TUTORIAL 07:
CREATE MY CROCO INTERANNUAL CONFIG

In this tutorial, we do all the steps to run a CROCO interannual configuration. We will connect to the LENGAU cluster and run Matlab to create interannual CROCO input files using CROCO tools. We will then run the simulation on the CHPC cluster.

STEP 0: Forewords and Prerequisites

So far, we have run climatological simulations in which surface and lateral boundary forcings were monthly (12 steps) and repeated themselves every year (modulo 360 days). Our objective now is to run an interannual simulation. Therefore, we will need to prepare CROCO input files with actual interannual conditions. We will use:

- Surface atmospheric forcing from the ECWMF ERA5 reanalysis (https://www.ecmwf.int/en/forecasts/dataset/ecmwf-reanalysis-v5). After registration, hourly surface data are available globally from 1940 to the present at a resolution of 0.25°x0.25° (~25km)
- Open Boundary forcing from the GLORYS12 (MERCATOR) reanalysis (https://data.marine.copernicus.eu/product/GLOBAL_MULTIYEAR_PHY_001_030). Monthly data can be accessed at a 1/12°x1/12° (~9km) resolution with 50 vertical levels from 1993 onward.

➥ Due to the size of the data we will need to access, we will run the interannual simulation over a **sample period of 3 (only) months**, from January 2013 to March 2013.

➥ To limit the network access, ERA5 and GLORYS12 data have already been downloaded over a limited domain extending from 170°W to 170°E and from 60°S to 60°N. So, when creating your grid:

Do not exceed a 100x100 grid
Do not overlap the equatorial zone by less than 2°
Do not create boundaries of less than 5 points
The domain is limited to [170°W-170°E; 60°S-60°N]

STEP 1: Logging onto the Lengau HPC cluster and create a new CROCO working directory

→ From a terminal/konsole, execute the following instruction:

```
ssh -X login@lengau.chpc.ac.za
```

Replace **login** with your corresponding account number.

→ Reserve one interactive processor to do the pre-processing steps (**STEP 4** from **#TUTORIAL01**):

```
[login@login2 ~]$ qsubi1
[login@cnode0220 ~]$
```

→ Go into your **croco-v2.0.1** directory (**lustre/CROCO/croco-v2.0.1**):

```
[login@cnode0220 ~]$ cd lustre/CROCO/croco-v2.0.1
[login@cnode0220 croco-v2.0.1]$
```

→ Create a new CROCO configuration called **Run_Inter**. For this, you have to repeat the **STEP 2** from **#TUTORIAL02**, i.e. edit **create_config.bash** and execute it:

```
[login@cnode0220 croco-v2.0.1]$ nedit create_config.bash &
[login@cnode0220 croco-v2.0.1]$ ./create_config.bash
[login@cnode0220 croco-v2.0.1]$ cd Run_Inter
```

STEP 2: Creating CROCO input files for Run_Inter

→ It is done with **MATLAB**

→ Launch **MATLAB** with the command **matlab -nodesktop** (or the alias **mat**):

```
[login@cnode0220 Run_Inter]$ matlab -nodesktop
[login@cnode0220 Run_Inter]$
```

NODES

→ Execute the command start and open **crocotools_param.m** using the Matlab command edit:

```
>> start
>> edit crocotools_param
```

➥ Activate the creation of graphics after each pre-processing step by changing **makeplot=0** to **makeplot=1** at line 130.

GRID & MASK

→ Create a new CROCO grid (see prerequisites in **#STEP 0**) using the Matlab command make_grid:

```
>> make_grid
>>
```

➥ Parameters for this script can be found in section 1 of **crocotools_param.m** ("1- Configuration parameters") at line 43.

➥ This script will create your horizontal/vertical grid (position of the grid point, size of the grid cells, bottom topography, land mask, etc…) using information from the global etopo2 data base. The grid will be stored in the NetCDF file **CROCO_FILES/croco_grd.nc**

→ Remember the size of your grid (LLm, MMm, and N)

INITIAL & BOUNDARY CONDITIONS

→ Create your CROCO initial (ini) and interannual boundary conditions (bry).

For this, we will use the **make_OGCM_mercator** Matlab script stored in the **croco_tools-v2.0.0/Oforc_OGCM** directory. This script uses parameters from the **crocotools_param.m** file.

❶ Adjust the boundary switches according to your new grid. See section 4 ("4- Open boundaries and initial conditions parameters"):

```
228      %  Open boundaries switches (! should be consistent with cppdefs.h !)
229      %
230 -    obc = [1 1 1 1]; % open boundaries (1=open , [S E N W])
```

❷ Create only initial and boundary files (*bry.nc), as it is cheaper that creating climatological (*clm.nc) files. Modify section 4, so that:

```
240 -    makeini    = 1;   % initial data
241 -    makeclim   = 0;   % climatological data (for boundaries and nudging layers)
242 -    makebry    = 1;   % lateral boundary data
```

❸ Adjust time parameters in section 6 ("6- Reference date and simulation times"), so that:

```
332 -    Ymin           = 2013;          % first forcing year
333 -    Ymax           = 2013;          % last  forcing year
```

❹ Adjust GLORYS12 data path in section 7 ("7- Parameters for interannual forcing"):

```
352 -    Download_data = 0;   % Get data from OPENDAP sites
423 -    OGCM          = 'mercator';        % Select OGCM: SODA, ECCO, mercator
425 -    OGCM_dir      = '/home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_DATA/MERCATOR_GLOB/';
```

➤➤➤ Execute the Matlab script **make_OGCM_mercator**:

```
>> make_OGCM_mercator
>>
```

SURFACE FORCING

→ The model surface forcing (momentum, heat, and freshwater) will be calculated by the model using bulk formulae using ECMWF ERA5 reanalysis. ERA5 (u10, v10, T2m, Q2m, Qsolar, Qlw_in, and Precip) files have been downloaded on the Lengau cluster using the python script in **croco_tools-v2.0.0/Aforc_ERA5**. The monthly files are stored in the directory **/home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_DATA/ERA5_GLOB**. You can list them:

```
[login@cnode0220]$ ls /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_DATA/ERA5_GLOB
[login@cnode0220]$
```

➥ We will not be using the **make_ERA5** Matlab scripts to interpolate ERA5 data onto your model grid and avoid generating large files (*blk.nc). Instead, global ERA5 fields will be interpolated onto your grid directly by the CROCO model. This is called ONLINE forcing. Steps will be detailed in the following sections (activate **#define ONLINE** in **cppdefs.h** and adjust ERA5 data path in **croco_inter.in**).

→ You are finished with **STEP 2**. You can exit Matlab

```
>> exit
>>
```

→ You can inspect all your input files stored in CROCO_FILES using the **ncdump** or **ncview** tools:

```
[login@cnode0220 Run_Inter]$ cd CROCO_FILES; ls
[login@cnode0220 CROCO_FILES]$ ncdump -h croco_ini_mercator_Y2013M1.nc
[login@cnode0220 CROCO_FILES]$ ncview croco_bry_mercator_Y2013M3.nc
[login@cnode0220 CROCO_FILES]$ cd ..
```

STEP 3: Compiling CROCO model

→ To compile CROCO on Lengau, you need to copy my **jobcomp_lengau** into your **Run_Inter** directory (same as in **#TUTORIAL04**):

```
cp /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/jobcomp_lengau .
```

→ Edit the CROCO parameter file **param.h** using **vi** or **nedit**:

```
[login@cnode0220 Run_Inter]$ nedit param.h &
[login@cnode0220 Run_Inter]$
```

➥ Create a new config (below line 199) by adding a 2-line **elif** statement, with a new config name and adjusted values for LLm0, MMm0, and N.
You can get these values by checking the size of your grid with **ncdump -h CROCO_FILES/croco_grd.nc** *and removing two ghost points from* **xi_rho** *and* **eta_rho**

➥ Check the parameters for the **parallelisation** (NP_XI=1, NP_ETA=4 line 251)

→ Edit the **cppdefs.h** file using **vi** or **nedit**:

```
[login@cnode0220 Run_Inter]$ nedit cppdefs.h &
[login@cnode0220 Run_Inter]$
```

➥ At line 71, activate your config (ex: **#define BENGUELA_LR**).

➥ Activate MPI parallelization at line 74 (**# define MPI**).

➥ Define which of your boundaries are open or closed (lines 95-98).

➥ Activate bry forcing with **# undef CLIMATOLOGY** (line 198) and **# define FRC_BRY** (line 212).

➥ Activate the CPPKEYS for bulk surface forcing, and ERA_ECMWF ONLINE interpolation: **#define BULK_FLUX** (line 170), **#define ONLINE** (line 180), and **#define ERA_ECMWF** (line 183)

→ Compile CROCO using the **jobcomp_lengau** script and verify that the executable **croco** has been created:

```
[login@cnode0220 Run_Inter]$ ./jobcomp_lengau
[login@cnode0220 Run_Inter]$ ls
```

STEP 4: Running CROCO

→ To launch your simulation on the Lengau cluster using the PBS Pro job scheduler, you need to copy my **run_croco_inter.pbs** file into your **Run_Inter** directory:

```
cp /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/run_croco_inter.pbs .
```

→ Edit the script **run_croco_inter.pbs** using **vi** or **nedit**:

```
[login@cnode0220 Run_Inter]$ nedit run_croco_inter.pbs &
[login@cnode0220 Run_Inter]$
```

➥ Check the PBS scheduler parameters (mpiprocs consistent with **param.h**) and email.

➥ Check the path of your **Run_Inter** working directory.

➥ Check the configurable section:

- Adjust NBPROCS (line 37), consistently with **param.h** and mpiprocs
- Put BULK_FILES=0 (line 56)
- Fix the name of the OGCM (OGCM=mercator, line 67)
- Choose the time step wisely (DT, line=72)
- Modify the period of the simulation (NY_START=2013, NY_END=2013, line 82-83)
- Choose no spin-up (NY_SPIN=0, line 94) and no restart (RSTFLAG=0, line 106).

→ Edit the CROCO parameter file **croco_inter.in** using **vi** or **nedit**:

```
[login@cnode0220 Run_Inter]$ nedit croco_inter.in &
[login@cnode0220 Run_Inter]$
```

➥ Fix your grid parameters at line 8: THETA_S, THETA_B, Hc (m)

➥ Check the outputs requested in sections:

- primary_history_fields and primary_history_fields for instantaneaous outputs
- primary_averages and primary_history_fields for averages

➥ Modify the path of the global ERA5 data at the end of the file, so that:

```
online:    byear  bmonth recordsperday byearend bmonthend / data path
           NYONLINE NMONLINE 24 2013 4
           /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_DATA/ERA5_GLOB/
```

→ Launch your first simulation using the PBS command **qsub**:

```
[login@cnode0220 Run_Inter]$ qsub run_croco_inter.pbs
[login@cnode0220 Run_Inter]$
```

➥ Your simulation will soon start. You will receive an email.

➥ **PBS** output/error files are in **Run_Inter/run_croco_inter.pbs.(o/e)xxxxx**

➥ Logs (***.out**) and outputs (**avg/hist .nc**) will be stored in the directory: **./SCRATCH**

STEP 5: Visualising model outputs

→ This can be done with **MATLAB**

→ Launch **matlab -nodesktop** (or the alias **mat**) to visualize your outputs using croco_gui:

```
>> start
>> croco_gui
```

→ You can concatenate model outputs to visualize them easily in croco_gui:

```
[login@cnode0220 Run_Inter]$ cd SCRATCH
[login@cnode0220 SCRATCH]$ ncrcat croco_avg_Y*.nc croco_avg_Y2013.nc
```

STEP 6: Exiting

→ When you are done, exit Matlab and logout from the super-computer:

```
>> exit
[login@cnode0220 Run_Inter]$ exit
[login@login2 ~]$ exit
```

STEP 7: Check List

→ Here is the list of the essential commands that you must execute during this hands-on session. The following table can help you confirm that you have executed all of them:

		Commands	✌
STEP 1	1	ssh -X login@lengau.chpc.ac.za	
	2	qsubi1	
	3	cd lustre/CROCO/croco-v2.0.1	
	4	nedit create_config.bash &	
	5	./create_config.bash	
	6	cd Run_Inter	
STEP 2	1	matlab -nodesktop	
	2	start	
	3	edit crocotools_param	
	4	make_grid	
	5	make_OGCM_mercator	
	4	exit	
STEP 3	1	cp /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/jobcomp_lengau .	
	2	nedit param.h &	
	3	nedit cppdef.h &	
	4	./jobcomp_lengau	
STEP 5	1	cp /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/run_croco_inter.pbs	
	2	nedit run_croco_inter.pbs &	
	3	nedit croco_inter.in &	
	4	qsub run_croco_inter.pbs	
STEP 6	1	matlab -nodesktop	
	2	start	
	3	croco_gui	
STEP 7	1	exit	
	2	exit	
	3	exit	

STEPS to design a CROCO Interannual configuration

⓪ Working Directory

in `~/croco-v2.0.1/` edit and execute `./create_config.bash`

❶ Model Inputs

- **Where?**

in `~/croco-v2.0.1/Run_Inter`

- **Which file(s) and param(s)?**

`crocotools_param.m`
- Grid param (horz/vert+topo),
- Time param (period/spin-up),
- ***bulk*** and ***bry*** choice…

- **Which command(s)?**

`./matlab -nodesktop`

make_grid;

make_OGCM; make_ERA5;

make_QSCAT_daily.nc

- **Expected output(s)?**

`LLM, MMm, N`

`Run_Inter/CROCO_FILES`

croco_grd.nc croco_ini.nc

croco_blk_Y----M--.nc

croco_bry_Y----M--.nc

❷ Compile

- **Where?**

in `~/croco-v2.0.1/Run_Inter`

- **Which file(s) and param(s)?**

`param.h`
- Grid param `LLM0, MMm0, N`
- MPI/OPENMP (nb procs)

`cppdefs.h`
- config, parallelization
- frc/blk choices + ONLINE
- clim/bry boundary forcing
- nesting, MPI_NOLAND, NC4PAR
- physical schemes, etc…

- **Which command(s)?**

`./jobcomp` (`./jobcom_lengau`)

- **Expected output(s)?**

⇨ Model executable `croco`

❸ Run

- **Where?**

in `~/croco-v2.0.1/Run_Inter`

- **Which file(s) and param(s)?**

`Run_Clim/CROCO_FILES`

`croco_inter.in`
- Vertical grid parameters (streching)
- Variables to be saved
- ONLINE bulk

`run_croco_inter.bash`
- input/output paths, nprocs
- DT, NAVG, simulation period
- restart (yes/no)
- ***frc*** + ***bry*** choice

- **Which command(s)?**

`./run_croco_inter.bash`

(`qsub run_croco_inter.pbs`)

- **Expected output(s)?**

⇨ Model outputs

croco_avg_Y----M--.nc,

croco_his_M----M--.nc

in `Run_Inter/SCRATCH`