Tutorial 07 LiveCROCO: Rivers

1. Purpose

The objective of this tutorial is to explain implementation of rivers in a CROCO simulation. In this document we will review cases of constant flow and variable flow rivers.

2. Case 1a: Constant flow rivers

In this exercise we will simulate low resolution Benguela case (BENGUELA_LR; case that has been analyzed in previous tutorials), where we will include two constant flow rivers, and define temperature and salinity that advects the flow. For this example, the location of these rivers is proposed to be 34.65 °S and 19.83 °E, and 28.7°S and 16.16 °E (Figure 1). Next we describe the steps to follow.

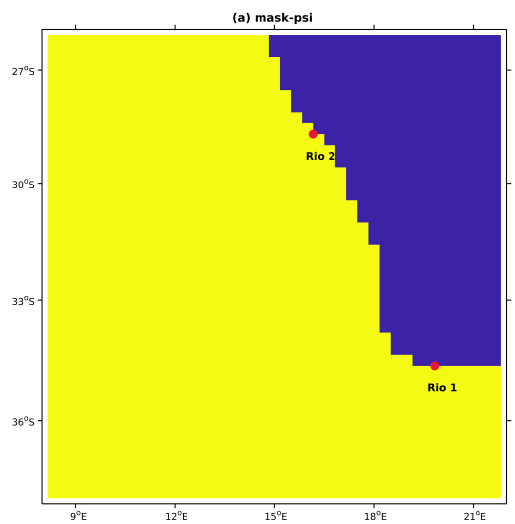

Figura 1: Position of rivers in Case 1a.

2.1. Create working directory

We will create the directory where we will work in this exercise

```
cd croco
nano create_run.bash
```

And we edit the name of directory:

```
# Name of configuration directory defined by user
#
MY_CONFIG_NAME='BENGUELA_R1A'
```

and we go to the folder

```
cd croco/BENGUELA_R1A
```

2.2. Creating input files

We continue with generation of input files. These files are the same files created in BENGUELA_LR case. As a reminder, the steps to follow in Octave terminal:

```
oct_start
make_grid
make_forcing
make_clim
```

(in make_grid we do not make modifications to the grid). When finished, in CROCO_FILES directory you will find croco_grd.nc, croco_frc.nc, croco_clm.nc and croco_ini.nc files. If you previously have these files, you can copy them to CROCO_FILES directory found in current directory

```
cp ../BENGUELA_LR/croco_grd.nc ./CROCO_FILES
cp ../BENGUELA_LR/croco_frc.nc ./CROCO_FILES
cp ../BENGUELA_LR/croco_clm.nc ./CROCO_FILES
cp ../BENGUELA_LR/croco_ini.nc ./CROCO_FILES
```

These files will also be available at:

```
http://mosa.dgeo.udec.cl/CROCO2021/Tutorial01/ArchivosIniciales
```

2.3. Modifying physical conditions: cppdefs.h

To include a constant flow river (or rivers) it is necessary to modify cppdefs.h:

```
nano cppdefs.h
```

In this case the only change we will make is in “Point Sources - Rivers“ section (line 240) where we will define PSOURCE option

```
                      /* Point Sources - Rivers */
# define PSOURCE
# undef PSOURCE_NCFILE
# ifdef PSOURCE_NCFILE
#  undef PSOURCE_NCFILE_TS
# endif
```

We save the changes and compile

```
./jobcomp
```

2.4. River properties

Next we will configure variables: position, direction, sense, flow, and tracer values of the rivers. These properties must be entered in croco.in file, which we will modify

```
nano croco.in
```

Near the end of this file (between lines 256 and 259), there is a special section for **psource** where it will be shown to us (not to be confused with the **psource_ncfile** section below).

```

psource:   Nsrc  Isrc  Jsrc  Dsrc  Qbar [m3/s]    Lsrc        Tsrc
             2
                   3    54     1    200.            T T      20. 15.
                   3    40     0    200.            T T      20. 15.
```

where Nsr is the number of rivers to include, Isrc and Jsrc are the position of the river in xi and eta directions respectively (or in this case, zonal and meridional), Dsrc is the orientation of river where 0 is zonal and 1 is meridional, Qbar is the river flow where positive indicates a south/north (or west/east) direction and negative a north/south (east/west) direction, Lsrc indicates whether the tracers advected by the river are estimated analytically (”F’ ’) or are supplied by the user (“T”), and Tsrc the values of the tracers. In this case, in Lsrc and Tsrc the first and second columns correspond to the temperature and salinity tracers, respectively.

As we mentioned at the beginning of the exercise, we will include two rivers ($Nsrc = 2$) with a flow of 20000 m3/s, where the mouth of the first (R1) and second (R2) rivers are located at 34.65 °S and 19.83 °E, and 28.7°S and 16.16 °E, respectively (Figure 1). From the location of the rivers we suggest that R1 flows from north to south, therefore $Dscr = 1$ and $Qbar = -20000$. In the case of R2 it will flow from east to west $Dscr = 0$ and $Qbar = -20000$. Since we want to define the tracers in the advection of the flows, we keep Lsrc as it originally appears. Tsr displays the temperature and salinity values in its first and second columns. Both rivers will have the same temperature and salinity, that is: 20 (°C) and 2 (psu). A slightly more complex aspect to define is the position of the rivers (Isrc,Jsrc). The detailed explanation of how to choose the position is explained in Annex A. In this part we will limit ourselves to mentioning that the position of Rio 1 (Rio 2) is Iscr=36 and Jsrc=13 (Iscr=24 and Jsrc=35) . Therefore replacing with the values mentioned above in croco.in:

```

psource:   Nsrc  Isrc  Jsrc  Dsrc  Qbar [m3/s]    Lsrc        Tsrc
             2
                  36    13     1     -20000.           T T      20. 2.
                  24    35     0     -20000.           T T      20. 2.
```

We save changes and then we send the task to the server

2.5. Running simulation

With the modifications made, we run the simulation

```
./croco croco.in
```

At the end of simulation we will have a his/avg and rst file in CROCO_FILES. If there was an error or problem, you can find the AVG file at:

```
http://mosa.dgeo.udec.cl/CROCO2021/Tutorial08/BENGUELA_R1A/CROCO_FILES/
```

The impact of the rivers will be observed through salinity. Using ncview

```
ncview CROCO_FILES/croco_avg.nc
```

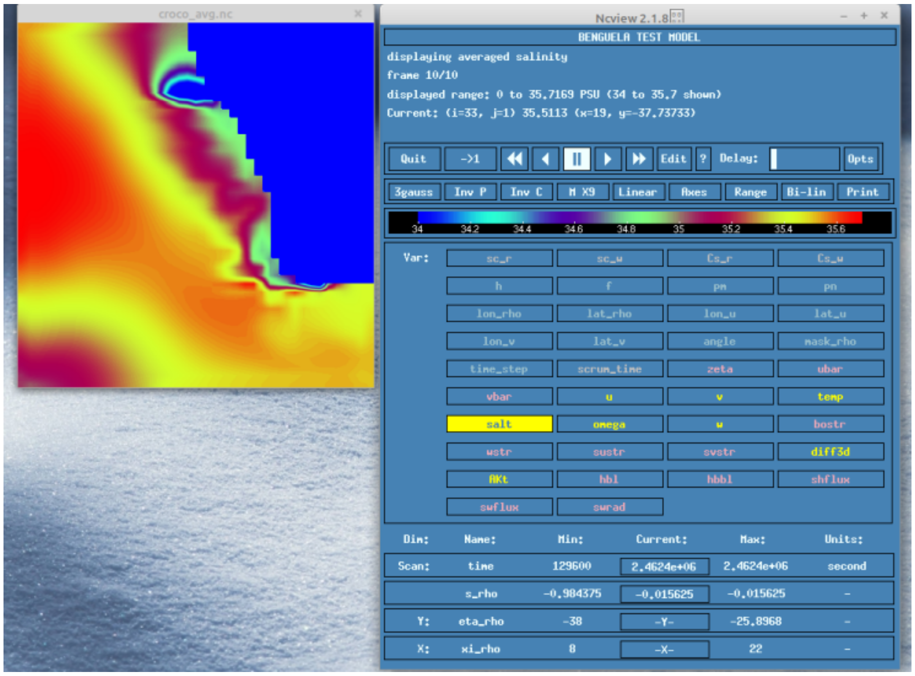

Figura 2: Case 1a: Surface salinity at t=10 obtained from croco_avg.nc (range 34-35.7 psu)

We select the most superficial depth and in the range option (located at the top of the panel) we modify the limits of the graph between 34 and 35.7 psu. In last record we clearly observe the impact of two rivers (Figure 2). However, it must be remembered that this situation is not realistic since the flow of these rivers was overestimated.

3. Case 1b: Constant flow rivers with changing vertical flow distribution

Up to this point we haven't mentioned about the configuration of the vertical flux distribution (*Qshape*). In this exercise we will run the same previous exercise, but modifying Qshape. For this, we will work in the same directory of the previous section and we will carry out the following steps

3.1. Create the working directory

We create the directory where we will work in this exercise

```
cd croco
nano create_run.bash
```

And we edit the name of the directory:

```
# Name of the configuration directory defined by the user
#
MY_CONFIG_NAME='BENGUELA_R1B'
```

and we go to the folder

```
cd croco/BENGUELA_R1B
```

3.2. Create input files

The same files used in the previous exercise will be used (see Section 2.2).

```
cp ../BENGUELA_R1A/croco_grd.nc ./CROCO_FILES
cp ../BENGUELA_R1A/croco_frc.nc ./CROCO_FILES
cp ../BENGUELA_R1A/croco_clm.nc ./CROCO_FILES
cp ../BENGUELA_R1A/croco_ini.nc ./CROCO_FILES
```

3.3. Modifying Qshape and Compile

For this, it is necessary to modify the analytical.F code found in CROCO code directory (i.e., croco/OCEAN). Since this file is part of the original code, before modifying, it is necessary to back up:

```
cp ../OCEAN/analytical.F ../OCEAN/analytical_respaldo.F
```

Now, we are in position to modify analytical.F

```
nano ../OCEAN/analytical.F
```

This file is a set of subroutines in which various analytical fields are provided to the model when required, such as the vertical flow distribution. We look for ana_psource (line 1759) subroutine , where the analytical part of the rivers is defined. This routine begins as follows:

```
!======================================================================
!                    subroutine ana\_psource
!======================================================================
!
#if defined PSOURCE && defined ANA_PSOURCE
!
!----------------------------------------------------------------------
!  Set analytical tracer and mass point sources and sinks
!----------------------------------------------------------------------
!
      subroutine ana_psource_tile (Istr,Iend,Jstr,Jend)
      implicit none
```

We look where the configuration for regional case is shown (line 1883) and we will find:

```
1  # elif defined REGIONAL
2  #  define CST_SHAPE
3  #  ifdef CST_SHAPE
4          cff=1./float(N)
5          do k=1,N                        ! Uniform vertical
6            do is=1,Nsrc                  ! distribution
7              Qshape(is,k)=cff
8            enddo
9          enddo
10 #  elif defined EXP_SHAPE
11         do is=1,Nsrc
```

The default option is CST_SHAPE. (# define CST_SHAPE; line 2 in top box). This option distributes the flow in the vertical in a uniform way. The other option available for regional cases is EXP_SHAPE, where the flow is exponentially distributed. We will work with this last option and define in analytical.F:

```
# elif defined REGIONAL
#  define EXP_SHAPE
#  ifdef CST_SHAPE
        cff=1./float(N)
        do k=1,N                       ! Uniform vertical
          do is=1,Nsrc                 ! distribution
            Qshape(is,k)=cff
          enddo
        enddo
#  elif defined EXP_SHAPE
        do is=1,Nsrc
```

and save the changes.

We modify the section "*Point Sources - Rivers*" (line 240) where we will define the PSOURCE option

```
                      /* Point Sources - Rivers */
# define PSOURCE
# undef PSOURCE_NCFILE
# ifdef PSOURCE_NCFILE
#  undef PSOURCE_NCFILE_TS
# endif
```

and compile:

```
./jobcomp
```

3.4. Running the simulation

We use the same configuration of the rivers (psource) of croco.in proposed in the previous section, therefore modifying croco.in

```
nano croco.in
```

And configuring psource

```
psource:   Nsrc  Isrc  Jsrc  Dsrc  Qbar [m3/s]    Lsrc        Tsrc
            2
                  36    13     1    -20000.           T T      20. 2.
                  24    35     0    -20000.           T T      20. 2.
```

Now, we launch the simulation

```
./croco croco.in
```

Similar to the previous exercise, in CROCO_FILES we will find the croco_exp_his/avg.nc output files. If you have not succeeded in generating the files, then they will be able to download it from

```
http://mosa.dgeo.udec.cl/CROCO2021/Tutorial08/BENGUELA_R1A/CROCO_FILES/
```

The impact of the change in the vertical distribution can be observed with ncview, selecting the superficial salinity (using a range between 34-35.7 psu; Figure 3), where differences are shown with respect to the previous case (Figure 2).

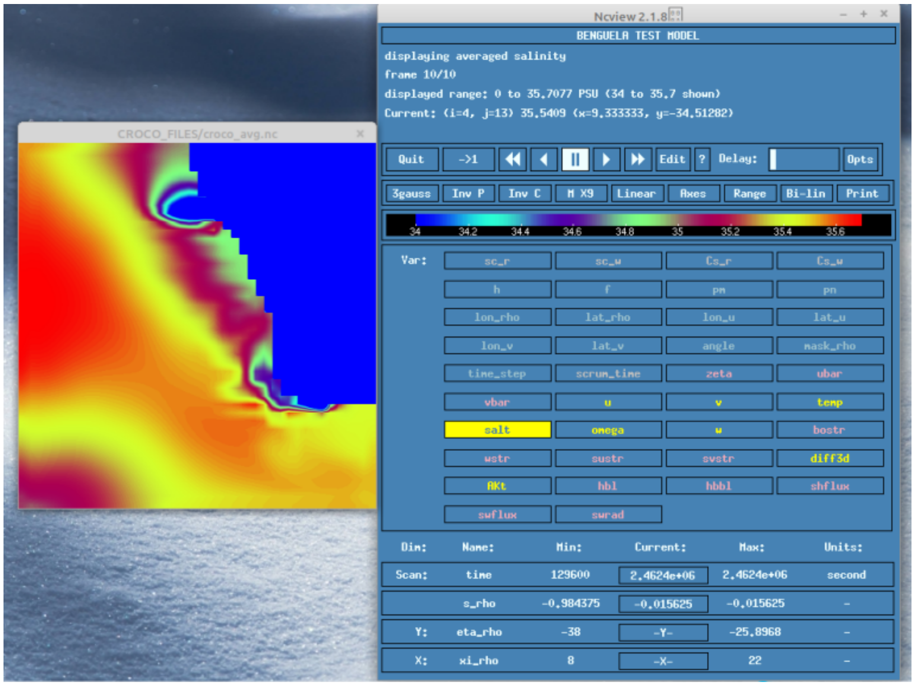

Figura 3: Case 1b: Surface salinity at t=10 obtained from croco_avg.nc (range 34-35.7 psu)

Additionally, Figure 4 shows vertical sections of salinity, perpendicular to the mouth of rivers 1 and 2, where the differences between using a constant distribution in the vertical (CST_SHAPE) and exponential (EXP_SHAPE) are observed.

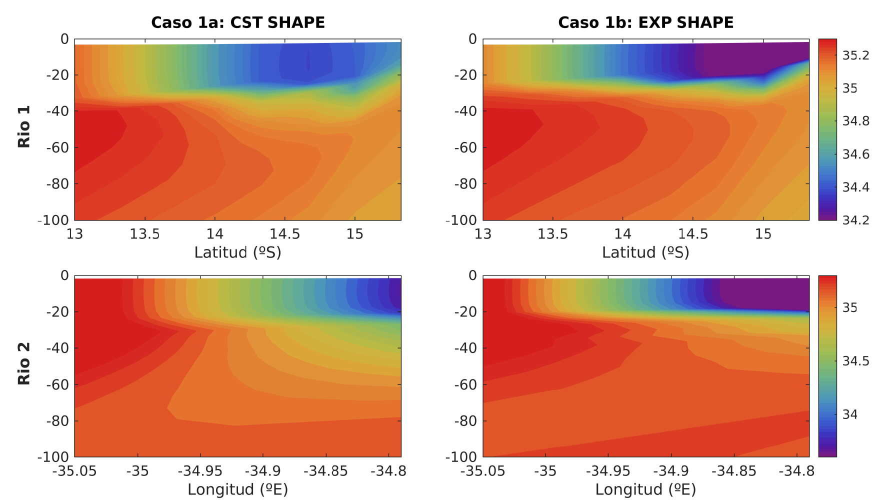

Figura 4: Vertical salinity sections, perpendicular to the mouth of River 1 (top) and 2 (bottom). Left (Right), Section using CST_SHAPE (EXP_SHAPE) settings

Note: It is recommended to modify analytical.F to keep the original configuration:

```
nano ../OCEAN/analytical.F
```

We look for the ana_psource subroutine, and in the section for regional cases (line 1883) we modify # define EXP_SHAPE to # define CST_SHAPE,

```
# elif defined REGIONAL
#  define CST_SHAPE
#  ifdef CST_SHAPE
        cff=1./float(N)
        do k=1,N                              ! Uniform vertical
          do is=1,Nsrc                        ! distribution
            Qshape(is,k)=cff
          enddo
        enddo
#  elif defined EXP_SHAPE
        do is=1,Nsrc
```

4. Conclusions

At the end of this guide, you will be able to implement rivers in your simulations, and modify their main properties

For more information:
Andrés Sepúlveda (asepulveda@dgeo.udec.cl)
Developed by:
Marcela Contreras
Collaborators:
Sebastian Inzunza
Mauro Santiago

If you found this tutorial useful, please send a postcard to:

Dr. Andrés Sepúlveda
Departamento de Geofísica
Casilla 160-C
Correo 3
Concepción
Chile

5. References

Dai, A., & Trenberth, K. E. (2002). Estimates of freshwater discharge from continents: Latitudinal and seasonal variations. Journal of Hydrometeorology, 3(6), 660-687.

ANNEXES

A. Position of the rivers

A slightly more complex aspect to define is the position of the rivers in the grid (Isrc, Jsrc). The position should be on the U/V face (depending on the flow) and on the border between land and sea (ie the coastline). So, to find the position it is recommended to look at the mask in psi coordinates (mask_psi in grd.nc) since its position corresponds to the edge of the cell and not to the center (as is case with rho coordinates).

To exemplify finding the position: If we have a north-south meridional flow at 34.65°S and 19.83°E (as is the case for Rio 1), then first finding the longitude position in the psi grid . For the BENGUELA_LR domain it is position 36. Since the flow is meridional, then it will traverse at the sea/land boundary of v-face. Then we must look for the most southerly position of the cell that corresponds to land (mask_psi=0). This position corresponds to 13 (See Figure A.1)

In case of River 2 the flow goes from east to west, so we look for the latitude position (position 34). Since it flows zonally, then the flow enters through the u-face, we look for the cell that is furthest to the west and corresponds to the earth mask (position 35; see Figure A.1).

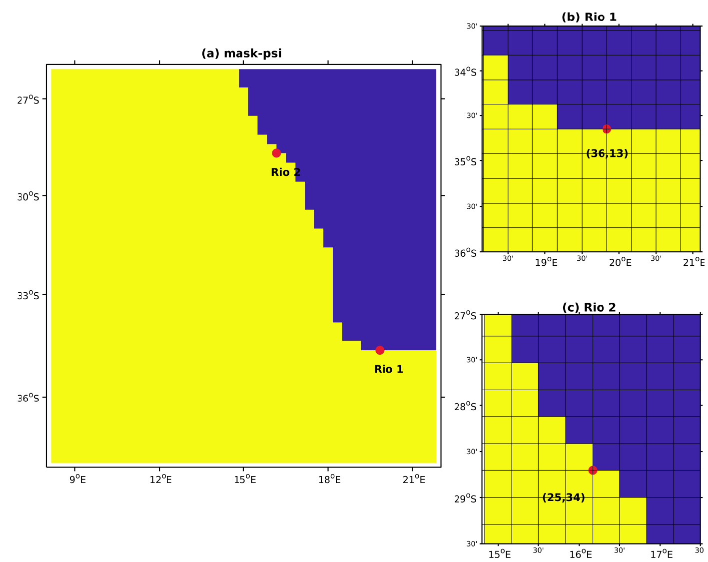

Figura A1: Vertical salinity sections, perpendicular to the mouth of Rio 1 (top) and 2 (bottom). Left (Right), Section using CST_SHAPE (EXP_SHAPE) settings

Here is an example routine for MATLAB/Octave to find the position of the rivers proposed in this document.

```
clear all
cd croco/BENGUELA_R1A
crocotools_param

maskp=ncread(grdname,'mask_psi')';
latp=ncread(grdname,'lat_psi')';
lonp=ncread(grdname,'lon_psi')';

posR1=[19.83 -34.65 ];
posR2=[16.16 -28.7]

%%
% RIO  1: Flujo de norte a sur. Se debe encontrar la cara v por donde fluye
% el flujo

% % Buscando Longitud
posxR1=find(lonp(1,:)>=posR1(1)-.1 & lonp(1,:)<=posR1(1)+.1)

% % Buscando latitud en que se encuentra el limite de la linea de costa, es
% decir el punto de las  mascara mas  al sur. Si el flujo fuera de norte a sur
% entonces se  buscaría la posición hacia el norte, entonces posyR1=max(find...
posyR1=min(find(maskp(:,posxR1)==0))

%%
% RIO  2: Flujo de este a oeste. Se debe encontrar la cara u por donde
% fluye el flujo
posyR2=find(latp(:,1)>=posR2(2)-.1 & latp(:,1)<=posR2(2)+.1)

% % Buscando latitud en que se encuentra el limite de la linea de costa, es
% decir el punto de las  mascara mas  al oeste. Si el flujo fuera de oeste a este,
% entonces se buscaria la  posición más hacia el este, y entonces posxR2=max(find(...
posxR2=min(find(maskp(posyR2,:)==0))
```