NEMO regional ocean configuration worked example: WED025

Pierre Mathiot and Katherine Hutchinson, July 2022
DOI: 10.5281/zenodo.6817000

Note that this is a training guide which uses a regional configuration setup for the Weddell Sea. The instructions can be adapted to any region of the ocean by changing the bathymetry, coordinates, initial conditions and forcing files.

1. Background on the WED025 configuration

The Weddell Sea of Antarctica is an area of interest to many oceanographers, as it provides the backdrop for a range of complex and globally important ocean and crosphere processes. It is home to the production of over half of the world's densest and deepest water mass, Antarctic Bottom Water (AABW). The parent waters of AABW are formed on the Weddell's southern and western continental shelves via interactions between the ocean, atmosphere, sea-ice and the large floating bodies of ice: the Filchner Ronne and Larsen C ice shelves. Additionally, the Weddell Sea is an area of interesting temperature-salinity structure and dynamic water mass transformations. Most of the freshwater flux to this region is in the form of ice-shelf basal melt and iceberg melt as there is very little rain (precipitation is mostly in the form of snow) and no surface runoff in the form of rivers. Tides play an important role in the Weddell Sea by driving elevated levels of mixing between the dense shelf waters and ambient water masses on the continental shelf break and slope, and they drive increased melt under the ice shelves by strengthening the background flow speeds.

As an introduction to regional modeling using NEMO, we have chosen the Weddell Sea as our domain of interest given the range of processes at play. This demonstrator uses the official NEMO release version 4.2. We have adapted the WED025 configuration of Bull et al. (2021) which uses a curvilinear grid with a quasi-isotropic horizontal resolution equivalent to approximately 1/4° in latitude and longitude. The configuration simulates the circulation under most of the Weddell Sea ice shelves allowing for explicit ocean-ice shelf interactions, yet maintaining a static cavity geometry (i.e. the ice shelf shape is fixed with time). We have chosen to parameterize the basal melt of the 5 smallest ice shelves and input this freshwater at the depth range between the base of the shelf and the bathymetry at the location of the ice shelf front (Mathiot et al., 2017). In doing so this configuration has a mix of explicit and parameterized freshwater flux from ice shelf melt. The configuration has the ocean coupled to SI3, the sea ice model of NEMO, so that we have dynamic ocean- sea ice interactions. We parameterize the role of icebergs by inputting freshwater flux from the bergs according to an iceberg climatology from the global 1/4° GO7 model (Storkey et al., 2018) forced by JRA winds (GO7-JRA). The initial conditions provided in this demonstrator are output from year 2000 of the control simulation from Bull et al. (2021). The boundary forcings are from the global GO7-JRA simulation. The barotropic tidal forcing is the same as in Jourdain et al. (2019).

Information on the NEMO namelist and parameters can be found here:
https://doi.org/10.5281/zenodo.6334656

2. System prerequisites:

- Netcdf package installed on your HPC
- XIOS package installed on your HPC
- An arch file for your HPC (this just sets paths to XIOS, netcdf and the compiling option which depends on your compiler)
- Python installed on your HPC or personal computer (for analysis)

3. Provided:

- Topography, coordinates, boundary conditions, ocean and sea ice initial conditions, iceberg climatology, runoff and atmospheric forcing.
- Namelist to build the domain_cfg.nc file from the topography and coordinates file, and namelist for WED025 adapted to the demonstrator.

You need to copy the tarball of proved files into your HPC work directory and you need to extract the contents

- E.g. *unzip WED025_demonstrator_forcings.tar.zip*
- Then tar -xvf WED025_demonstrator_forcings.tar

You should have a folder WED025_demonstrator_forcings (save the path).

4. Instructions:

4.1 Compile XIOS:

Output in NEMO is managed via an external library called XIOS. It allows a lot of flexibility in the choice of the output variables and output frequencies (amongst other things). So before running NEMO we need to compile this library.

- Connect to your HPC system and go to your work directory.

- Download XIOS:

svn co http://forge.ipsl.jussieu.fr/ioserver/svn/XIOS/trunk *yourxiosdirname*

Setup your arch files (one for the environment (.env), another for the compiler (.fcm) and a last one for the path (.path)):

- cd arch to see examples of what this file looks like
  E.g. arch/arch-X64_JEANZAY.*

- Then you compile XIOS (cd .. to *yourxiosdirname*) referring to your set of arch files (example here for X64_JEANZAY):

./make_xios --arch *X64_JEANZAY* --full --prod --job 8

- Now cd .. back to your workdir

XIOS is now compiled

4.2 Compile NEMO

Now that XIOS is compiled, before running any simulation, you need to compile NEMO on your HPC.

- Clone NEMO to *yournemodirname*:

git clone --branch 4.2.0 https://forge.nemo-ocean.eu/nemo/nemo.git *yournemodirname*

cd *yournemodirname*

- You will now see the first level tree structure of NEMO where: arch contains the compilation settings which we discuss below, cfgs contains the reference configurations of which WED025 is one, doc has some useful documentation, ext takes you for example to the two-way nesting package of AGRIF, mk has the various compilation scripts, in src the actual .F90 model routines can be found, idealized test-cases are located in tests, and tools for pre and post processing are located in tools.

- Same as for XIOS, you need to have an arch file for your HPC. See NEMO user guide for instructions on setting up your arch file:
  https://sites.nemo-ocean.io/user-guide/install.html#download-and-install-the-nemo-code

  e.g. used here is that from arch/CNRS/X64_JEANZAY

- In this arch file you need to make sure that your xios path, refers to *yourxiosdirname.*
  You then refer to this arch file when using makenemo to compile NEMO:

./makenemo -m '*X64_JEANZAY*' -r WED025 -n 'WED025_demonstrator' -j 8

The WED025_demonstrator is now compiled. Next we need to compile the tools that help to create the domain_cfg.nc input file which gives NEMO information on your bathymetry, ice shelf draft, grid cell scale factors and more…

4.3 Compile domain_cfg tool

NEMO requires as input a file describing the domain (i.e. the horizontal and vertical mesh, scale factors, bathymetry and ice shelf draft). This file can easily be created using the DOMAINcfg tool. This tool requires an input file for bathymetry, ice shelf draft and coordinates for the horizontal grid and domain geometry. The exact details of the vertical grid are specified in the namelist.

Before building such a file, we need to compile the domain_cfg tool.To do so, follow the steps:

cd *yournemodirname/*tools

- Use the same archfile as the one you used to compile NEMO and do:

./maketools -m *X64_JEANZAY* -n DOMAINcfg

For more details on this tool, see this website for how to compile domain_cfg tools:
https://forge.nemo-ocean.eu/nemo/nemo/-/tree/main/tools/DOMAINcfg

The DOMAINcfg tool is now compiled

4.4 Build domain_cfg input file

Once the tool DOMAINcfg is compiled, we are ready to create the "domain_cfg.nc" file. To create this file from the provided information:

- Go back to your forcing directory, WED025_demonstrator_forcings, and create your own DOMAIN Folder, e.g. *mkdir DOMAIN_WED025*

- Then from the DOMAINcfg tools folder copy or link the following into your local DOMAIN_WED025 directory:

ln -s $yourWORKdir/yournemodirname/tools/DOMAINcfg/BLD/bin/make_domain_cfg.exe .

ln -s $yourWORKdir/yournemodirname/tools/DOMAINcfg/BLD/bin/dom_doc.exe .

cp $yourWORKdir/yournemodirname/tools/DOMAINcfg/namelist_ref .

- From your forcing folder, copy namelist_cfg_dom, bathy_meter_WED025.nc and coordinates_WED025.nc from your forcings folder into your DOMAIN_WED025 subfolder

- Then in your subfolder (DOMAIN_WED025) rename namelist_cfg_dom namelist_cfg (*cp namelist_cfg_dom namelist_cfg*)

Instead of providing a ready to use namelist, the namelist provided needs to be personalized to your configuration:

- Open namelist_cfg and change the experience name:

```
!-----------------------------------------------------------------------
&namrun        !   parameters of the run
!-----------------------------------------------------------------------
   cn_exp      =  "WED025"  !  experience name
```

- Rename to read in our files:

```
!-----------------------------------------------------------------------
&namdom        !   space and time domain (bathymetry, mesh, timestep)
!-----------------------------------------------------------------------
   cn_fcoord   =  'coordinates_WED025.nc'      ! external coordinates file (jphgr_msh = 0)
   cn_topo     =  'bathy_meter_WED025.nc'      ! external topo file (nn_bathy =1/2)
   cn_fisfd    =  'bathy_meter_WED025.nc'      ! external isf draft (nn_bathy =1 and ln_isfcav =
.true.)
```

- Change the dimensions to match the bathymetry file and change the name of the configuration:

```
!-----------------------------------------------------------------------
&namcfg        !   parameters of the configuration
!-----------------------------------------------------------------------

 cp_cfg      =  "WED"  !  name of the configuration

 jpidta      =     322   !  1st lateral dimension ( >= jpi )
 jpjdta      =     328   !  2nd    "         "    ( >= jpj )
 jpkdta      =      75   !  number of levels      ( >= jpk )
 Ni0glo      =     322   !  1st dimension of global domain --> i =jpidta
 Nj0glo      =     328   !  2nd    -                  -    --> j  =jpjdta
 jpkglo      =      75
```

- Also change the minimum thickness of the ice shelf draft and water column thickness

```
!-----------------------------------------------------------------------
&namzgr_isf    !   isf cavity geometry definition                     (default: OFF)
!-----------------------------------------------------------------------
   rn_isfdep_min    = 20.        ! minimum isf draft tickness (if lower, isf draft set to this value)
   rn_glhw_min      = 0.01        ! minimum water column thickness to define the grounding line
```

Finally, we can create the domain_cfg.nc file:

- Execute domain_cfg.exe:

./make_domain_cfg.exe

> *If you do not have enough computing resources, you can create and submit a script that executes *make_domain_cfg.exe* on 1 core : this submission script is very dependent on your HPC, so a general recommendation is to look at your cluster documentation or ask your friends or colleagues.

Domain_cfg.nc file is now created.

*Note: if you decide to adapt this demonstrator for your own domain and it is bigger and requires multiple cores to run, then you need to rebuild your domain_cfg as you end up with, for example, 4 domain_cfg_000x.nc files if you use 4 cores. To merge all these files in one, see and adapt section : Creating a single mesh_mask for plotting in this demonstrator.

*Suggestion: If you would like to add namelist_cfg to your domain_cfg.nc (in order to improve traceability of your domain_cfg.nc file), you can add the namelist as a variable inside the domain_cfg.nc file by simply running: ./dom_oce.exe

4.5 Run a WED025 simulation

Now you have XIOS and NEMO compiled and your domain_cfg.nc file has been created. All the other input files that are required are provided within the forcing tarball. So you are ready to run your first WED025 simulation!

- Go to the demonstrator directory of your regional configuration:

cd *yournemodirname*/cfgs/*WED025_demonstrator*

- Copy the experiment to a reference to keep it safe:

cp -r EXP00 EXPREF

- Enter in your simulation directory:

cd *$yourWORKdir*/yournemodirname/cfgs/WED025_demonstrator/EXP00

- Copy domain_cfg.nc here:

ln -s *$yourWORKdir/pathtoyourDOMAIN_WED025/*domain_cfg.nc .

- Link all the WED025 demonstrator forcings netcdfs to this experiment folder and copy in namelist_cfg so you replace the one you have by default:

ln -s *$yourWORKdir*/WED025_demonstrator_forcings/*nc .
rm namelist_cfg
cp *$yourWORKdir*/WED025_demonstrator_forcings/namelist_cfg ./

Now you have everything you need to run your regional configuration.

For this you need to build a script to run on HPC. We suggest you use 32 MPI.
Suggestion: look at the supercomputer documentation or copy from a friend

Before you launch the job, you need to change the last timestep of the simulation (nn_itend) to match the length of the simulation you want to do. In this case we will do a one month simulation because our forcing data only covers 1.5 months.

Example how to calculate your end timestep:

Timestep:
rn_Dt = 1200.

So we want 31 days x (24 x 60 x 60) = 2678400 seconds
Now divide by the timestep 2678400 / 1200 = 2232

- Edit namelist_cfg

nn_itend = 2232 ! last time step

We would also like to have monthly output. At the moment the output files will be for every 5 days. *For more info, see this useful tutorial of XIOS:
https://forge.ipsl.jussieu.fr/ioserver/raw-attachment/wiki/WikiStart/XIOS-tutorial.pdf
Or you can look at chapter 10, section 10.1 and 10.2 for of the NEMO book for more info on XIOS: https://zenodo.org/record/6334656#.YrLap-xBwW8

- To change the output frequency, open file_def_nemo-oce.xml and file_def_nemo-ice.xml which is where you define the output you want. Note that field_def is where all the output variables available are listed, then in file_def you define which variables you output at what frequency.

- Replace each mention of 5d with 1mo in file_def_nemo-oce.xml and file_def_nemo-ice.xml and at the end of the script, remove this line:

<file_group id="1m" output_freq="1mo" output_level="10" enabled=".TRUE."/> <!-- real monthly files →

Hint for replacement using vi: :%s/5d/1mo/gc

Now you can submit your job to run the simulation.

Once terminated, you now have run your first WED025 simulation.

4.6 Creating a single mesh_mask for plotting

When NEMO creates a mesh_mask file, it builds one file per subdomain. In order to use this file for post- or pre-processing (python script analysis for example), you need to gather all these files in one. This is the rebuilding step. For this you need to use rebuild_nemo.

See instructions here:
https://forge.nemo-ocean.eu/nemo/nemo/-/tree/main/tools/REBUILD_NEMO

- REBUILD_NEMO is compiled in the same manner as all the other tools. The compilation script is maketools and the options are very similar to the makenemo command where you have to refer to your HPC.
  e.g. used here is X64_JEANZAY

cd *$yourWORKdir*/yournemodirname/tools/

./maketools -m X64_JEANZAY -n REBUILD_NEMO

- Now return to your run directory and execute rebuild_nemo to combine all the mesh_masks as you will have run on multiple MPI.

cd *$yourWORKdir*/yournemodirname/cfgs/WED025_demonstrator/EXP00

- Command to rebuild your mesh mask:

yourWORKdir/yournemodirname/tools/REBUILD_NEMO/rebuild_nemo -m mesh_mask 32

*Note: In case you need to rebuild another output file, simply adapt the prefix file name (e.g. mesh_mask) and the number of domains (e.g. 32).

Now you should have a mesh_mask.nc file for your whole domain.

5. <u>Plotting in Python</u>

You have now completed the simulation, have your output files and are ready to do some analysis.

- You need to copy across the following files to your python space:

WED025_1mo_20000101_20001231_grid_T.nc
WED025_1mo_20000101_20001231_grid_U.nc
WED025_1mo_20000101_20001231_grid_V.nc
domain_cfg.nc
mesh_mask.nc

- Before running the plotting script to check your outputs, you need to be sure that *python* and the corresponding modules are installed. To do so, *conda* is your friend.

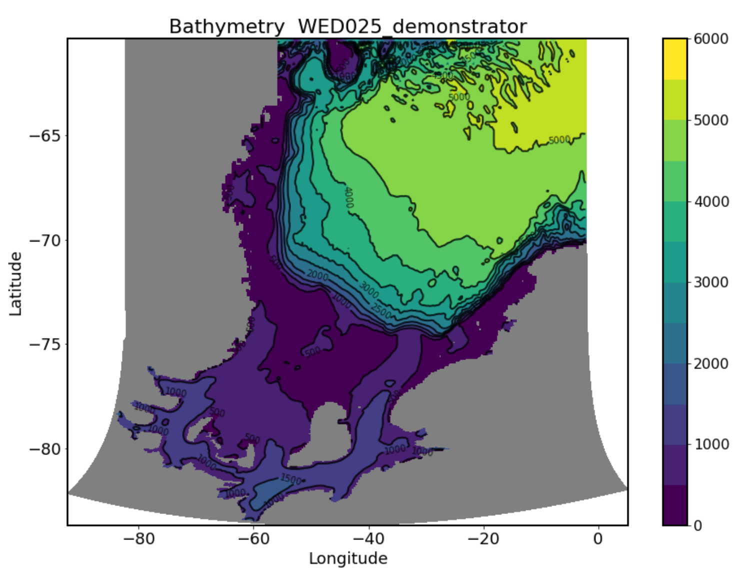
If conda is not installed in your system, you can install conda following the instructions on the conda website where you have instruction for linux, MacOS and Windows:
https://docs.conda.io/projects/conda/en/latest/user-guide/install/index.html#installation

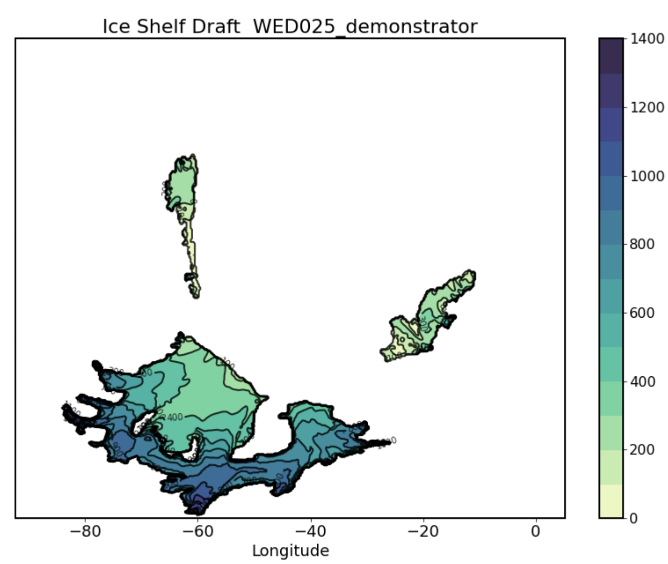
- Once conda is installed, you can create the WED025 environment. To do so, simply run:

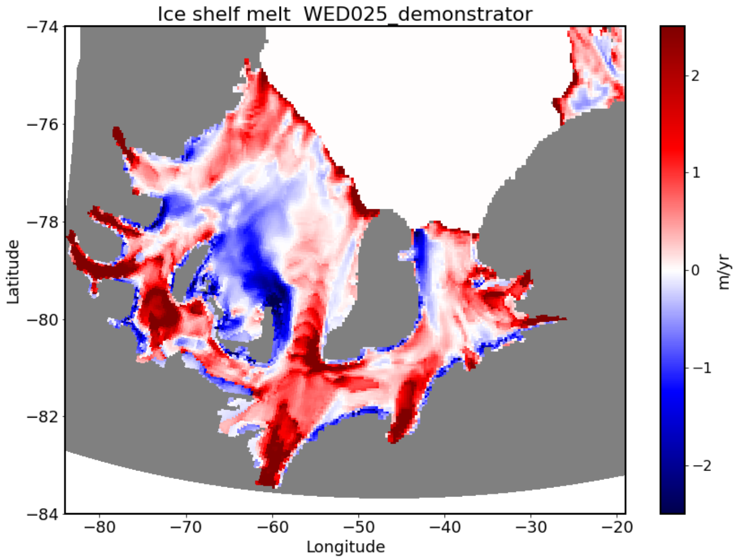
conda env create -f WED025.yml
then
Conda activate WED025

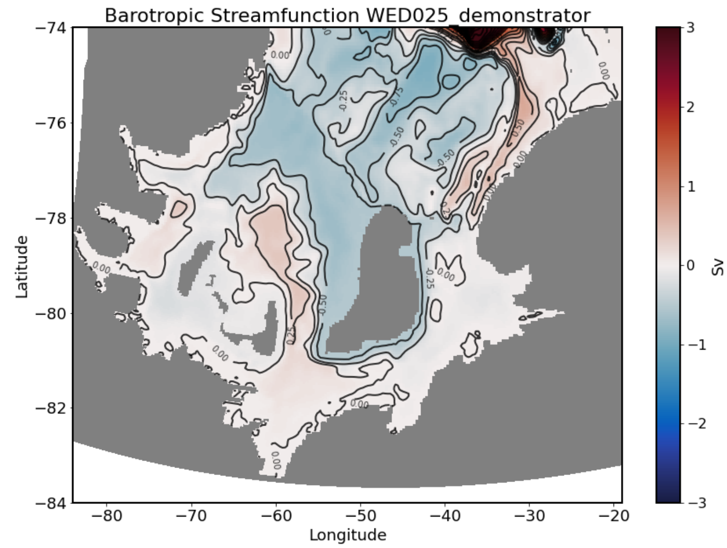
- Now you can run the plot.py script : *python plot.py*

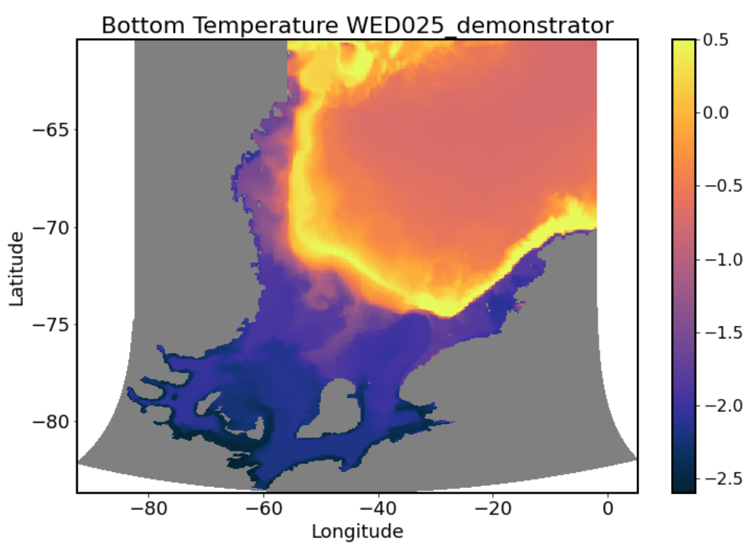
- Check that your results look like this:

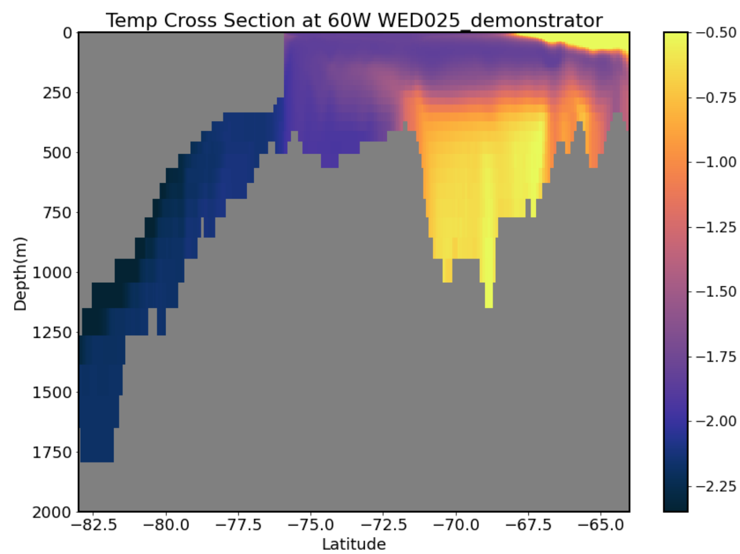





Ice shelf melt WED025_demonstrator
Latitude
Longitude
−74
−76
−78
−80
−82
−84
−80
−70
−60
−50
−40
−30
−20
m/yr
2
1
0
−1
−2


Barotropic Streamfunction WED025_demonstrator
Latitude
Longitude
−74
−76
−78
−80
−82
−84
−80
−70
−60
−50
−40
−30
−20
Sv
3
2
1
0
−1
−2
−3


Bottom Temperature WED025_demonstrator
Latitude
Longitude
−65
−70
−75
−80
−80
−60
−40
−20
0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5


Temp Cross Section at 60W WED025_demonstrator
Depth(m)
Latitude
0
250
500
750
1000
1250
1500
1750
2000
−82.5
−80.0
−77.5
−75.0
−72.5
−70.0
−67.5
−65.0
−0.50
−0.75
−1.00
−1.25
−1.50
−1.75
−2.00
−2.25

6. References

Bull et al., (2021) C., Vaňková, I., Holland, P. R., Mathiot, P., et al. (2021). Remote control of Filchner-Ronne ice shelf melt rates by the Antarctic slope current. Journal of Geophysical Research: Oceans, 126, e2020JC016550. https://doi.org/10.1029/2020JC016550

Jourdain, N. C., Molines, J. M., Le Sommer, J., Mathiot, P., Chanut, J., de Lavergne, C., & Madec, G. (2019). Simulating or prescribing the influence of tides on the Amundsen Sea ice shelves. Ocean Modelling, 133, 44-55. https://doi.org/10.1016/j.ocemod.2018.11.001

Mathiot, P., Jenkins, A., Harris, C., & Madec, G. (2017). Explicit representation and parametrised impacts of under ice shelf seas in the z∗ coordinate ocean model NEMO 3.6. Geoscientific Model Development, 10(7), 2849-2874.

Storkey, D., Blaker, A. T., Mathiot, P., Megann, A., Aksenov, Y., Blockley, E. W., Calvert, D., Graham, T., Hewitt, H. T., Hyder, P., Kuhlbrodt, T., Rae, J. G. L., and Sinha, B. (2018). UK Global Ocean GO6 and GO7: a traceable hierarchy of model resolutions, Geosci. Model Dev., 11, 3187–3213, https://doi.org/10.5194/gmd-11-3187-2018, 2018.