

Tutorial CROCO Avanzado: Simulaciones Interanuales

1. Propósito

En este tutorial revisaremos las instrucciones para realizar una simulación interanual, es decir, correspondiente a una fecha, del pasado, específica.

2. Creando el directorio de trabajo INTERANUAL

Para esta sección, revise las instrucciones del primer tutorial del curso. Edite el archivo `create_run.bash` con las instrucciones

```
1 cd croco
2 nano create_run.bash
```

Modificar esta sección para crear una carpeta de trabajo llamada INTERANUAL

```
1 #-----
2 # BEGIN USER MODIFICATIONS
3 #
4 # Get CROCO directory
5 CROCO_DIR="/home/courses/student11/croco"
6 #
7 SOURCES_DIR="/home/courses/student11/croco"
8 #
9 TOOLS_DIR="/home/courses/student11/croco/croco_tools"
10 #
11 MY_CONFIG_PATH=${SOURCES_DIR}
12 #
13 # Name of the configuration directory defined by the user
14 #
15 MY_CONFIG_NAME='INTERANUAL'
16 #
17 #
18 # END USER MODIFICATIONS
19 #-----
```

y después ejecute la instrucción

```
1 ./create_run.bash
```

3. Compilando CROCO

Primero vamos a compilar el ejecutable de CROCO usamos las siguientes instrucciones. Modificamos el `cppdefs.h` para activar la opción de paralelización **MPI** y las opciones

```
1 cd INTERANUAL
2 nano cppdef.h
```

donde cambiamos las opciones a

```
# define MPI
# define BULK_FLUX
# undef CLIMATOLOGY
# define FRC_BRY
```

Ahora compilamos el código

```
1 ml purge
2 ml intel/2019b
3 ml netCDF-Fortran/4.4.4
4 ./jobcomp
```

4. Creando los archivos de entrada

Para crear los archivos de entrada que leerá el ejecutable **croco** usaremos una serie de funciones llamadas en su conjunto **CROCO_TOOLS**. Estas funciones fueron escritas en Matlab (Penven et al., 2008) y han sido adaptadas para funcionar en Octave, usando el paquete OCTCDF.

4.1. El código de CROCO_TOOLS

Recuerde editar el archivo **start.m**

```
1 nano start.m
```

y modificar la siguiente línea

```
1 tools_path='/home/courses/student11/croco/croco_tools/';
```

4.2. Modificaciones al crocotools_param.m

```
1 %
2 CROCOTOOLS_dir = '/home/courses/instructor05/croco/croco_tools/';
3 %
```

```
1 %
2 % MERCATOR case:
3 % =====
4 % To download data: set login/password (http://marine.copernicus.eu)
5 % and path to croco's motuclient python package;
6 % or set pathMotu='' (empty) to use your own motuclient
7 %
8 % Various sets of data are proposed in the
9 % Copernicus web site (Mercator, UK Met Office ...)
10 %
11 if strcmp(OGCM,'mercator')
12     user      = 'gcambon';
13     password = 'J0pozono';
14     =====
```

```
1 %
2 CROCOTOOLS_dir = '/home/courses/instructor05/croco/croco_tools/';
3 %
```

4.3. Archivos auxiliares - ECMWF

Para bajar archivos desde el ECMWF necesita abrir una cuenta y obtener una clave, que será guardada en la raíz de su cuenta con el nombre **.ecmwfapirc**

En este curso usaremos la siguiente clave (que será desactivada mañana)

```

1 copy .ecmwfapirc in ~/
2 {
3   "url": "https://api.ecmwf.int/v1",
4   "email": "gildas.cambron@ird.fr",
5   "key": "76e820b695ccb3b108eb845ecfbb49d"
6 }
```

Asegúrese que esté usando **python2.7**

```

1 pip install --user ecmwf-api-client
2 mkdir ~/croco/INTERANUAL/DATA/ERA1
```

pues la api no funciona aún con python3

El archivo **ERA-Interim_request.py** será usado para obtener las salidas de ERA-Interim. Este archivo está ubicado en **croco_tools/Aforc_ECMF**, y debe modificar

```

1 #####
2 # Parameters
3 #####
4 # Period
5 yearStart = 2005
6 yearEnd   = 2005
7 monthStart = 1
8 monthEnd   = 1
9
10 # Area of interest
```

para obtener el forzamiento atmosférico del mes de Enero del 2005.

4.4. Estructura de Archivos

Es importante entender que las salidas originales de los modelos que va a usar, atmósfera u océano serán bajados al directorio

```

1 INTERANUAL/DATA
```

mientras que la información, ya interpolada a su dominio estará en

```

1 INTERANUAL/CROCO_FILES
```

4.5. Usando Matlab

Para crear los archivos de entrada usando Matlab, normalmente primero tenemos que cargar el programa usando

```

1 ml purge
2 ml Matlab/2017
3
4 LD_PRELOAD=/home/lmod/software/Core/ifort/2019.2.187
5 -GCC-8.2.0-2.31.1/
6 compilers_and_libraries_2019.6.324/linux/compiler/lib/intel64/libirc.so
7
8 matlab -nodesktop -nosplash

```

y luego **make_grid**, etc. Sin embargo como este proceso corre en el nodo maestro y demora mucho, será automáticamente terminado antes de que termine de procesar los archivos. Para evitar problemas, mandamos el proceso de obtención de archivos e interpolación a un nodo de cálculo con el script

```

#!/bin/bash
#SBATCH --job-name=mk_config
#SBATCH --mail-user=gildas.cambron@ird.fr
#SBATCH --mail-type=ALL
#SBATCH --output=%j_%x.out
#SBATCH --error=%j_%x.err
#SBATCH --partition=slims
#SBATCH -n 1
#SBATCH --ntasks-per-node=1
#SBATCH --mem-per-cpu=10000
#SBATCH --license=matlab #agregar si se va a ocupar MATLAB

```

```

module purge
ml Matlab/2017
LD_PRELOAD=/home/lmod/software/Core/ifort/2019.2.187
-GCC-8.2.0-2.31.1/compilers_and_libraries_2019.6.324/
linux/compiler/lib/intel64/libirc.so
matlab -nodisplay -nosplash -nodesktop < mymatlab.m > matlab.out

```

Llame este script

```
1 batch_croco_tools.bash.slurm
```

Lo puede obtener de

```
1 http://mosa.dgeo.udec.cl/CROCO2021/Avanzado/Tutorial_Interanual/scripts/
```

4.6. CB Oceánica - Mercator (GLORYS 1/4)

Para obtener la condición de borde oceánica **mymatlab.m** debe contener

```

1 start
2 make_grid
3 make_OGCM_mercator

```

Asegúrese que en **crocotools_param.m** la siguiente línea diga

```
1 Download_data = 1
```

Luego envíe la solicitud de bajar datos mediante

```
1 sbatch batch_croco_tools.bash.slurm
```

Cuando termine, deberá encontrar en **CROCO_FILES** los archivos

```
1 croco_grd.nc
2 croco_bry_mercator_Y2005M1.nc
3 croco_clm_mercator_Y2005M1.nc
4 croco_ini_mercator_Y2005M1.nc
```

4.7. ERAI - ECMWF

La obtención de los archivos de ERAI depende del archivo de clave **.ecmwfapirc** descrita anteriormente.

En este caso debemos bajar los archivos con el script python **ERA-Interim_request.py** que modificó anteriormente haciendo

```
1 python ERA-Interim_request.py
```

desde el directorio **Aforc_ECMWF**.

Después asegúrese que en **crocotools_param.m** la siguientes líneas digan

```
1 Download_data = 0
2 Reformat_ECMWF = 1
```

pues no queremos bajar los datos, ya lo hicimos con el script python, pero si queremos que transforme esos archivos al formato NetCDF que lee CROCO.

Ahora modifique el **mymatlab.m**

```
1 start
2 make_ECMWF
```

y lance el script

```
1 sbatch batch_croco_tools.bash.slurm
```

Cuando termine, deberá encontrar en **CROCO_FILES** el archivo

```
1 croco_blk_ECMWF_Y2005M1.nc
```

4.8. Archivos Iniciales - Repositorio

Si tuvo problemas con esta etapa, copie esos archivos en el directorio **CROCO_FILES** para avanzar a la siguiente sección desde el directorio

```
1 http://mosa.dgeo.udec.cl/CROCO2021/Avanzado/Tutorial_Interanual/CROCO_FILES
```

5. Lanzando la simulación

Hay que modificar el archivo **croco.in** para darle el nombre de los archivos que vamos a leer

```
1 bulk_forcing: filename
2   CROCO_FILES/croco_blk_ECMWF_Y2005M1.nc
3 bounday_forcing: filename
4   CROCO_FILES/croco_bry_mercator_Y2005M1.nc
5 initial: NRREC / filename
6   1
7   CROCO_FILES/croco_ini_mercator_Y2005M1.nc
```

Para lanzar la simulación copie el archivo **run_nlhpc.bash** que usó anteriormente. También puede encontrar un ejemplo de este archivo en el siguiente enlace

```
1 http://mosa.dgeo.udec.cl/CROCO2021/Tutorial01/run_nlhpc.bash
```

Lance la simulación con la instrucción

```
1 sbatch run_nlhpc.bash
```

Cuando termine, deberá encontrar en **CROCO_FILES** los archivos

```
1 croco_his.nc
2 croco_rst.nc
3 croco_ini.nc
```

6. Visualización de resultados

6.1. ncview

Use el programa **ncview** para mirar el contenido de las salidas interanuales

```
1 ml purge
2 ml icc/2019.2.187-GCC-8.2.0-2.31.1 impi/2019.2.185 ncview/2.1.7
3 ncview CROCO_FILES/croco_avg.nc
```

7. Conclusión

En este tutorial aprendió lanzar una simulación interanual forzando su modelo con GLORYS y ERA-Interim

Para más información:

Gildas Cambon (gildas.cambon@ird.fr)

Contribuciones de:

Andrés Sepúlveda

8. Enlaces útiles

8.1. ECMWF

<https://www.ecmwf.int/en/forecasts/access-forecasts/ecmwf-web-api>

8.2. Mercator

<https://marine.copernicus.eu/user-corner/technical-faq/what-are-motu-client-motuclient-and-python-requirements>

8.3. Otras herramientas

<https://github.com/trondkr/model2roms>