesh. The `coordDim` dimension is the number of dimensions of the points making up the mesh (i.e. the spatial dimension of the mesh). For example, a 2D planar mesh would have `coordDim` equal to 2.

The `nodeCoords` variable contains the coordinates for each node. `nodeCoords` is a two-dimensional array of dimension `(nodeCount, coordDim)`. The `centerCoords` contains the coordinates of the mesh centers.

The `elementConn` variable describes how the nodes are connected together to form each element. For each element, this variable contains a list of indices into the `nodeCoords` variable pointing to the nodes which make up that element.

12 Appendix C : ESMF v7.1.0r remapping files

12.1 Creation of ESMF v7.1.0r remapping files from bggd to nogt

For neareststod, bilinear and patch remapping the following command is used to calculate the ESMF remapping weight file from bggd to nogt:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
 - **-source** bggd_SCRIP_NCL_BR_INTERPOL.nc \
 - **-destination** nogt_SCRIP_NCL_MR_INTERPOL.nc \
 - **-method** remap_method - **-weight** weights-file-name \
 - **-extrap_method** neareststod

For conservative remapping (conserve or conserve2nd) the command used is:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
 - **-source** bggd_SCRIP_NCL_BR_INTERPOL.nc \
 - **-destination** nogt_SCRIP_NCL_MR_INTERPOL.nc \
 - **-method** remap_method - **-weight** weights-file-name \
 - **-ignore_unmapped** - **-norm_type** fracarea

12.2 Creation of ESMF v7.1.0r remapping files between bggd and icos

When the grid is unstructured at the ESMF format, it is necessary to specify which part of the grid is used for regridding for non-conservative remapping. For conservative regridding with unstructured grid, only the center location is supported (4- *ESMF Reference Manual for Fortran, Version 7.1.0r* (2018)).

As the mask is defined on the centers of the grid, we used the center of the icos grid to calculate the remapping weight files.

If icos is the destination grid, the following command is used to calculate the ESMF remapping files for neareststod, bilinear and patch:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
 - **-source** bggd_SCRIP_NCL_BR_INTERPOL.nc \
 - **-destination** icos_ESMF_unstruct_six_corners_INTERPOL.nc \
 - **-method** remap_method - **-weight** weights-file-name \
 - **-extrap_method** neareststod - **-dst_loc** "center"

If icos is the source grid, we use the command:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
 - **-source** icos_ESMF_unstruct_six_corners_INTERPOL.nc \
 - **-destination** bggd_SCRIP_NCL_BR_INTERPOL.nc \
 - **-method** remap_method - **-weight** weights-file-name \
 - **-extrap_method** neareststod - **-src_loc** "center"

For conservative remapping of first or second order (conserve or conserve2nd) from icos to bggd we use:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
`- -source icos_ESMF_unstruct_six_corners_INTERPOL.nc \`
`- -destination bggd_SCRIP_NCL_BR_INTERPOL.nc \`
`- -method remap_method - -weight weights-file-name \`
`- -ignore_unmapped - -norm_type fracarea`

Source and destination option are reverse if the conservative remapping is performed from bggd to icos.

12.3 Creation of ESMF v7.1.0r remapping files between bggd and ssea

The creation of the remapping files is exactly the same as for 11.1 except for conservative remapping. In this case we had an error linked to the calculation of the areas of the cells for our new grid so we choose to use the option `- -user_areas` to calculate the remapping weight files. The area must be given in the source and the destination grids. For example if ssea is the source grid we used the command:

- `time mpirun -np 1 ./ESMF_RegridWeightGen \`
`- -source ssea_ESMF_unstruct_12_corners_INTERPOL.nc \`
`- -destination bggd_SCRIP_NCL_BR_INTERPOL.nc \`
`- -method remap_method - -weight weights-file-name - -ignore_degenerate \`
`- -ignore_unmapped - -user_areas - -norm_type fracarea`

