Tutorial 04 LiveCROCO: Simulation with Tides

1. Purpose

In this tutorial we will review how to perform a simulation of the Benguela domain including the effect of the tide on the boundaries, the important files for this type of simulation, and its validation.

2. Creating the MAREAS (tides in spanish) working directory

The simplest example of CROCO is the configuration called BENGUELA_LR which corresponds to a low resolution Benguela Upwelling Zonde domain. This is the default configuration in CROCO code and what we will do is similar to what is described in Penven et al. (2001)

We will assume that you already have a local copy of the CROCO code and CROCO_TOOLS in your root directory. This is achieved by following the steps in Tutorial 01.

First step is to edit **create_run.bash** file with the instructions to create a new working directory that we will call **MAREAS**

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

Now you have to modify this section to put the correct directories

```
#===============================================================================
# BEGIN USER MODIFICATIONS
#
# Get CROCO directory
CROCO_DIR="/home/livecroco/croco"
#
SOURCES_DIR="/home/livecroco/croco"
#
TOOLS_DIR="/home/livecroco/croco_tools"
#
MY_CONFIG_PATH=${SOURCES_DIR}
#
# Name of the configuration directory defined by the user
#
MY_CONFIG_NAME='MAREAS'
#
#
# END USER MODIFICATIONS
#===============================================================================

```

and then execute the statement

```
./create_run.bash
```

3. REGIONAL configuration with TIDES

For this tutorial we will add the tidal option to model physics.

3.1. MAREAS

We change to **MAREAS** directory

```
cd MAREAS
```

In **cppdefs.h** file we activate tides by modifying

```
                        /* Open Boundary Conditions */
# define  TIDES
```

activating this option has more consequences, as you can see below in the code

```
                      /* Open Boundary Conditions */
# ifdef TIDES
#  define SSH_TIDES
#  define UV_TIDES
#  define POT_TIDES
#  undef  TIDES_MAS
#  ifndef UV_TIDES
#   define OBC_REDUCED_PHYSICS
#  endif
#  define TIDERAMP
# endif
# define OBC_M2CHARACT
# undef  OBC_M2ORLANSKI
# define OBC_M3ORLANSKI
# define OBC_TORLANSKI
# undef  OBC_M2SPECIFIED
# undef  OBC_M3SPECIFIED
# undef  OBC_TSPECIFIED
```

To start, we will highlight the option

```
#  define TIDERAMP
```

this implies that tidal wave will be attenuated for the first three days, in order to avoid numerical instabilities in simulation.

3.2. Parallelization

Don't forget to turn on parallelization for your simulation.

```
                      /* Parallelization */
# undef  OPENMP
# define  MPI
```

4. Tides - param.h

Inside *param.h* setting **TIDES** option implies that

```
!----------------------------------------------------------------------
! Tides
!----------------------------------------------------------------------
!
#if defined SSH_TIDES || defined UV_TIDES
      integer Ntides             ! Number of tides
```

```
                                  ! ====== == =====
# if defined IGW
      parameter (Ntides=1)
# else
      parameter (Ntides=8)
# endif
#endif
```

that is, our model will use 8 tidal components. These are defined in section 5 of **crocotools_param.m**

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% 5 - Parameters for tidal forcing
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% TPXO file name (TPXO6 or TPXO7)
%
tidename=[DATADIR,'TPXO7/TPXO7.nc'];
%
% Self-Attraction and Loading GOT99.2 file name
%
sal_tides=1;
salname=[DATADIR,'GOT99.2/GOT99_SAL.nc'];
%
% Number of tides component to process
%
Ntides=10;
%
% Chose order from the rank in the TPXO file :
% "M2 S2 N2 K2 K1 O1 P1 Q1 Mf Mm"
% " 1  2  3  4  5  6  7  8  9 10"
%
tidalrank=[1 2 3 4 5 6 7 8 9 10];
%
% Compare with tidegauge observations
%
lon0 =  18.37;   % Example:
lat0 = -33.91;   % Cape Town location
Z0   =  1;       % Mean depth of tide gauge
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

That is, for tidal forcing we will be using the global solution TPX07 (Egbert and Erofeeva, 2002).

The option **sal_tides=1** implies a correction of 5 % of amplitude, a lag of 180° with ocal tide or even more in coastal areas.

```
https://www.esr.org/data-products/antarctic_tg_database/ocean-tide-and-ocean-tide-loading/
```

5. Compiling CROCO

Next we compile the model

```
./jobcomp
```

6. Input files

The input files that **croco** executable will read will be created with **CROCO_TOOLS**. It mainly depends on two files, **oct_start.m** and **crocotools_param.m**

6.1. MAREAS - Using Octave

To create input files using Octave, we first have to load the program using

```
octave-cli
```

Now the instructions to use, from **MAREAS** working directory are:

```
oct_start
make_grid
make_forcing
make_clim
```

in this case, we will add the instruction

```
make_tides
```

In terminal we will see

```
mkdir: cannot create directory '/home/livecroco/Desktop/MAREAS/CROCO_FILES/':
File exists
Start date for nodal correction : 1-Jan-2005
Reading CROCO grid parameters ...
Tidal components : M2 S2 N2 K2 K1 O1 P1 Q1 Mf Mm
Processing tide : 1 of 10
  ssh...
Getting ssh_r for time index 1
Getting ssh_i for time index 1
  u...
Getting u_r for time index 1
Getting u_i for time index 1
  v...
Getting v_r for time index 1
Getting v_i for time index 1
Convert to tidal ellipse parameters...
Process equilibrium tidal potential...
Process tidal loading and self-attraction potential...
Get total tidal potential...
```

Obtaining graphs of amplitude and phase of components used, for example for K1 Figs. 1 y 2

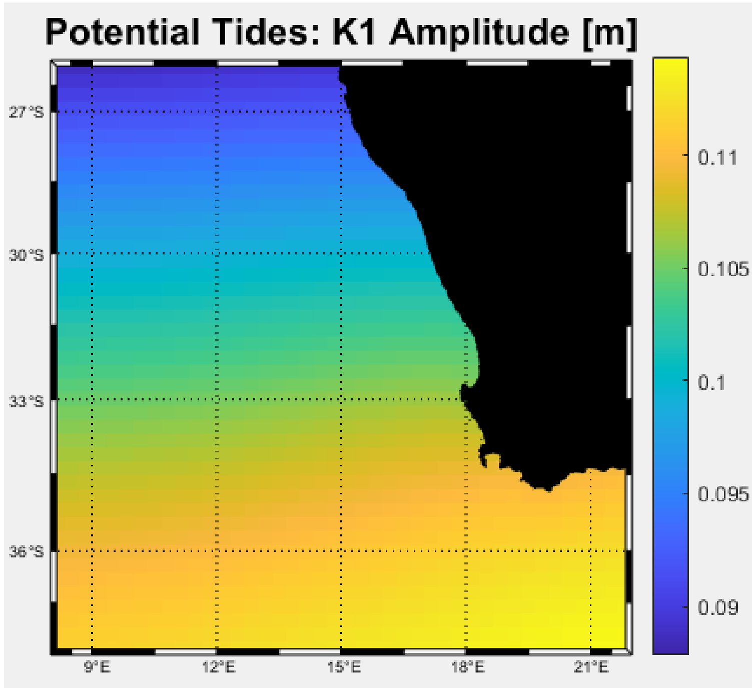

Figura 1: Amplitude K1 component

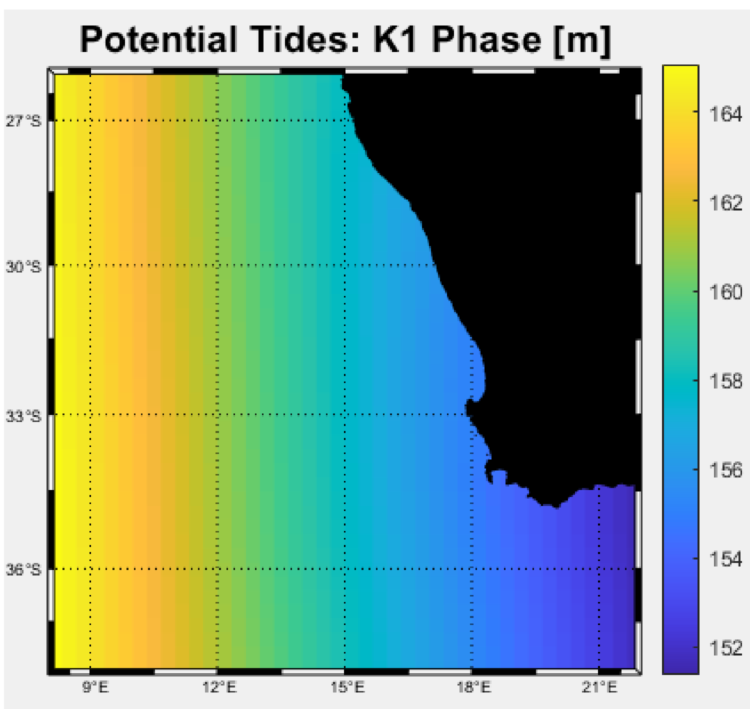

Figura 2: Phase K1 component

and a time series at a point in the domain (Fig. 3)

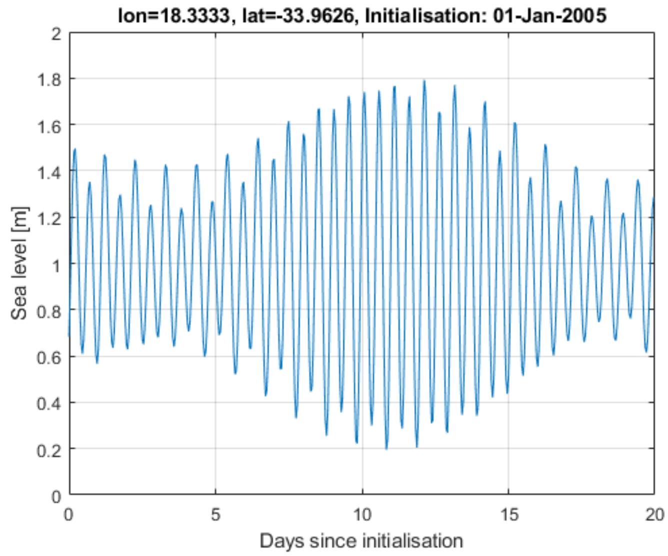

Figura 3: Time series at a point in the domain

This adds variables associated with tide **croco_frc.nc** file
The files you get should be the same as those located in

```
http://mosa.dgeo.udec.cl/LiveCROCO/Tutorial04/ArchivosIniciales/
```

If you had problems with this step, copy those files to **CROCO_FILES** directory to proceed to the next section.

7. MAREAS - croco.in

In analyzing the tides in our simulation it is important to modify **croco.in** file to record the output of the **HIS** file every 1 hour. Since the time step of this setup is 3600 seconds, we will record the output then every 1 time step

```
history: LDEFHIS, NWRT, NRPFHIS / filename
            T      1     0
    CROCO_FILES/croco_his.nc
```

Once this is done, we launch the simulation

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

Once simulation finishes successfully, we will find in **CROCO_FILES** directory the following output files

```
croco_avg.nc
croco_his.nc
croco_rst.nc
```

where the file of our interest will be **croco_his.nc**. However, since we use **TIDERAMP** option in **cppdefs.h** the sea level for first three days is unreal.
To correct this edit **cppdefs.h** again and configure

```
#  undef TIDERAMP
```

then recompile model with

```
./jobcomp
```

replace the **INI** file

```
cp croco_rst.nc croco_ini.nc
```

and launch simulation again

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

8. Tidal Simulation Validation

8.1. Simulation analysis

We will analyze our simulation and observations using **t_tide** program (Pawlowicz et al., 2002). A version of t_tide is already included in CROCO_TOOLS. For this we will use following code

```
https://raw.githubusercontent.com/AndresSepulveda/CROCO_plots/master/t_tide/analyze_t_tide.m
```

which was adapted from

```
https://isabeljalonrojas.com/analisis-de-armonicos-de-marea-astronomica-con-t_tide/
```

8.2. Analysis of observations

We will first get measurements for your zone from University of Hawaii Sea Level Center

`https://uhslc.soest.hawaii.edu/`

or on old website

`http://uhslc.soest.hawaii.edu/data/`

For example, we use the hourly data for Simon's Town, South Africa (latitude -34.18300, longitude 18.43300) valid between 1959-04-24 and 2020-08-31.

```
wget http://uhslc.soest.hawaii.edu/data/nc/fdh/OS_UH-FDH221_20170628_R.nc
```

The **analyze_t_tide.m** file is configured to use only 720h, since it is the same period that we will have available from numerical model.

```
octave-cli
>> oct_start
>> analyze_t_tide
```

what should i give

```
Add the paths of the different toolboxes
   Points used: 721 of 721
   percent of var residual after lsqfit/var original:  2.72 %
   Phases at central time
   Using nonlinear bootstrapped error estimates
   Generating prediction without nodal corrections, SNR is 2.000000
   percent of var residual after synthesis/var original:  2.82 %
   Points used: 721 of 721
   percent of var residual after lsqfit/var original:  2.72 %
   Phases at central time
   Using nonlinear bootstrapped error estimates
   Generating prediction without nodal corrections, SNR is 2.000000
   percent of var residual after synthesis/var original:  2.82 %

TOTAL =

  4×30 char array

    'M2  0.0805 0.502       250.504'
    'S2  0.0833 0.295       215.455'
    'N2   0.079  0.093       36.242'
    'K1  0.0418 0.054       310.811'

 Nombre Frecuencia Amplitud Fase
Factor de Forma (F)

F =

    0.0941

Marea Semidiurna
>>
```

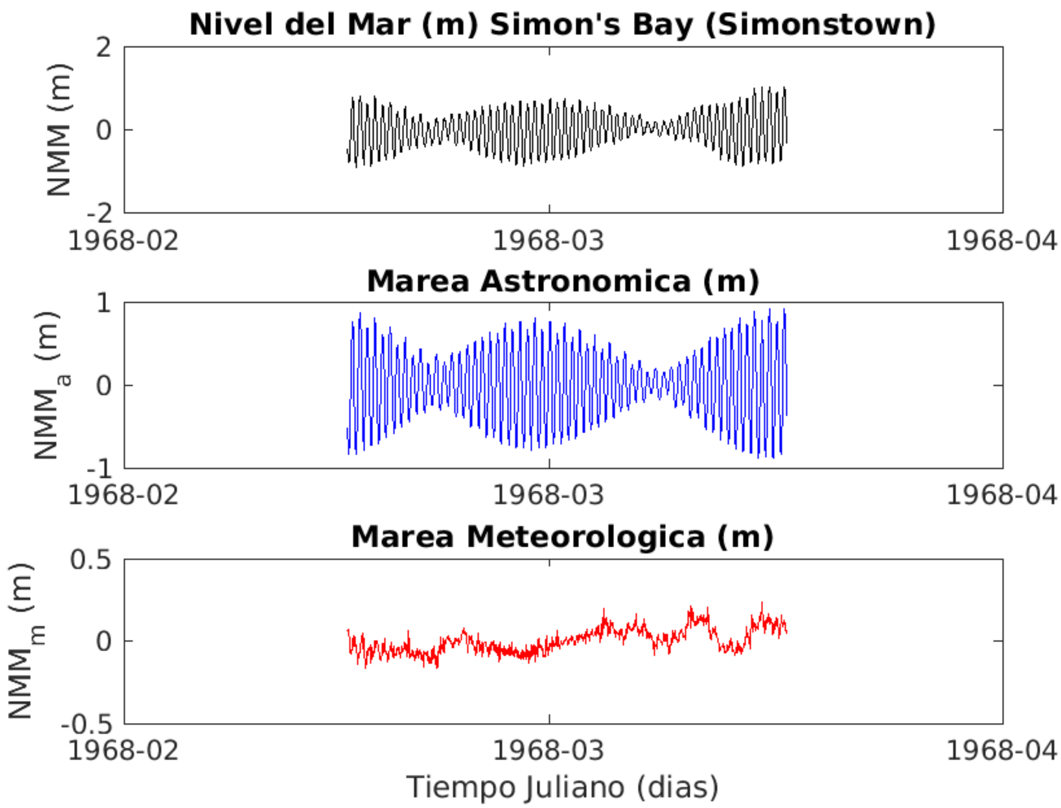

Figura 4: Time series for Simon's Town, South Africa. Observations.

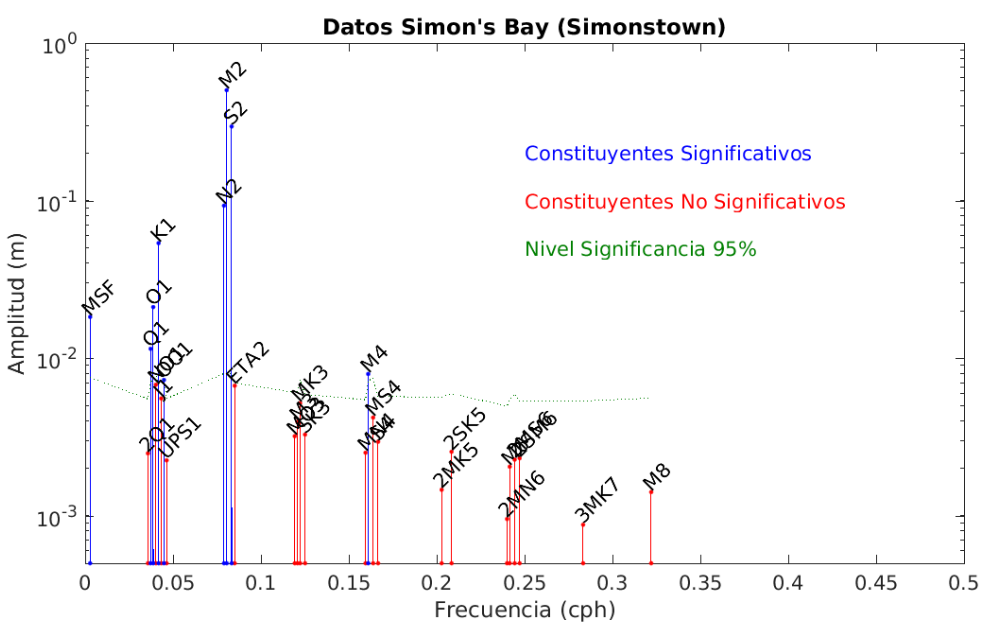

Figura 5: Tidal components for Simon's Town, South Africa. Observations

Compare the model results with UHSLC data using a table. It will be necessary to locate coordinate (i,j) of the grid point closest to the sea level station to be studied and extract time series from variable **zeta** in **croco_his.nc** file. An example of that is done in the file

```
wget http://mosa.dgeo.udec.cl/LiveCROCO/Tutorial04/Tides_Modelo_Benguela.m
```

When using it,

```
octave-cli
>> oct_start
>> Tides_Modelo_Benguela
```

should give figures similar to Figs. 6-7

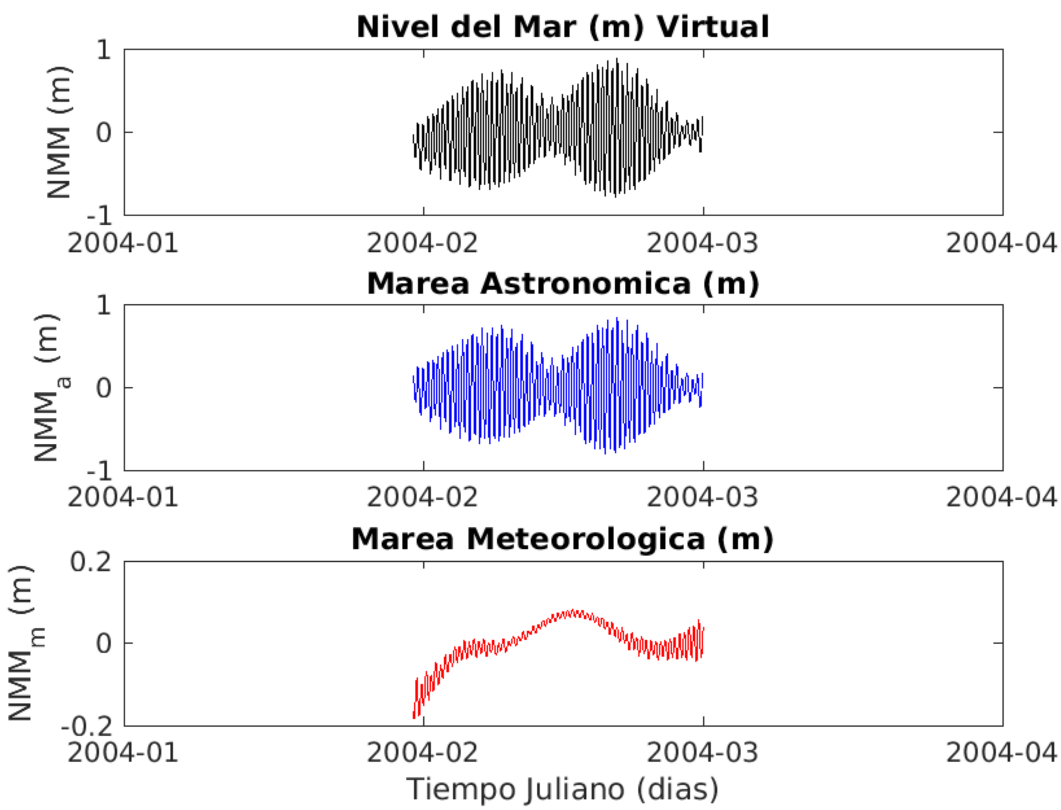

Figura 6: Time series for Simon's Town, South Africa. Model

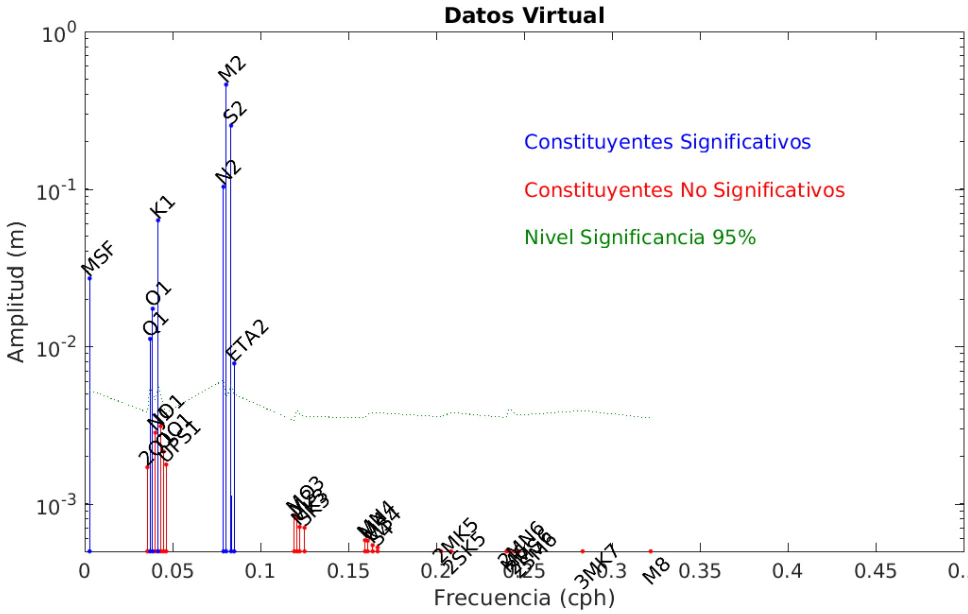

Figura 7: Tidal components for Simon's Town, South Africa. Model

and the text

```
Add the paths of the different toolboxes
   Points used:721 of 721
   percent of var residual after lsqfit/var original:  0.28 %
   Phases at central time
   Using nonlinear bootstrapped error estimates
   Generating prediction without nodal corrections, SNR is 2.000000
   percent of var residual after synthesis/var original:  0.29 %
   Points used: 721 of 721
   percent of var residual after lsqfit/var original:  0.28 %
   Phases at central time
   Using nonlinear bootstrapped error estimates
   Generating prediction without nodal corrections, SNR is 2.000000
   percent of var residual after synthesis/var original:  0.29 %

TOTAL =

  4×30 char array

    'M2  0.0805 0.461       290.979'
    'S2  0.0833 0.252        88.38'
    'N2   0.079  0.104      272.562'
    'K1  0.0418 0.065        87.148'

 Nombre Frecuencia Amplitud Fase
Factor de Forma (F)

F =

    0.1164

Marea Semidiurna
>>
```

Construct a table with amplitude and phase of significant tidal components of model and observations, including the respective uncertainty. Use at most 2 significant figures. If you analyze several stations you can make a graph like Figure 3 of Artal et al. (2019).

If you are interested in inverse problems, you can try using Leffler & Jay (2009) or Codiga (2011) code, which contain more robust parameter estimation. Other codes (NS_TIDE) allow non-stationary harmonics to be compared (Matte et al. 2013).

9. Advanced Work

Run a simulation for a domain of interest and analyze tidal signal.

10. Conclusion

In this tutorial you learned more details about **cppdefs.h**, **param.h**, and **croco.in** files, and modifications that must be made to make a simulation of tides, as well as attributes of results validation.

For more information:
Andrés Sepúlveda (asepulveda@dgeo.udec.cl)
Collaborators of:
Marcela Contreras
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

11. References

Artal, O., Pizarro, O., & Sepúlveda, H. H. (2019). The impact of spring-neap tidal-stream cycles in tidal energy assessments in the Chilean Inland Sea. Renewable Energy, 139, 496-506.

Codiga, D.L., 2011. Unified Tidal Analysis and Prediction Using the UTide Octave Functions. Technical Report 2011-01. Graduate School of Oceanography, University of Rhode Island, Narragansett, RI. 59pp. https://github.com/wesleybowman/UTide

Egbert, Gary D., and Svetlana Y. Erofeeva. "Efficient inverse modeling of barotropic ocean tides." Journal of Atmospheric and Oceanic Technology 19.2 (2002): 183-204.

Leffler, K. E., & Jay, D. A. (2009). Enhancing tidal harmonic analysis: Robust (hybrid L1/L2) solutions. Continental Shelf Research, 29(1), 78-88.

Matte, P., Jay, D.A., Zaron, E.D., 2013. Adaptation of classical tidal harmonic analysis to nonstationary tides, with application to river tides. J. Atmos. Ocean. Technol. 30, 569{589. doi:10.1175/JTECH-D-12-00016.

Pawlowicz, R., Beardsley, B., & Lentz, S. (2002). Classical tidal harmonic analysis including error estimates in Octave using T_TIDE. Computers & Geosciences, 28(8), 929-937.

Penven, P., Roy, C., Brundrit, G. B., De Verdière, A. C., Fréon, P., Johnson, A. S., Lutjeharms J. R. E. & Shillington, F. A. (2001). A regional hydrodynamic model of upwelling in the Southern Benguela. South African Journal of Science, 97(11-12), 472-475.